Reversible Watermarking of 3D Mesh Models Using Prediction-error Expansion

Anping Zhu*, Caiming Zhang*†,Xingqiang Yang*,Xifeng Gao*
*School of Computer Science and Technology
Shandong University, Jinan, China, 250101
†School of Computer Science and Technology
Shandong Economic University, Jinan, China, 250014

***Abstract*—A reversible watermarking scheme for 3D meshes based on prediction-error expansion is proposed in this paper. In our algorithm, we embed the watermark by slightly changing the positions of some vertices in the mesh model. For a vertex to be altered for embedding, the centroid of its adjacent vertices is calculated as datum position which is kept unchanged during both embedding and extracting processes. The prediction-error, that is, half the distance from the centroid to the vertex position, is expanded for data embedding. One of the most notable features of our algorithm is that the distortion can be easily controlled in spite of the complex topology of the mesh models. Further more, the watermarked model is capable of resisting such attacks as translation, rotation and random noise. Experimental results demonstrate high capacity and low distortion of the proposed data hiding scheme.**

I. Introduction

Digital watermarking is a method of embedding useful information into a digital medium, such as an image, audio, or video for the purpose of copyright communication and protection, content authentication, forensic tracking, broadcast monitoring, etc. Different from encryption technology, digital watermarking technology hides confidential information into public information and transmits confidential information through the transmission of public information. It is difficult for potential detectors or illegal interceptors to verify the existence of the confidential information and intercept it so that the security is guaranteed. As for most watermarking methods, embedding some information will inevitably change the original data when the watermark is extracted, though it is usually unnoticeable. In some applications such as medical, military, and legal domains, even the imperceptible distortion is unacceptable. In order to satisfy these needs, reversible watermarking emerges, where the original data can be fully recovered when the information bits are extracted.

There have been plenty of researches of reversible watermarking for 2D still image, which can be classified into three categories in literature. Many of the early approaches to reversible watermarking can be categorized as modulo-arithmetic-based additive spread-spectrum techniques [1]–[3]. The second category of approaches involves methods to compress a set of selected features from an image and embed the payload in the space saved due to the compression, which often achieves higher embedding capacity [4]–[7]. The third category of high-capacity reversible data embedding algorithm is classified as expansion embedding approaches [8]–[10].Specially, Todi,et al [11], introduced histogram shifting technique to improve Tian's scheme and proposed a new reversible watermark method denoted as prediction-error expansion in which the unprocessed neighbors of a pixel are exploited for prediction and the difference between the predicted and real values is expanded for data embedding.

With the development of three-dimensional laser scanning and modeling technology, 3D models have been widespread used in various applications such as digital archives, visualization, entertainment and virtual reality. Digital watermarking, as a digital copyright protection tool, has been applied to 3D meshes and has made great progress [12]–[17]. At the same time, reversible watermarking for 3D meshes has drawn researchers' attention due to its specific application. In 2003, Dittmann and Benedens first presented reversible watermarking for 3D meshes [18], in which extra faces and vertices are added into the original mesh with the public verifiable digital signature protocol. In [19], Wu and Cheung proposed a fragile watermark method to authenticate 3D meshes by modulating the distance from mesh faces to the center achieving high capacity with very little distortion. Li et al [20] proposed a reversible data hiding approach for 3D mesh in the predictive VQ domain. Wu [21] proposed a reversible data hiding method for 3D meshes based on prediction-error expansion. Their algorithm has large capacity yet the distortion is serious.

In this paper, we propose a new reversible data hiding scheme for 3D meshes based on prediction-error method which is presented in[12]. We first predict a vertex by calculating the centroid of its neighbors, and then expand the prediction error, which is half the distance from the predicted position to the real position, to embed the watermark. Compared to the previous schemes, our method has advantages in low distortion and easy distortion control . After expanded, the new position is nearly coinciding with the original position, which achieves very low distortion. Further more, the watermark can be extracted when the mesh model is attacked by translation, rotation and random noise.

The rest of the paper is organized as follows. The proposed algorithm of reversible data hiding for 3D meshes is described in Section II and its performance is analyzed in Section III.Experimental results are shown in Section IV and the conclusion is drawn in Section V.

II. Reversible Watermarking Algorithm

A. Prediction-error Expansion

Prediction-error expansion embedding technique involves using a predictor, instead of a difference operator, to create the feature elements into which expansion embedding is done [11].

Let's consider an integer x into which a bit b is to be embedded. A predictor operates on a set of assigned integers which are connected to x and obtains $\bar{x}$ as its predicted value. The prediction-error is

$$p = x - \bar{x} \tag{1}$$

The bit b is embedded into x by expanding the prediction-error,

$$p' = 2p + b \tag{2}$$

$$x' = \bar{x} + p' = x + p + b \tag{3}$$

According to the formulas above, we can extract the bit b and restore the original integer x with the same predicted value $\bar{x}$.

$$b = p' - 2 \times \lfloor p'/2 \rfloor \tag{4}$$

$$x = x' - \lfloor p'/2 \rfloor - b \tag{5}$$

Prediction-error expansion combines the advantages of expansion embedding with the superior decorrelating abilities of a prediction, resulting in a higher data-embedding capacity than difference expansion. In digital images, since the range of pixel values is limited, not all the prediction-errors can be expanded to prevent overflow and underflow. However, in 3D mesh models, vertex coordinates are presented by floating point numbers and the representation is much more precise so that all the prediction-errors can be expanded for data embedding.

B. The Embedding Process

A mesh model consists of a number of vertices denoted by $V = \left\{v_i \in \Re^3 | 1 \le i \le N\right\}$ where a vertex position v_i specifies the coordinate (x_i, y_i, z_i) and N is the number of the vertices. Each vertex has an index and is connected to other vertices based on the topology. We define v_i and all of its adjacent vertices as a embedding unit denoted as EU_i, in which the coordinates of v_i will be modulated in order to embed the watermark bit.Figure 1 shows the structure of the embedding unit EU_i,in which v_i^j represents the adjacent vertex of v_i.

We traverse the vertices of the mesh model in ascending order of their indices to select the embedding units. Once a vertex appears in one embedding unit, it will be marked immediately and will never turn up in other embedding units. The vertex indexed by 0 is first traversed. Since none of its adjacent vertices has been traversed, the first traversed vertex and its adjacent vertices can be chosen for the embedding unit. All the vertices in the unit will be marked then . For the vertex to be traversed next, only if none of its adjacent vertices has been marked, it can be chosen for the embedding unit; otherwise, the traversal process will continue until all the vertices are traversed.

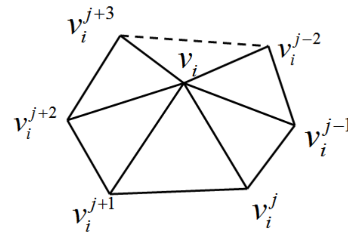


Fig. 1. The structure of the embedding unit EU_i

For the embedding unit EU_i, we calculate the centriod of the adjacent vertices to predict the real position of v_i,

$$c_i = \frac{1}{n_i} \sum_{j=1}^{n_i} v_i^j \tag{6}$$

in which v_i^j represents the adjacent vertex of v_i and n_i is the number of the adjacent vertices. We expand the distance value to embed the watermark and adjust the vertex position v_i along the connection v_i and c_i. The prediction-error, that is, half the distance from the predicted position and the real position is calculated via,

$$d_i = \frac{1}{2}\sqrt{(v_{ix} - c_{ix})^2 + (v_{iy} - c_{iy})^2 + (v_{iz} - c_{iz})^2} \tag{7}$$

where (v_{ix}, v_{iy}, v_{iz}) and (c_{ix}, c_{iy}, c_{iz}) are the coordinates of v_i and c_i respectively. Without loss of generality, we suppose the coordinates in the original mesh model are at precision lever of 10^{-n} (n is an integer).Then d_i is expanded for embedding according to (8)

$$\tilde{d_i} = d_i + int(d_i \times 10^m) \times 10^{-m} + b_i \times 10^{-m} \tag{8}$$

where m is an appointed integer and should be no higher than n. Finally the new position $\tilde{v}_i$ is calculated as the following,

$$\tilde{v} = c_i + \frac{\tilde{d}_i}{d_i}(v_i - c_i) \tag{9}$$

Now the watermark bit b_i has been embedded in the unit EU_i. When all the units are processed, the watermarked mesh model is obtained. From the embedding process we can see that the maximum capacity equals the total of all the embedding units which can be selected in the mesh model. Obviously, the less vertices are contained in a unit, the more independent units can be obtained. In fact, if we take two adjacent vertices as an embedding unit, the capacity can reach half the amount of vertices in the mesh; if we adopt the traversal tragedy proposed in [21], the capacity can achieve the amount of the vertices. In the practical applications, we can apply different traversal methods to gain a satisfying capacity.

C. The extracting Process

The mesh traversal in the recovery process is performed in the same order as in the embedding so that we can select the same embedding units. For the embedding unit EU_i ,we first calculate the distance from the predicted position c_i and the watermarked position $\tilde{v}_i$,

$$\tilde{d}_i = \sqrt{(\tilde{v}_{ix} - c_{ix})^2 + (\tilde{v}_{iy} - c_{iy})^2 + (\tilde{v}_{iz} - c_{iz})^2} \tag{10}$$

Then the original vertex position v_i and the embedded watermark bit b_i are obtained by the inverse transform of (8) and (9),

$$\begin{aligned} b_i &= (\tilde{d}_i \times 10^m)\%2 \\ d_i &= (\tilde{d}_i - int(\tilde{d}_i \times 10^m) \times 10^{-m} - b_i \times 10^{-m}) \times 2 \\ v_i &= c_i + \frac{d_i}{\tilde{d}_i}(\tilde{v}_i - c_i) \end{aligned} \tag{11}$$

The original mesh model is recovered and the watermark is extracted after all the embedding units are processed.

III. Distortion Analysis

We expand half the distance from the predicted position to the real position for embedding to improve the visual quality. For the unit EU_i, (v_{ix}, v_{iy}, v_{iz}) is the coordinates of the original position v_i and $(\tilde{v}_{ix}, \tilde{v}_{iy}, \tilde{v}_{iz})$ is the coordinates of the new position $\tilde{v}_i$ after embedding. Let ε denote the modulation of v_i and $\tilde{v}_i$.

$$\varepsilon = \sqrt{(\tilde{v}_{ix} - v_{ix})^2 + (\tilde{v}_{iy} - v_{iy})^2 + (\tilde{v}_{iz} - v_{iz})^2} \tag{12}$$

With the equation (8) and (9), (12) is equivalent to (13)

$$\varepsilon = |b_i \times 10^{-m} - (d_i - int(d_i \times 10^m) \times 10^{-m})| \tag{13}$$

Since $b_i \in \{0,1\}$,then $\varepsilon < 10^{-m}$ .That is, the distortion caused by embedding can be easily controlled by choosing an appropriate value for the parameter m .In fact, the distortion can not be distinguished by human visual system when the value of m is 3 .

In the literature, the Signal to Noise Ratio, SNR for short,is often used to measure the distortion of a mesh model. Here we take the definition of the SNR in [21] .

However, the ability to resist random noise attack decreases as the value of m increases. Weighing the gains and losses, we assign m to 4.

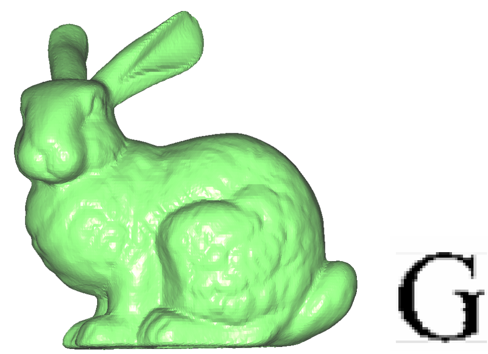
IV. Experiment Results

A. Invisibility test

In our experiment, the standard models of "Bunny" "Buddha""Dinosaur" "Venus" "cow" are chosen as test cases and the watermark is a randomly generated binary bits stream. We embed the same amount of watermark bits into each model by our and Wu's method. In addition, if there is no special instructions, the parameter value of m is 4. Results are shown in Table I. It can be easily seen, the invisibility of our algorithm is better than the algorithm in [21]. Figure 2 shows the original and the watermarked mesh models using our embedding algorithm and the algorithm proposed in [21].

TABLE I
Capacity and SNR values of models after embedding the watermark

model	Number of Vertics	Our proposed method		Wu's method	
		Capacity (bit)	SNR (db)	Capacity (bit)	SNR (db)
Bunny	35947	8853	89.18	8853	53.49
Buddha	32328	9277	86.51	9277	49.49
Dinosaur	11322	2758	85.64	2758	53.40
Venus	2838	696	86.87	696	46.30
cow	6475	1733	88.28	1733	39.00



(a) (b)

Fig. 3. (a)"Bunny"model(35947,69451). (b)The watermark

B. Robustness Analysis

In this paper, we take distance property as the information carrier, and our algorithm can resist such attacks as translation, rotation, reflection and random noise. For robustness test, we use the standard "Bunny" model with 35947 Vertices and 69451 faces. We transform the image shown in Figure 3(b) into binary bit stream, and use it as the watermark to embed into the model. In this experiment, we repeat embedding the watermark eight times in order to enhance the resistance to noise.

The random uniform noise is added to the coordinates of the vertices in the model. Suppose (x_i, y_i, z_i) is the coordinate of vertex v_i in the original mesh model and (x'_i, y'_i, z'_i) is the coordinate of vertex v'_i in the attacked mesh model after the noise added,then $x'_i = x_i + \Delta x_i, y'_i = y_i + \Delta y_i, z'_i = z_i + \Delta z_i$ while $(\Delta x_i, \Delta y_i, \Delta z_i)$ is a variable vector randomly distributed in the interval [-a, a] and a is the noise strength.

Experimental results show that the watermark information can be exactly extracted after attacks of rotation, translation and reflection shown in Figure 4. Besides, the watermark can be still displayed when part of the vertices in the model are attacked by random noise, as is shown in Figure 5,where different rows indicate that different amplitude of noise is added to the mesh and different columns indicate that different percentage of vertices in the mesh are attacked by the noise.

In the robust watermarking method, the BER, which stands for the ratio of the correct bits of watermark detected to the bits of watermark embedded, is usually adopted to evaluate robustness. Here, we also use the BER to measure the robust-

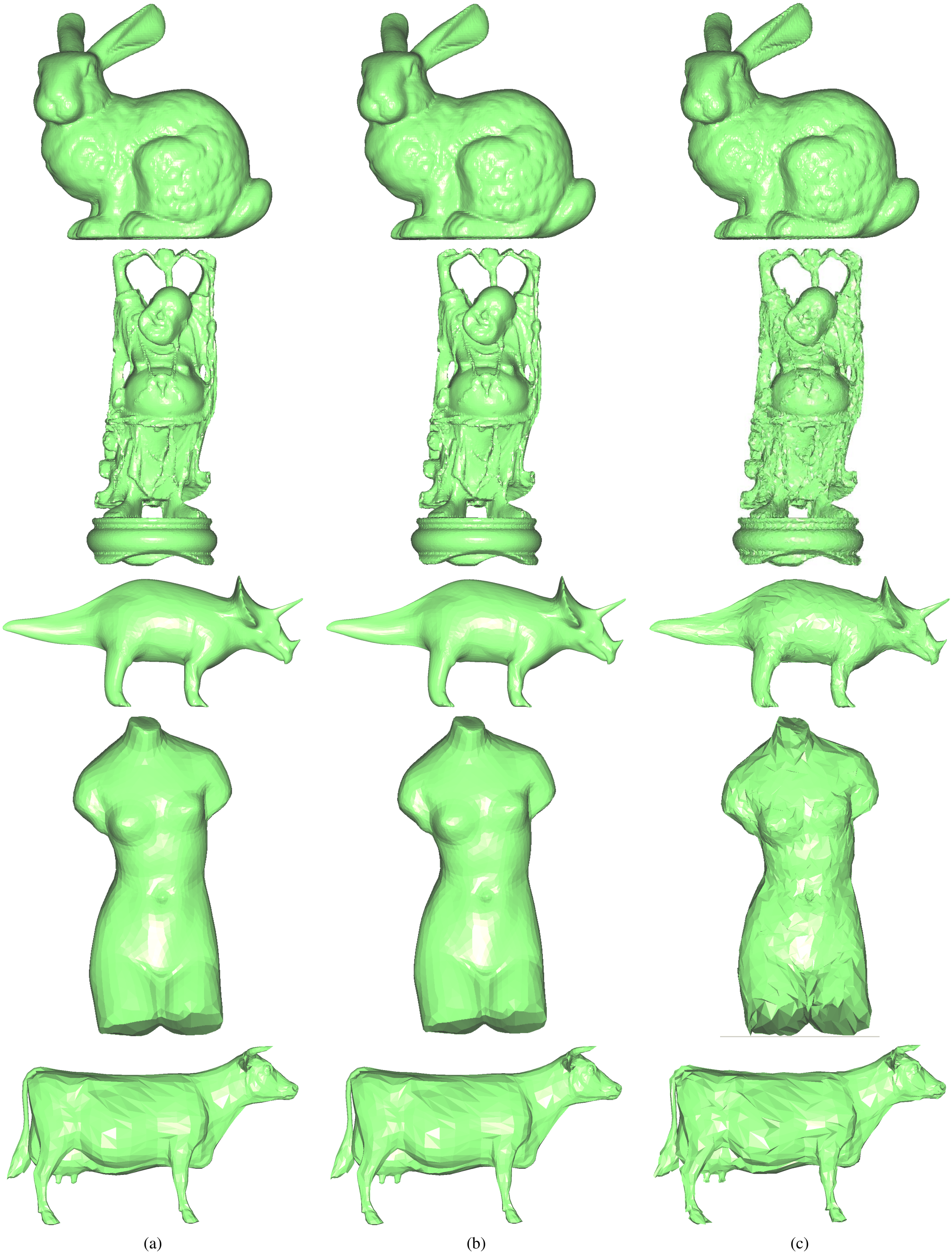


Fig. 2. (a)The original mesh models of ”Bunny”,”Buddha”,”Dinosaur” ”Venus”,”cow”.(b)The watermarked mesh models by our method.(c)The watermarked mesh models by Wu’s method.

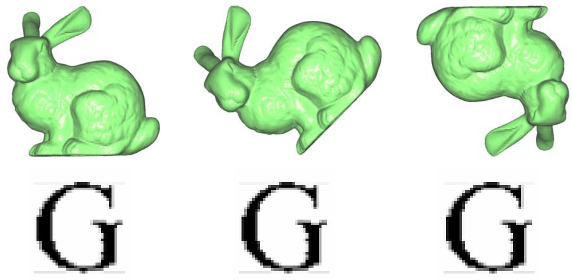


Fig. 4. The watermark extracted from the mesh attacked by translation, rotation and reflection respectively.

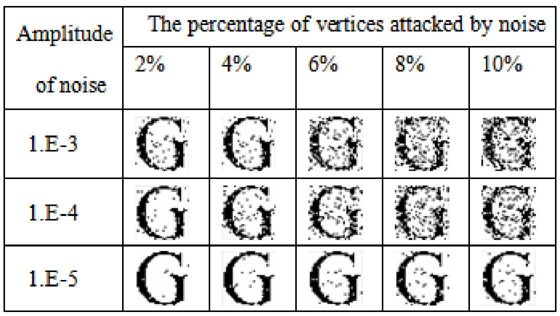


Fig. 5. The watermark extracted from the mesh attacked by noise

TABLE II
CAPACITY AND SNR VALUES OF MODELS AFTER EMBEDDING THE WATERMARK

Amplitude of noise	The percentage of vertices attacked by noise				
	2%	4%	6%	8%	10%
1.E-3	93.56%	87.31%	85.45%	79.59%	75.88%
1.E-4	95.51%	91.02%	86.82%	81.94%	81.35%
1.E-5	99.22%	98.34%	97.47%	95.51%	96.59%

ness of the proposed algorithm .In the test, we add the random uniform noise to part of the vertices in the mesh and record the BER under different amplitude of noise attack in Table II.

V. CONCLUSION

In this paper, we propose a new scheme of reversible watermark for 3D mesh models based on the prediction-error expansion technique. The watermark bits are embedded by slightly modulating the distance from a given vertex to the centroid of its adjacent vertices with the topology structure unchanged. Experimental results demonstrate high capacity and low distortion of the proposed algorithm, as well as the ability to resist such attacks as translation, rotation and random noise.

ACKNOWLEDGMENT

This work is supported by National 863 Plans Projects(2009 AA01Z304),the National Nature Science Foundation of China(60933008,60703081) and Key Science and Technology Project of Shandong Province (No. 2007gg10001004).

REFERENCES

[1] J. M. Barton, "Method and apparatus for wmbedding authentication information within digital data," *United States Patents*, no. 5,646,997, pp. 371–372, July 1997.

[2] C. W. Honsinger, P. W. Jones, M. Rabbani, and J. C. Stiffel, "Lossless recovery of an original image containing embedded data," *United States Patents*, no. 6,278,791, Aug. 2001.

[3] B. Macq, "Multiresolution transform for image authenticating watermarking," in *Proc. EUSIPCO '2000*, Sept. 2000, pp. 533–536.

[4] J. Fridrich, M. Goljan, and R. Duand, "Invertible authentication," in *Proc. Security and Watermarking of Multimedia Contents III*, vol. 4314, 2001, pp. 197–208.

[5] J. Fridrich, M.Goljan, and R.Du, "Invertible authentication watermark for jpeg images," in *Proceedings of International Conference on Information Technology*, April 2001, pp. 223–227.

[6] M. Goljan and J.Fridrich, "Distortion free data embedding," in *Information Hiding Workshop*, vol. 2137. New York: LNCS Spring-Verlag, 2001, pp. 27–41.

[7] J. Fridrich, M. Goljan, and R. Duand, "Lossless data embedding for all image formats," in *Proc. Security and Watermarking of Multimedia Contents IV*, vol. 4314, 2002, pp. 572–583.

[8] J.Tian, "Reversible data embedding using a difference expansion," *IEEE Transactions on Circuits and System for Video Tecnology*, vol. 13, no. 8, pp. 890–896, Aug 2003.

[9] A. M.Alattar, "Reversible watermark using the difference expansion of a generalized integer transform," *IEEE Transactions on Image Process*, vol. 13, no. 8, pp. 1147–1156, Aug 2004.

[10] L.Kamstra and H.Heijmans, "Reversible data embeedding into images using wavelet techniques and sorting," *IEEE Transactions on Image Process*, vol. 14, no. 12, pp. 2082–2090, Dec 2005.

[11] M.Thodi and J.J.Rodriguez, "Prediction-error based reversible watermarking," in *Proceedings of the IEEE International Conference on Image Processing*, vol. 3, Mar 2007, pp. 1549–1552.

[12] O.Benedens, "Geometry-based watermarking of 3-d polygonal models," *IEEE Computer Graphics and Applications*, vol. 19, no. 1, pp. 46–55, Jan 1999.

[13] R. Ohbuchi and H.Masuda, "Embedding data in 3d models," in *Proc.Intl Workshop Interactive Distributed Multimedia Systems and Telecommunication Services*, vol. 1309. Springer Berlin, Sept 1997, pp. 1–11.

[14] O.Benedens, "Robust watermarking and affine registration of 3d meshes," in *Information Hiding*, vol. 2578. Springer Berlin, Jan 2003, pp. 177–195.

[15] E.Praun, H. Hoppe, and A. Finkelstein, "Robust mesh watermarking," *SIGGRAPH*, pp. 49–56, 1999.

[16] A.Kalivas, A.Tefas, and I.Pitas, "Watermarking of 3d models using principal component analysis," in *2003 IEEE International Conference on Accoustics, Speech, and Signal Processing*, vol. 5, April 2003, pp. 676–679.

[17] S. Zafeiriou and A. Tefas, "Blind robust watermarking schemes for copyright protection of 3d mesh objects," *IEEE Transactions On Visualization and Computer Graphics*, vol. 11, no. 5, Sept 2005.

[18] J.Dittman and O.Benedens, "Invertible anthentication for 3d meshes," in *Security and Watermarking of Multimedia Contents*, vol. 5020, June 2003, pp. 653–664.

[19] H.T.Wu and Y.M.Cheung, "A reversible data hiding approach to meshes authentication," in *International Conference on Web Intelligence*, vol. 2005, Sept 2005, pp. 774–777.

[20] Z.M.Lu and Z.Li, "High capacity reversible data hiding for 3d meshes in the pvq domain," in *Proceedings of the 6th International Workshop on Digital Watermarking*, vol. 5041 LNCS, Dec 2007, pp. 593–596.

[21] H. tian Wu and J.-L. Dugelay, "Reversible watermarking of 3d mesh models by prediction-error expansion," in *Proceedings of the 2008 IEEE 10th Workshop on Multimedia Signal Processing, MMSP 2008*, Oct 2008, pp. 797–802.