CrossMark

Reversible data hiding for 3D mesh models with three-dimensional prediction-error histogram modification

Ruiqi Jiang[1] · Weiming Zhang[1] · Dongdong Hou[1] · Hui Wang[1] · Nenghai Yu[1]

© Springer Science+Business Media New York 2017

Abstract This study proposes a reversible data hiding (RDH) algorithm for 3D mesh models based on the optimal three-dimensional prediction-error histogram (PEH) modification with recursive construction coding (RCC). Firstly, we design a double-layered prediction scheme to divide all the vertices of 3D mesh model into the "embedded" set and the "referenced" set, according to the odd-even property of indices into the vertex list. Thanks to the geometrical similarity among neighboring vertices, we obtain the prediction errors (PEs) with a sharp histogram. Then we combine every three adjacent PEs into one prediction-error triplet (PET) and construct the three-dimensional PEH with smaller entropy than one-dimensional PEH by utilizing the correlation among PEs. Next we project the three-dimensional PEH into one-dimensional space for scalar PET sequence which is suitable for using RCC. And also, we define the distortion metrics for 3D mesh models, by which we can estimate the optimal probability transition matrix (OTPM) indicating the optimal PEH modification manner. After that, we modify the PET sequence and embed data by RCC according to OTPM. The experimental results show that our method is superior to two state-of-the-art spatial-domain RDH algorithms for 3D mesh models a lot.

1 Introduction

Data hiding is applied extensively in the community of signal processing, such as ownership protection, fingerprinting, authentication and secret communication, but the most

✉ Weiming Zhang
zhangwm@ustc.edu.cn

[1] Key Laboratory of Electromagnetic Space Information, Chinese Academy of Sciences, School of Information Science and Technology, University of Science and Technology of China, Hefei 230026, China

Springer

classical data hiding technique will lead to permanent distortions. In the last decades, reversible data hiding (RDH) [1, 3, 4, 8, 9, 14, 16, 18–20, 23, 24] as a new type of information hiding technique, has received much attention. RDH provides not only extracting embedded data precisely, but also restoring the original cover image without any distortion. This special property has been found much useful in many sensitive fields, such as medical imagery, military imagery and law forensics, where any change on the host signal is not allowed. There are two important metrics to evaluate the performance of an RDH algorithm: the embedding capacity and the distortion. In fact, we always keep the balance between such two metrics, because a higher embedding capacity often cause more distortion and decreasing the distortion will result in a lower embedding capacity (EC). The Peak-Signal-to-Noise-Ratio (PSNR) value is usually used to evaluate the performance of RDH algorithm.

The classic RDH algorithms mainly fall into three categories, lossless compression [3], difference expansion (DE) [18] and histogram shifting (HS) [14]. Recently, many researchers apply DE or HS to more centrally distributed prediction error [PE] to achieve better performance because it exploits the correlation between neighboring pixels [1, 4, 8, 9, 16, 19, 20]. This extended method is called prediction-error expansion (PEE), which has been proved to be the most powerful technique of RDH.

Almost all state-of-the-art PEE algorithms consist of two steps. The first step generates a PE sequence with small entropy which can be realized by utilizing kinds of pixel prediction strategies combined with the sorting or pixel selection technique. The second step reversibly embeds the message into the PE sequence by modifying its histogram with methods like HS and DE, and recursive construction coding (RCC) is proved to be the optimized histogram modification in this process.

The research of RDH for digital image has last for many years and the algorithm performance has already approached to the theoretical optimal solution, but for other digital multimedia, such as audio, video, especial for 3D models, it is still at the starting stage. With the rapid development of 3D related applications, 3D models have been widely used on the internet, no matter mesh models or point cloud models, so researchers gradually try to adopt 3D models as the cover media for RDH. The former RDH for 3D models can be classified into three domains: spatial-domain, compressed-domain and transform-domain [2, 11–13, 17, 21, 22]. Spatial-domain-based RDH [2, 21, 22] just modifies the vertex coordinates thus it has low computational complexity. Compressed-domain-based RDH [11, 17] is proposed in the predictive vector quantization (VQ) domain to embed data into a mesh stream. Transform-domain-based RDH [12, 13] transforms the original models into a certain transform domain and then embed data into the transform coefficients. Objectively, there exists much space to improve the performance for 3D model RDH.

In this paper, we design a double-layered prediction scheme to obtain the PEs with sharp histogram and then propose an optimal three-dimensional PEH modification method using RCC, the purpose is to maximize the EC for the same distortion or minimize the distortion for the same EC for 3D mesh model RDH. Firstly, we divide all the vertices of the input models into the "embedded" set and the "referenced" set according to the odd-even property of their indices into the vertex list, then select vertices on the "embedded" set for data embedding, and use their neighboring vertices on the "referenced" set to estimate the predicted position. By calculating the difference between the real position and the predicted position, we obtain PEs. Because of the geometric similarity among neighboring vertices, the PEH describing the distribution of PEs will be very steep, which is beneficial for RDH.

Secondly, by exploiting the correlation among PEs, through combining every three adjacent PEs into one prediction-errors triplet (PET), we construct 3D PEH, and then use an injective mapping function to project 3D PEH into the 1D space for scalar PET sequence. In addition, we define the distortion metrics for 3D models so as to estimate the OTPM which means the optimal PEH modification manner. Next, we modify the PET sequence and embed data using RCC to achieve OTPM. Finally we modify the coordinate for the vertices on the "embedded" set without changing the topology, and at the same time, the vertices on the "referenced" set keep unchanged for sake of data extraction and recovery completely. It should be mentioned that the prediction and embedding procedure can be operated on the "embedded" set and the "referenced" set alternatively for several rounds so as to increase the EC if the distortion is under the limit. Compared to two existing RDH algorithms for 3D mesh models, we can decrease the distortion for the same EC and increase the EC for the same distortion restriction. Experimental results prove that our proposed method has superior performance and it is feasible for 3D mesh model RDH considering the capacity and distortion.

The rest of this paper is organized as follows. Section 2 briefly introduces the RCC for reversible data hiding. The proposed 3D PEH modification method is presented in Section 3. The experimental results and some discussion are demonstrated in Section 4. And finally, the conclusion is drawn in Section 5.

2 Overview for RDH by recursive construction coding

As mentioned above, the PE based RDH methods embed messages by modifying the prediction errors with sharp histogram. Many schemes have been proposed to chase a better balance between the embedding capacity and the distortion. Actually, for a given PE sequence, with a distortion constraint, the upper bound of the payload is given by Kalker and Willems in [7]. They formulated the RDH as a special rate-distortion problem, and obtained the rate-distortion function, i.e., the upper bound of the embedding rate under a given distortion constraint Δ, as follows:

$$\rho(\Delta) = Maximize\{H(Y)\} - H(X) \tag{1}$$

Where X and Y denote the random variables of host sequence and marked sequence respectively. The maximum entropy is over all transition probability matrices $P_{Y|X}(y|x)$ satisfying the distortion constraint $\sum_{x,y} P_X(x)P_{Y|X}(y|x)D(x, y) \leq \Delta$. The distortion metric $D(x, y)$ is usually defined as the square error distortion, $D(x, y) = (x - y)^2$.

Therefore, instead of using expanding and shifting technique for the given PEs, Lin et al. [10] proposed a pixel by pixel recursive code construction (RCC) to approach the rate distortion bound for square error distortion. And alternatively, by improving RCC, Zhang et al. [24] obtain the optimal embedding method for general gray-scale PE sequence, which performs in a bin by bin manner respectively.

Assume that the input PE sequence is a single probability distribution histogram, we briefly elaborate the optimal PEH modification procedure to embed data with RCC.

–   Firstly, using the prediction strategy such as rhombus prediction [16] or two dimensional prediction [15], a PE sequence is obtained, which acts as the host signal X in (1).

–   Secondly, in order to minimize the embedding distortion for a fixed length of mes-
    sage payload, use the fast algorithm in [5] to estimate the optimal marginal distribution
    $P_Y(y)$ of the marked signal Y.
–   Finally, according to the estimated marginal distribution $P_Y(y)$, by modifying the host
    PE sequence X to the marked PE sequence Y, messages can be optimally embedded
    through Lin et al.'s code construction [10].

Zhang et al. propose another optimal PEH modification scheme by a bin by bin manner,
which is shown in Fig. 1. And the data extraction and cover restoration are executed in a
backward manner. For more detail about RCC method, we refer readers to [10].

It is proved that, for a given host signal and defining the distortion restriction, RCC can
approach the rate-distortion bound of RDH for digital images. However, if we want to apply
RCC for our 3D case, there exists some problems needed be solved.

–   Firstly, how to construct a host signal (PE sequence) with small entropy? Observing
    from the rate-distortion formulation (1), a small value of $H(X)$ means more capacity
    exists in the difference between the two entropies $H(Y) - H(X)$, thus better embedding
    performance can be achieved.
–   Secondly, if three-dimensional PEH with the smaller entropy than one-dimensional
    PEH by utilizing the correlation about PEs, how to project the three-dimensional PEH
    into one-dimensional space to get a scalar sequence which is suitable for RCC?
–   Finally, what is the distortion metrics for 3D mesh models by which we can estimate the
    optimal transition probability matrix (OTPM) indicating the optimal PEH modification
    manner?

3 Proposed method

As for the problems we mentioned in Section 2, we propose the solution scheme in this sec-
tion so as to apply RCC for 3D mesh model RDH. Basically, our proposed RDH scheme is
based on three-dimensional PEH modification. Firstly, we design a double-layered predic-
tion method to construct the prediction errors with shape histogram. Then we combine the
adjacent three PEs as a prediction-error triplet (PET) and manipulate the three-dimensional
PEH which has the smaller entropy than one-dimensional PEH. With an injective mapping
function, we project the three-dimensional PEH into one-dimensional PET sequence which
is suitable for RCC. At last, we define the distortion metrics which is also significant for
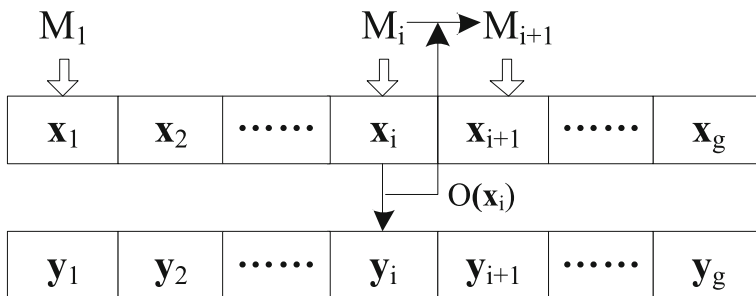guiding the optimal histogram modification manner.

Fig. 1 Illustration of recursive construction coding with a bin by bin manner

3.1 Prediction scheme

A 3D model represents a 3D object using a collection of points in the 3D space, connected by various geometric entities such as triangles, lines, curved surfaces, etc. Among various of representations for 3D model, polygonal meshes are the most popular by far, thus we use polygonal mesh models in .ply format as our experimental model. For a .ply file, it consists of a header followed by a list of vertices and then a list of polygons, the header specified how many vertices and polygons are in the file, and also states what properties are associated with each vertex, such as (x, y, z) coordinates, normals and color. The vertex list contains the indices of vertices, x-axis coordinate, y-axis coordinate, and z-axis coordinate, and the face list defines the polygon faces which are simply lists of indices into the vertex list, and each face begins with a count of the number of elements in each list.

A polygonal mesh models consists of a set of vertices, as denoted by $V = \{v_1, v_2, ... v_N\}$ and a vertex position v_i specifies the coordinates $\{v_{ix}, v_{iy}, v_{iz}\}$ in $\Re^3$. The mesh topology means the connectivity between vertices. Two vertices are termed neighbors if one edge connects them so that the vertices in the same face are co-neighbors, thus we can record the neighbors for each vertex efficiently by traversing the face list in order to find the faces which contain the given vertex. Based on this file format property, we design the double-layered prediction scheme for 3D mesh models. It should be noted that, this prediction method is also significant for 3D models in .obj format which is another popular file format for 3D mesh models.

Depending on the odd-even property of indices into the vertex list, we divide the vertices into two sets: the "embedded" set and the "referenced" set. The "embedded" set is used for embedding data and the "referenced" set is used for prediction, and if needed, we can use the "referenced" set to embed data and use the modified "embedded" set for prediction at the second round. This alternative procedure can be operated several rounds to increase the EC if the distortion is under the limited. It should be mentioned here, without loss of generality, these two sets are independent with each other, which means changing the coordinates of vertices on the "embedded" set do not affect vertices on the "referenced" set, and vice versa. With the use of this independent property, we make sure the coordinates of predicted vertices keep unchanged, for sake of extracting data and restoring cover losslessly.

Since the multi-round prediction and data embedding process are similar, we only use the first-round procedure for illustration, which is shown in Fig. 2, and Table 1 shows the file format for this local region. In Fig. 2, the labeled numbers are vertex indices into the
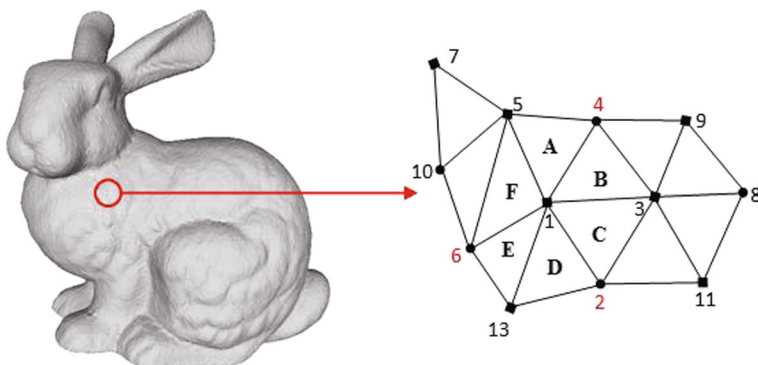

Fig. 2 The illustration of the prediction method

Table 1 File format for Fig. 2

Vertex List				Face Information	
Index of vertex	x-axis coordinate	y-axis coordinate	z-axis coordinate	count of the number of elements in each face	elements in each face
1	v_{1x}	v_{1y}	v_{1z}	3	(1,4,5)
2	v_{2x}	v_{2y}	v_{2z}	3	(1,3,4)
3	v_{3x}	v_{3y}	v_{3z}	3	(1,2,3)
4	v_{4x}	v_{4y}	v_{4z}	3	(1,2,13)
5	v_{5x}	v_{5y}	v_{5z}	3	(1,6,13)
6	v_{6x}	v_{6y}	v_{6z}	3	(1,5,6)
7	v_{7x}	v_{7y}	v_{7z}	3	(2,3,11)
8	v_{8x}	v_{8y}	v_{8z}	3	(3,4,9)
9	v_{9x}	v_{9y}	v_{9z}	3	(3,8,9)
10	v_{10x}	v_{10y}	v_{10z}	3	(3,8,11)
11	v_{11x}	v_{11y}	v_{11z}	3	(5,6,10)
12	v_{12x}	v_{12y}	v_{12z}	3	(5,7,10)
13	v_{13x}	v_{13y}	v_{13z}		

vertex list, and the vertices with odd index are defined as the "embedded" set for embedding (square point), while the vertices with even index belong to the "referenced" set for prediction (circle point). We traverse all the vertices on the "embedded" set in ascending order by their indices and find their neighbors on the "referenced" set. Through calculating the difference between the real position and predicted position, we construct the PEs sequence. For example, 1 is firstly traversed, we traverse the face list to find the faces containing 1, by this we record 2, 3, 4, 5, 6, 13 as its neighboring vertices, and then we select the neighboring vertices belong to the "referenced" set, which are 2, 4, 6, to predict the position of 1, because they will not be changed in the first-round embedding process for sake of reversibility. Similarly, we use 2, 4, 8 to predict 3, and, 4, 6, 10 for 5, keep this way until all the vertices on the "embedded" set have been predicted.

It should be mentioned here, we predict the position of the vertices on the "embedded" set by calculating the average coordinate value of their neighboring vertices belonging to the "referenced" set. Just taking vertex 1 as an example, the real coordinate of v_1 $=\{v_{1x}, v_{1y}, v_{1z}\}$, and the predicted coordinate of $v_1' =\{v_{1x}', v_{1y}', v_{1z}'\}$, the prediction-error e_1 can be represented by:

$$
\begin{aligned}
e_1 &= \{e_{1x}, e_{1y}, e_{1z}\} = \{v_{1x} - v_{1x}', v_{1y} - v_{1y}', v_{1z} - v_{1z}'\} \\
v_{1x}' &= \tfrac{1}{3}(v_{2x} + v_{4x} + v_{6x}) \\
v_{1y}' &= \tfrac{1}{3}(v_{2y} + v_{4y} + v_{6y}) \\
v_{1z}' &= \tfrac{1}{3}(v_{2z} + v_{4z} + v_{6z})
\end{aligned}
\tag{2}
$$

In previous work [6], the distance difference between real position and predicted position was usually used for embedding data so that it only embedded one data by per target vertex. However, in our proposed algorithm, through modifying $\{e_{ix}, e_{iy}, e_{iz}\}$, we can embed three bits of data with the use of e_i, in this way, we can improve EC three times if the number of target vertices is the same. Therefore, we calculate the difference for traversing all the vertices on the "embedded" set and then obtain the prediction errors which contain

$\{e_{1x}, e_{1y}, e_{1z}, e_{3x}, e_{3y}, e_{3y}...e_{(N-1)x}, e_{(N-1)y}, e_{(N-1)z}\}$ if there is N vertices in the mesh models and N is even. Because of the geometric similarity of neighboring vertices, the prediction is accurate so that the PE sequence obeys a Laplacian-like distribution centered at 0 or close to 0, which is shown in Fig. 3. Due to the restriction for showing composite 3D PEH in the paper, we only give the separated PEH for x-axis, y-axis and z-axis respectively.

It should be noted that, followed by [21], we also use truncation function to truncate the part under the precision level. If the coordinates in the original mesh are at the precision level of 10^{-n}, before our prediction, all the coordinates of vertices multiply by 10^m and truncate the decimal part, where $m \leq n$ and m is usually set to 3 or 4. By this, we operate the embedding algorithm on integer arithmetic, thus we need to divide the coordinate of vertices by 10^m after data embedding to reconstruct the marked mesh models.

For the PE sequence obtained by our prediction scheme, we can generate one-dimensional PEH by counting the frequencies of PEs, then embed data by modifying the PEH through expansion, shifting, even RCC which has been proved the optimal histogram modification manner in Section 2. But intuitively, it is not the best choice to rudely manipulate the one-dimensional PEH because it has not exploited the spatial correlation for 3D models. In our 3D case, the PE sequence $\{e_{1x}, e_{1y}, e_{1z}, e_{3x}, e_{3y}, e_{3z}...\}$ is not independently distributed. In fact, adjacent PEs are usually highly correlated, especially for $\{e_{ix}, e_{iy}, e_{iz}\}$, because the difference between real position and predicted position for v_i is:

$$d_i = \sqrt{(e_{ix})^2 + (e_{iy})^2 + (e_{iz})^2} \qquad (3)$$

One words to remark, there also exits the correlation between $\{e_{ix}, e_{iy}, e_{iz}\}$ and $\{e_{jx}, e_{jy}, e_{jz}\}$, but it is beyond the scope of this paper and would be done in our future work. Thanks to the correlation property, we propose to construct a three-dimensional PEH where the correlations within PET can be exploited, in order to achieve a better performance for RDH by decreasing the entropy of host signal X. By considering every three adjacent PEs together, the sequence $\{e_{1x}, e_{1y}, e_{1z}, e_{3x}, e_{3y}, e_{3z}...\}$ can be transformed into a new PET sequence $\{e_1, e_3, ...e_{N-1}\}$ with $\{e_i = e_{ix}, e_{iy}, e_{iz}, i = 1, 3...N-1\}$, and the associated 3D PEH is

$$g(k_1, k_2, k_3) = \#\{1 \leq i \leq N - 1 : e_{ix} = k_1, e_{iy} = k_2, e_{iz} = k_3\} \qquad (4)$$

Where # denotes the cardinal number of a set. For simplicity, the sequence $\{e_{1x}, e_{1y}, e_{1z}, e_{3x}, e_{3y}, e_{3z}, ..., e_{(N-1)x}, e_{(N-1)y}, e_{(N-1)z}\}$ is denoted as (X, Y, Z) and $\{e_{1x}, e_{3x}, ..., e_{(N-1)x}\}, \{e_{1y}, e_{3y}, ..., e_{(N-1)y}\}, \{e_{1z}, e_{3z}, ..., e_{(N-1)z}\}$ are denoted as X, Y, Z, respectively. The function H calculates the entropy of a distribution and we can clearly get the inequation as follows

$$H(X, Y, Z) \leq H(X) + H(Y) + H(Z) \qquad (5)$$

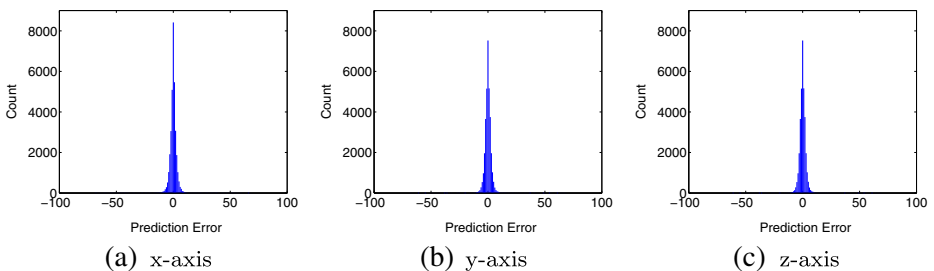(a) x-axis   (b) y-axis   (c) z-axis

Fig. 3 Prediction-Error Histogram

It demonstrates that three-dimensional PEH utilizes the correlation among PEs compared to one-dimensional PEH and has a potential to yield an improved RDH, so it is significant to combine every three adjacent PEs into one PET, from the beginning to the end of the PE sequence.

3.2 3D PEH to 1D PET sequence

To apply RCC for our 3D case, we project the 3D PEH into the 1D space to get a scalar sequence by using an injective mapping function $Project()$ defined by

$$project(e_i) = project(e_{ix}, e_{iy}, e_{iz}) = (e_{ix} + T) \times (2T + 1)^2 + (e_{iy} + T) \times (2T + 1) + (e_{iz} + T) \tag{6}$$

For simplicity, we set $T = 1$ and use Fig. 4 to show the injective mapping relationship.

For the PE sequence, we set a threshold parameter T to avoid modifying the PEs with too large magnitude so that the selected PET meets $|e_{ix}| \leq T$, $|e_{iy}| \leq T$ and $|e_{iz}| \leq T$. And also, Euclidean distance between the real position and predicted position measures the prediction accuracy, thus it is reasonable to utilize the Euclidean of $\{e_{ix}, e_{iy}, e_{iz}\}$ as our sorting method and we set the threshold T_{ED} for selecting PET. The sorting technique benefits RCC, because a sharper histogram has a smaller entropy so that better rate distortion performance is gained, and also, the sorting technique shortens the bit length of the compressed PEH since the number of PEs is decreased. It should be noted that the amplitude threshold T and sorting threshold T_{ED} can be tuned to achieve different results, for the given embedding rate. How to find the best values for these threshold parameters in the proposed method remains a problem to be researched in the subsequent work.

3.3 The distortion matrix

For one-dimensional PE sequence, $D(x, y)$ is usually defined as the square error distortion, $D(x, y) = (x - y)^2$, because the algorithm performance for RDH is evaluated by:

$$PSNR = 10\log_{10} \frac{255^2}{\frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} (b_{ij} - b'_{ij})^2} db \tag{7}$$

In our 3D case, the geometrical distortion of a mesh model including N vertices after adding some noise to the mesh content can be measured by SNR which is defined as follows:

$$SNR = 10\log_{10} \frac{\sum_{i=1}^{N} [(v_{ix} - \bar{v}_x)^2 + (v_{iy} - \bar{v}_y)^2 + (v_{iz} - \bar{v}_z)^2]}{\sum_{i=1}^{N} [(g_{ix} - v_{ix})^2 + (g_{iy} - v_{iy})^2 + (g_{iz} - v_{iz})^2]} \tag{8}$$

$T=1 \longrightarrow project(x, y, z)=project(x+1)*9+project(y+1)*3+project(z+1)$

$X=(-1,0,1)$
$Y=(-1,0,1)$ $\longrightarrow$
$Z=(-1,0,1)$

0	1	2	3	4	5	6	7	8
(-1,-1,-1)	(-1,-1,0)	(-1,-1,1)	(-1,0,-1)	(-1,0,0)	(-1,0,1)	(-1,1,-1)	(-1,1,0)	(-1,1,1)
9	10	11	12	13	14	15	16	17
(0,-1,-1)	(0,-1,0)	(0,-1,1)	(0,0,-1)	(0,0,0)	(0,0,1)	(0,1,-1)	(0,1,0)	(0,1,1)
18	19	20	21	22	23	24	25	26
(1,-1,-1)	(1,-1,0)	(1,-1,1)	(1,0,-1)	(1,0,0)	(1,0,1)	(1,1,-1)	(1,1,0)	(1,1,1)

Fig. 4 Mapping relationship

Where $\{\bar{v}_x, \bar{v}_y, \bar{v}_z\}$ is the mean coordinates of models, $\{v_{ix}, v_{iy}, v_{iz}\}$ is the original coordinates of v_i, $\{g_{ix}, g_{iy}, g_{iz}\}$ is the modified coordinates of v_i. Clearly, from (10), the distortion metric $D(e_i, e_i')$ should be defined by the Euclidean distance between the corresponding PET e_i and e_i':

$$D(e_i, e_i') = (e_{ix}' - e_{ix})^2 + (e_{iy}' - e_{iy})^2 + (e_{iz}' - e_{iz})^2 \tag{9}$$

As mentioned above, the distortion metric is significant when using RCC for data embedding, because it is related to the OTPM. For example, as for the PET $(e_{1x}, e_{1y}, e_{1z}) = (0, 0, 0)$, by embedding data, it could be changed to $(0, 0, 0)$, $(0, 0, 1)$, $(0, 1, 0)$, $(1, 0, 0)$, $(0, 1, 1)$, $(1, 0, 1)$, $(1, 1, 0)$, $(1, 1, 1)$, as shown in Fig. 5a, but the distortion with these eight situations are different, with the guidance of distortion metrics, we hope to modify $(0, 0, 0)$ to $(0, 0, 0)$, $(0, 0, 1)$, $(0, 1, 0)$, $(1, 0, 0)$ with high probability, but not $(0, 1, 1)$, $(1, 0, 1)$, $(1, 1, 0)$, $(1, 1, 1)$, as shown in Fig. 5b, unless the available EC provided by $(0, 0, 0)$, $(0, 0, 1)$, $(0, 1, 0)$ and $(1, 0, 0)$ is insufficient. In summary, after selecting the PET with higher prediction accuracy, we also want to embed data with less distortion.

3.4 Procedure of proposed algorithm

With double-layered prediction scheme, we obtain the PE sequence, and using (6), we project the 3D PEH into 1D space for scalar PET sequence which is suitable for RCC. After defining the distortion metric, we can estimate the optimal $P_{Y|X}(y|x)$ for the given restriction by using a two-stage scheme described in [15]. Firstly, we use the fast algorithm in [5] to get the marginal distribution $P_Y(y)$ and the corresponding dual variable. Secondly, we use the solution for linear program in [25] to finally estimate the OTPM, which indicates the optimal modification manner. After that, we can modify the host sequence into the marked sequence and optimally embed data through RCC. Finally, by adding the marked PE sequence to the prediction position and refreshing the vertex coordinate, we can get the marked 3D mesh model.
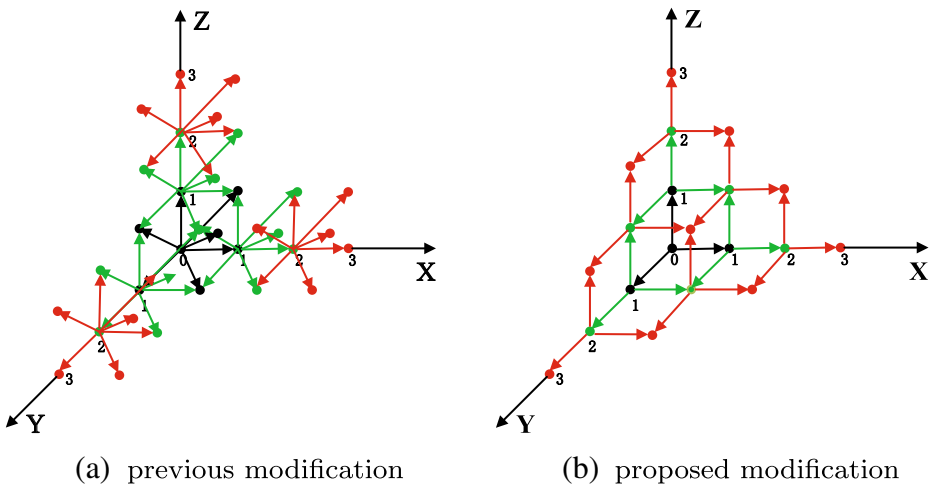

(a) previous modification   (b) proposed modification

Fig. 5 adaptively modification with less distortion

Figure 6 is the framework of our proposed method. Even though the prediction and embedding processing can be operated on the "embedded" set and the "referenced" set alternatively for several rounds in order to increase EC, without loss of generality, we only introduce the embedding and extraction procedure on the "embedded" set at the first round. The detail is listed in Algorithm 1.

Algorithm 1 Reversible data hiding for 3D mesh models with three-dimensional PEH modification

Reversible data embedding procedure

- **1**. All the coordinates of vertices are multiplied by 10^m and truncate the decimal part.
- **2**. According to the odd-even property of indices into the vertex list, all the vertices are divided into the "embedded" set and the "referenced" set. Then, we traverse the vertices on the "embedded" set in ascending order by their indices and use their neighboring vertices on the "referenced" set for the predicted position. At this processing, the vertices on the "embedded" set are used to embed data and the vertices on the "referenced" set keep unchanged for sake of extract data and restore cover losslessly.
- **3**. Calculate the difference between the real position and predicted position for all the vertices on the "embedded" set, we obtain the PE sequence. By combining every three adjacent PEs into one PET, we construct the 3D PEH with smaller entropy than one-dimensional PEH and then project 3D PEH into 1D space for scalar PET sequence which is suitable for RCC. The PET sequence acts as the host sequence X.
- **4**. For the host sequence distribution $P_X(x)$ and distortion measure matrix $D(x, y)$, with a given restriction, we estimate OTPM by a two-step strategy which we introduced above.
- **5**. Modify the host sequence X into the marked signal Y and optimal embed data through RCC according to OTPM.
- **6**. Modify the coordinates of vertices on the "embedded" set by adding the marked sequence to their predicted position, then divide all the vertices by 10^m.
- **7**. Refreshing the vertex coordinate for all the vertices of 3D mesh model and then get the marked model.

  Data extraction and recovery procedure
- **1**. All the coordinates of vertices are multiplied by 10^m.
- **2**. The same as the embedding process, use the neighboring vertices on the "referenced" set to predict the vertices on the "embedded" set.
- **3**. Calculate the difference between the real position and predicted position for all the vertices on the "embedded" set, by this we can obtain the marked PE sequence. Then combine every three adjacent PEs into one PET and project 3D PEH into 1D PET sequence, which acts as the marked sequence Y.
- **4**. Estimate optimal transition probability matrix $P_{X|Y}(x|y)$ according to $P_Y(y)$, $D(x, y)$ and given restriction.
- **5**. Extract data and recover the host sequence X by inverse recursive construction coding according to OTPM.
- **6**. Recover coordinates of vertices on the "embedded" set by adding the host sequence X to their prediction values.
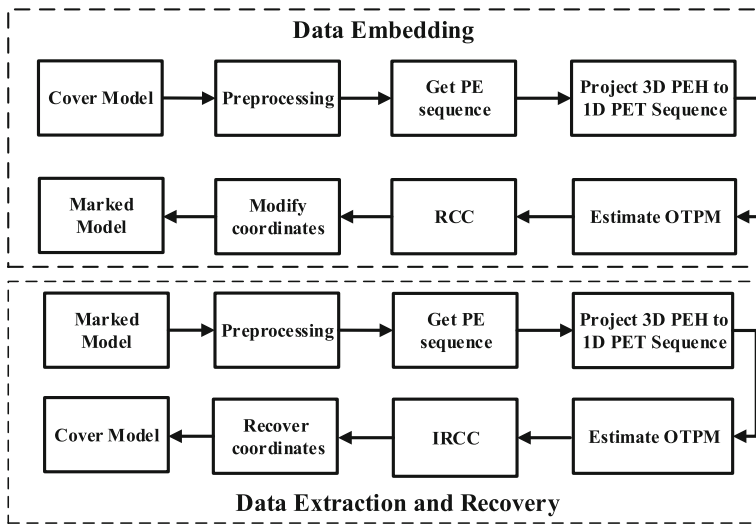- **7**. Restore the 3D mesh model by refreshing all the vertex coordinates.

Fig. 6 Procedure of our proposed RDH algorithm

4 Experimental result

Our proposed method processes 3D model and operates reversible data hiding with the experimental environment of MATLAB R2014 and OpenGL under windows 7, and we download 3D mesh models with .ply format and .obj format from Stanford University library, including "Bunny", "Dragon", "Armadillo", "horse", "lion", "hand", "happy" and so on. Compared to the algorithm proposed by Wu in [21] and Huang in [6], by embedding randomly generated 0/1 bit strings within the same models and setting the same parameter, we want to show that our algorithm is superior to the others.

Firstly, we present the experimental results of our proposed algorithm. Table 2 lists the SNR values by different embedding rates for kinds of 3D models, here we use bits per vertex (bpv) to measure the embedding capacity which equals to the length of embedded bits dividing the amounts of vertex. There are two things needed to be mentioned:

Table 2 Model information and SNR value with different embedding capacity for kinds of 3D models

Model	N_V	N_F	C=0.14	C=0.28	C=0.42	C=0.56	C=0.70
Bunny	35947	69451	57.48	55.65	54.35	52.77	51.26
Horse	48485	96966	58.63	57.56	56.60	55.59	54.61
Armadillo	172974	345944	59.43	58.04	56.94	55.92	54.97
Hand	327323	654666	69.75	67.20	65.30	63.36	61.50
Dragon	437645	871414	42.17	41.50	-	-	-

N_V represents for number of vertices, N_F for number of faces, **C** for embedding rate, and the unit of 0.14, 0.28, 0,42, 0.56, 0.70 is bpv.
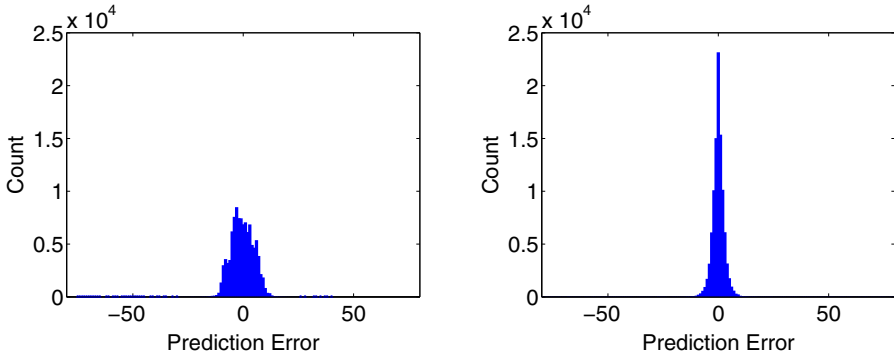
(a) PEH got by Wu *et al*'s method.   (b) PEH got by the proposed method.

Fig. 7 Prediction-Error Histogram for "bunny" got by Wu's method and ours

– The results on the table is derived from the first round embedding procedure. We can increase the embedding capacity by several rounds if the distortion is accepted.
– For one round, the embedding performance for different models is not the same, as shown in Fig. 8. Especially for "Hand" and "Dragon", the performance is totally different.

We firstly compare the mesh traversing strategy analogy with the raster-scan order for digital images in [21], the PEH got by Wu et al.'s method is listed in Fig. 7a, and that of the proposed method is listed in Fig. 7b. Obviously, the PEH in our proposed method is more centered distributed, which means it will achieve better performance for reversible data hiding, because it has a smaller entropy.

We define the complexity of models according to the 3D object shape and surface roughness, "Dragon" belongs to complex models and "Hand" belongs to simple models. It seems
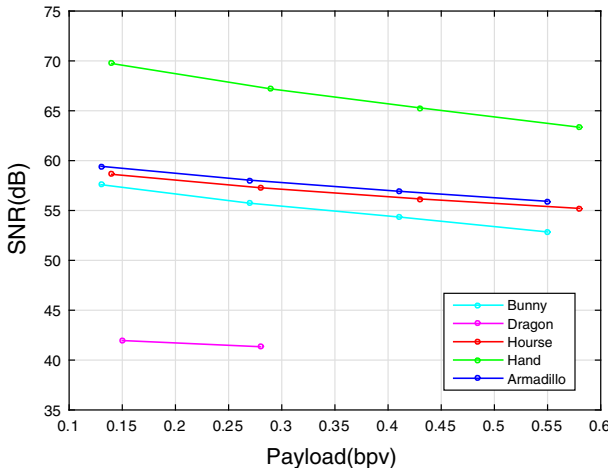

Fig. 8 PSNR-Payload curve for "horse" model with 48485 vertices and 96966 faces

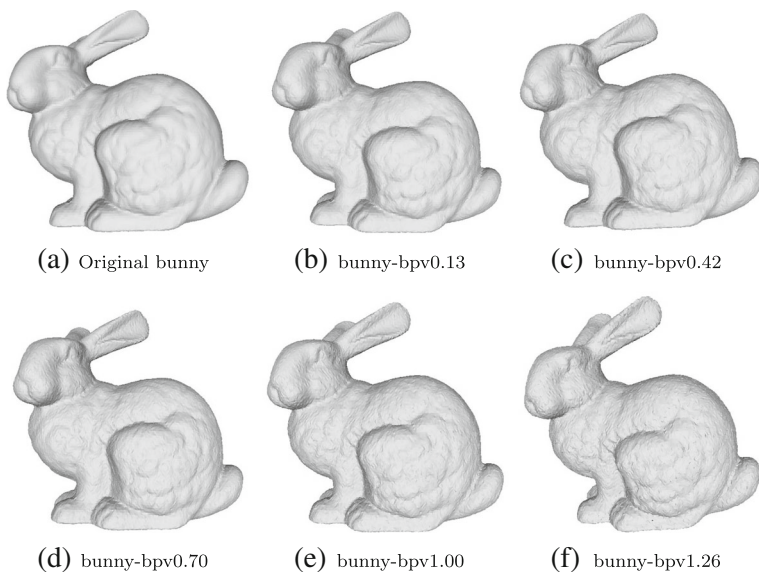**Table 3** Capacity and *SNR* results for "bunny" and "horse"

Bunny	C	0.13	0.27	0.42	0.55	0.70	0.84	1.00	1.12	1.26	1.41
	SNR	57.59	55.73	54.35	52.86	51.26	49.83	48.30	46.93	45.63	44.35
Horse	C	0.14	0.29	0.44	0.58	0.73	0.88	1.02	1.17	1.32	1.47
	SNR	58.63	57.28	56.16	55.21	54.34	53.39	52.27	51.19	50.22	49.21

C represents for embedding rate (bpv).

that the complex model is weaker for data embedding compared to the simple model, probably because the complex surface makes the prediction worse, which leads to the PEs within the threshold are limited (Fig. 8). We will analysis the influence caused by model shape and surface roughness on data embedding in our subsequence work.

Table 3 shows the SNR value with different embedding rate for "bunny" and "horse", and it proves that our algorithm can achieve a large embedding rate. Figures 9 and 10 show their visual effect with different embedding rate, if the embedded message is not too much, we can keep the distortion under the acceptable limit. Compared to the prediction-error expansion algorithm proposed by Wu et al. in [21], setting the same parameter, our proposed method is obviously better than theirs and the results for "bunny" and "horse" are shown on Fig. 11.

Compared to the histogram shifting algorithm which was described as the best of histogram-based algorithm for 3D models proposed by Huang in [6], we use Table 4 to support our benefits. Due to the reason that Huang et al. do not give the SNR value for model distortion in his paper, we just put emphasis on the maximal embedding capacity within the same distortion percentage. For "VenusHead","Dragon", "Armadillo" and "Golfball" which appeared in his paper, we give our results. From the comparison, we prove that our method can increase the embedding capacity with the same distortion. In Huang's method,

(a) Original bunny   (b) bunny-bpv0.13   (c) bunny-bpv0.42

(d) bunny-bpv0.70   (e) bunny-bpv1.00   (f) bunny-bpv1.26

Fig. 9 Visual effect for bunny with different embedding capacity

(a) Original horse   (b) horse-bpv0.29   (c) horse-bpv0.58

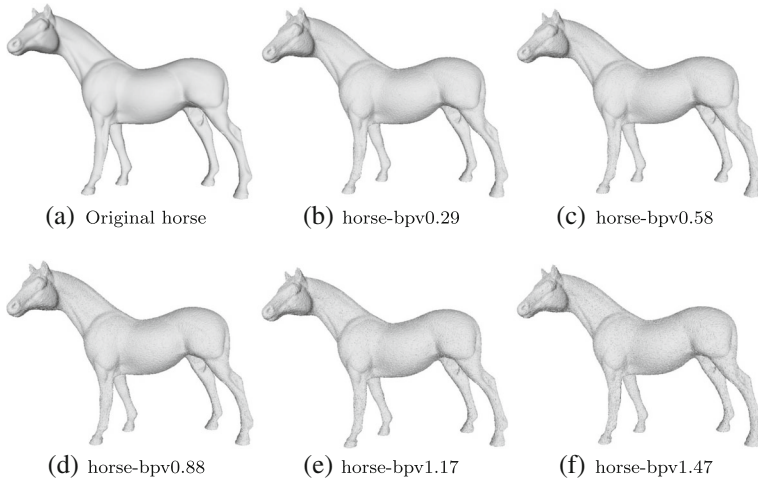(d) horse-bpv0.88   (e) horse-bpv1.17   (f) horse-bpv1.47

Fig. 10 Visual effect for horse with different embedding capacity

the histogram construction is based on the geometric similarity between neighboring vertices. But the geometric similarity they proposed is not able to generate a sharply distributed histogram. As a result the distortion caused by histogram shifting is large. Only by decreasing the embedding capacity can their method achieve a satisfactory visual effect of the stego model with the secret message embedded. Thus their embedding capacity is small.

In this paper, we proposed several improved steps in order to apply RCC for 3D mesh models and by this we hope to realize the optimal 3D mesh model RDH. From the results in the section, we proved that our algorithm can decrease the distortion which is measured by SNR for the same embedding rate, and increase the embedding capacity with the same embedding distortion. Thus, our algorithm is better than the state-of-the-art spatial-domain RDH algorithms for 3D models, and our superiority is mainly thanks to projecting the three-dimensional PEH into one-dimensional space by utilizing the correlation among PEs and defining suitable distortion matrix for RCC.
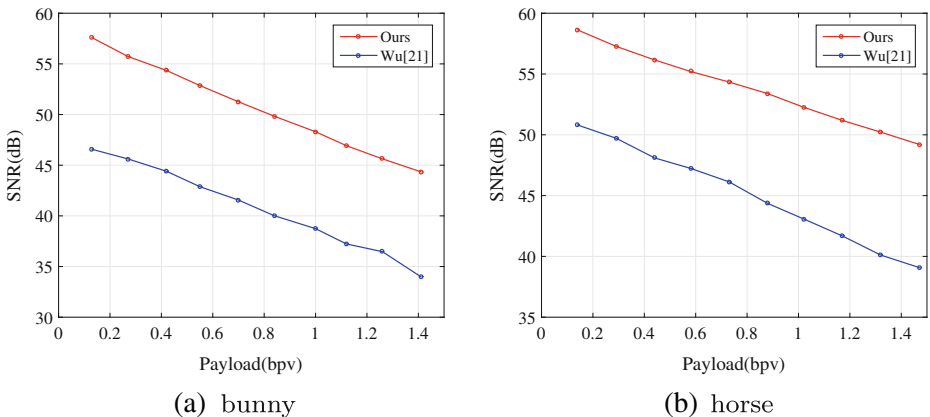

(a) bunny

(b) horse

Fig. 11 SNR-Capacity curve for "bunny" and "horse"

Table 4 The maximize embedding capacity (bpv) with the same distortion for Huang [6] and Ours

Model	N_V	N_F	Huang	Ours
VenusHead	327323	654666	0.093	0.580
Dragon	437645	871414	0.219	0.289
Armadillo	172974	345944	0.158	0.551
Golfball	122882	245760	0.514	0.776

5 Conclusion

In this paper, we propose 3D mesh model RDH with the optimal three-dimensional PEH modification using RCC. We firstly design a double-layered prediction scheme to obtain PEs with a shape histogram by utilizing the geometrical similarity among neighbouring vertices. Next we combine every three PEs into one PET in order to construct the 3D PEH with smaller entropy than 1D PEH by exploiting the correlation among PEs, and then project the 3D PEH into 1D space for scalar PET sequence which is suitable for RCC. And also, we define the distortion metrics for 3D models, by which we estimate the OTPM by two-step strategy. Finally, we modify the PET sequence and optimally embed data through RCC in order to achieve the optimal PEH modification. Without changing the topology and keep the vertices for prediction unchanged in the embedding process, we guarantee the reversibility of our method. As a result, we can operate the prediction and embedding procedure for several rounds in order to increase EC if the distortion is under the limit. Compared to two state-of-the-art spatial-domain RDH for 3D models, experimental results show that our proposed algorithm is significantly better than others.

The proposed framework shows its generalizing ability and good performance for 3D mesh model RDH. But as mentioned in our paper, how to design a better prediction scheme to maximally exploit the geometrical similarity of 3D models, how to greatly explore the correlation between PETs, and how to adaptively select the threshold for RDH based on the model shape and surface roughness, will be our future work.

References

1. Coltuc D (2012) Low distortion transform for reversible watermarking. IEEE Trans Image Process 21(1):412–417
2. Chou D, Jhou CY, Chu SC (2009) Reversible watermark for 3D vertices based on data hiding in mesh formation. Int J Innov Comput Inf Control 5(7):1893–1901
3. Fridrich J, Goljan M, Du R (2002) Lossless data embedding for all image formats. In: International Society for Optics and Photonics, pp 572-583
4. Hu Y, Lee HK, Li J (2009) DE-based reversible data hiding with improved overflow location map. IEEE Trans Circ Syst Vid Technol 19(2):250–260
5. Hu X, Zhang W, Hu X et al (2013) Fast estimation of optimal marked-signal distribution for reversible data hiding. IEEE Trans Inf Forensic Secur 8(5):779–788
6. Huang YH, Tsai YY (2015) A reversible data hiding scheme for 3D polygonal models based on histogram shifting with high embedding capacity. 3D Res 6(2):1–12
7. Kalker T, Willems FMJ (2003) Capacity bounds and constructions for reversible data-hiding. In: International Society for Optics and Photonics, Electronic Imaging, pp 604–611

8. Luo L, Chen Z, Chen M (2010) Reversible image watermarking using interpolation technique. IEEE Trans Inf Forensic Secur 5(1):187–193

9. Li X, Yang B, Zeng T (2011) Efficient reversible watermarking based on adaptive prediction-error expansion and pixel selection. IEEE Trans Image Process 20(12):3524–3533

10. Lin SJ, Chung WH (2012) The scalar scheme for reversible information-embedding in gray-scale signals: Capacity evaluation and code constructions. IEEE Trans Inf Forensic Secur 7(4):1155–1167

11. Lu ZM, Li Z (2007) High capacity reversible data hiding for 3D meshes in the PVQ domain. In: Digital Watermarking, Springer Berlin Heidelberg, pp 233–243

12. Luo H, Lu ZM, Pan JS (2006) A reversible data hiding scheme for 3D point cloud model. In: IEEE International Symposium on Signal Processing and Information Technology, pp 863–867

13. Luo H, Pan JS, Lu ZM (2006) Reversible data hiding for 3D point cloud model. In: IEEE International Conference on Intelligent Information Hiding and Multimedia Signal Processing(IIH-MSP), pp 487–490

14. Ni Z, Shi YQ, Ansari N (2006) Reversible data hiding. IEEE Trans Circ Syst Video Technol 16(3):354–362

15. Ou B, Li X, Zhao Y (2013) Pairwise prediction-error expansion for efficient reversible data hiding. IEEE Trans Image Process 22(12):5010–5021

16. Sachnev V, Kim HJ, Nam J (2009) Reversible watermarking algorithm using sorting and prediction. IEEE Trans Circ Syst Video Technol 19(7):989–999

17. Sun Z, Lu ZM, Li Z (2006) Reversible data hiding for 3D meshes in the PVQ-compressed domain. In: IEEE International Conference on Intelligent Information Hiding and Multimedia Signal Processing(IIH-MSP), pp 593–596

18. Tian J (2003) Reversible data embedding using a difference expansion. IEEE Trans Circ Syst Vid Technol 13(8):890–896

19. Thodi DM, Rodrguez JJ (2007) Expansion embedding techniques for reversible watermarking. IEEE Trans Image Process 16(3):721–730

20. Tsai P, Hu YC, Yeh HL (2009) Reversible image hiding scheme using predictive coding and histogram shifting. Signal Process 89(6):1129–1143

21. Wu H, Dugelay JL (2008) Reversible watermarking of 3D mesh models by prediction-error expansion. In: IEEE 10th Workshop on Multimedia Signal Processing, pp 797–802

22. Wu H, Cheung Y (2005) A reversible data hiding approach to mesh authenticationWeb Intelligence. In: The 2005 IEEE/WIC/ACM International Conference on Proceedings, pp 774–777

23. Zhang W, Chen B, Yu N (2012) Improving various reversible data hiding schemes via optimal codes for binary covers. IEEE Trans Image Process 21(6):2991–3003

24. Zhang W, Hu X, Li X et al (2013) Recursive histogram modification: establishing equivalency between reversible data hiding and lossless data compression. IEEE Trans Image Process 22(7):2775–2785

25. Zhang W, Hu X, Li X et al (2015) Optimal transition probability of reversible data hiding for general distortion metrics and its applications. IEEE Trans Image Process 24(1):294–304

Ruiqi Jiang received his B.S. degree in 2011 from Harbin Institute of Technology, M.S. degree in 2013 from The George Washington University (USA). He is currently pursuing the Ph.D. degree in electronic engineering in University of Science and Technology of China. His research interests including multimedia security and information hiding.

Weiming Zhang received his M.S. degree and Ph.D. degree in 2002 and 2005 respectively from the Zhengzhou Information Science and Technology Institute, Zhengzhou, China. Currently, he is an associate professor with the School of Information Science and Technology, University of Science and Technology of China. His research interests include information hiding and cryptography.

Dongdong Hou received his B.S. degree in electronic engineering from Hefei University of Technology in 2014. He is currently pursuing the M.S. degree in electronic engineering in University of Science and Technology of China. His research interests include image processing and information hiding.

Hui Wang received his B.S. degree in electronic engineering from Hefei University of Technology in 2014. He is currently pursuing the M.S. degree in electronic engineering in University of Science and Technology of China. His research interests include multimedia security and information hiding.

Nenghai Yu received his B.S. degree in 1987 from Nanjing University of Posts and Telecommunications, M.E. degree in 1992 from Tsinghua University and Ph.D. degree in 2004 from University of Science and Technology of China, where he is currently a professor. His research interests include multimedia security, multimedia information retrieval, video processing and information hiding.