

A high capacity combined reversible watermarking scheme for 2-D CAD engineering graphics

Di Xiao · Shulei Hu · Hongying Zheng

Published online: 31 October 2013
© Springer Science+Business Media New York 2013

Abstract Compared with the wide research on reversible watermarking scheme for raster image, the research on its counterpart for 2-D CAD engineering graphics lags far behind. In this paper, based on improved quantization index modulation (IQIM) and improved difference expansion (IDE), a combined reversible watermarking scheme is proposed for 2-D CAD engineering graphics. The proposed scheme can solve the embedding-limitation problem existing in IQIM technique and increase the watermark embedding capacity greatly. Theoretical analysis and experimental results also show that the proposed scheme has good imperceptibility and robustness.

Keywords Reversible watermarking · 2-D CAD engineering graphics · Improved quantization index modulation (IQIM) · Improved difference expansion (IDE) · Embedding capacity

1 Introduction

For traditional watermarking schemes, to accomplish watermark embedding in the host data, certain distortion of the original information will inevitably occur. However, such a strategy is not suitable for some special applications like medical, judicature, and military ones, where the data integrity demands are quite strict, and no distortion is allowed. To solve this problem, the concept of reversible watermark was proposed [4, 5, 7]. In recent years, research on reversible data-hiding methods was mainly focused on raster images [1–3, 6, 15, 16, 19, 20], while only a few lossless watermarking techniques had been developed for vector graphics. Voigt et al. [17] first proposed the algorithm of reversible data hiding in vector maps. They embedded watermark information by modifying the 8-point integer DCT coefficients of the map coordinates. However, its distortion controlling mechanism was complex and the payload was low. Another reversible watermarking method for vector graphics based on difference expansion was proposed in [18]. The scheme hid data by modifying the difference between the adjacent coordinates or Manhattan distances, which achieved high capacity. For 3D mesh models, Ji et al. [8] presented a reversible watermarking method based on ratio expansion. The distances between a vertex and its adjacent vertices were dealt with and then expanded to embed watermarks. Zhu proposed a reversible watermarking scheme for 3D meshes based on

D. Xiao (✉) · S. Hu · H. Zheng
College of Computer Science, Chongqing University, Chongqing 400044, China
e-mail: xiaodi_cqu@hotmail.com

prediction-error expansion in [21]. The watermark was embedded by slightly modulating the distance from a given vertex to the centroid of its adjacent vertices with the topology structure unchanged. A new reversible 3D mesh watermarking scheme was proposed in conjunction with progressive compression in [10]. Each level of detail was compressed and watermarked by modifying the geometry of refined vertices with respect to the center of mass of the original 3D mesh. The proposed scheme was robust against channel noise and intentional attacks while having slightly additional compression rate cost.

2-D CAD engineering graphics is a kind of vector graphics, which has been widely applied in mechanics, architecture, and costume design, etc. Recently, since the illegal copying and distribution of engineering graphics occur frequently, research on watermarking for 2-D CAD engineering graphics has been continuously increasing. The digital watermarking method for CAD drawings was firstly proposed in [9]. The scheme embedded watermarks using an adaptive algorithm related to the length of the line, radius of the circle, and angle of the arc, which was robust against various attacks. In [11], a watermarking scheme was presented based on k-means for CAD design drawings. The proposed technique embedded the watermark into the geometric distribution of POLYLINE, ARC and 3DFACE objects in the main layers, which was tolerant to file format conversion, layer deletion, and other geometric attacks. A semi-fragile digital watermarking scheme for 2-D CAD engineering graphics based on log-polar coordinate mapping was proposed in [12] by Peng et al. The algorithm divided the vertices into groups, and embedded the watermark in the mantissa of the real-valued log-polar coordinates via bit substitution. It was robust against incidental global operations, and could also detect and locate malicious attacks.

Nevertheless, little work has been done on reversible watermarking for 2-D CAD engineering graphics. As a representative example, a reversible watermarking scheme for 2-D CAD engineering graphics based on IQIM was proposed in [13]. The technique embedded the watermark into the relative amplitudes or the relative phases of entity vertices and recovers data with a secret key, which could achieve a good balance among imperceptibility, robustness and security. A drawback of this scheme was that not every vertex could be embeddable. Another representative reversible watermarking scheme for 2-D CAD engineering graphics based on IDE was presented in [14]. It embedded the watermark in the ratio data of the distance between two vertexes, which ensured that every vertex could embed at least 1-bit watermark and was robust against typical operations such as translation, scaling and rotation. In this paper, a combined reversible watermarking scheme for 2-D CAD engineering graphics is proposed to overcome the embedding-limitation problem existing in [13] and increase the watermark capacity of the schemes in [13, 14] as well as keep other merits, such as good imperceptibility, security and robustness.

The remaining sections are organized as follows. The two above-mentioned hiding techniques are reviewed briefly in Section 2. In Section 3, the combined technique is proposed. Section 4 gives out the calculating process of some important parameters and the reversibility proof of the proposed technique. A detailed description of the proposed scheme for 2-D CAD engineering graphics is presented in Section 5. The experimental results and analysis are displayed in Section 6. Finally, the conclusions are summarized in Section 7.

2 Preliminaries

In this section, the concept of data precision of 2D CAD engineering graphic is firstly defined along with the measurement of reversibility for CAD graphics, and then the two watermarking methods which are used in our combined scheme are reviewed briefly.

2-D CAD engineering graphic consists of different entities such as points, lines and circles. Each entity has at least one two-dimensional vertex whose coordinates are real numbers. In most application cases, high precision of the graphics is one of the most important requirements. While there is a precision limitation of data stored in computer, the coordinate values are usually required to have a precision of 6–8 decimal digits in practice. As long as the loss between the original graphic and the recovered graphic is within this precision, the watermark scheme will have no influence on the application of 2-D CAD engineering graphics and can be regarded as reversible.

2.1 The improved quantization index modulation technique (IQIM) [13]

2.1.1 Watermark embedding

Given a quantization coefficient f (a real number), the embedding process is as follows.

Step 1: Calculate m, r according to

$$m = \left\lfloor \frac{f}{2^b \Delta} \right\rfloor, \quad r = f - 2^b m \Delta, \quad (1)$$

where m (a positive integer) is the ordinal number of the quantization interval, b (a positive integer) is an embedding strength, Δ (a small real) is the quantization step, and r (a nonnegative real and $r \in [0, 2^b \times \Delta)$) is the relative distance of f in the m^{th} quantization interval.

Step 2: Embed the watermark $w \in \{0, 1, \dots, 2^b - 1\}$ in the quantization coefficient f according to

$$f^w = 2^b m \Delta + w \Delta + \frac{r}{2^b}, \quad (2)$$

where f^w is the watermarked quantization coefficient.

2.1.2 Watermark extraction and data recovery

The process of watermark extraction and data recovery is as follows.

Step 1: Calculate the ordinal number of the quantization cell n and its relative distance d according to

$$n = \left\lfloor \frac{f^w}{\Delta} \right\rfloor, \quad d = f^w - n \Delta. \quad (3)$$

Step 2: Extract the watermark w^* according to

$$w^* = n - 2^b \left\lfloor \frac{n}{2^b} \right\rfloor. \quad (4)$$

Step 3: Recover the original real coefficient f^* according to

$$f^* = 2^b \left\lfloor \frac{n}{2^b} \right\rfloor \Delta + 2^b d. \quad (5)$$

For example, given $f=4.23246$, $\Delta=0.001$, $b=1$, $w=1$, the parameters can be calculated as follows.

Watermark embedding:

$$m = \left\lfloor \frac{f}{2^b \Delta} \right\rfloor = \left\lfloor \frac{4.23246}{2 \times 0.001} \right\rfloor = 2116$$

$$r = f - 2^b m \Delta = 4.23246 - 2 \times 2116 \times 0.001 = 0.00046$$

$$f^w = 2^b m \Delta + w \Delta + \frac{r}{2^b} = 2 \times 2116 \times 0.001 + 1 \times 0.001 + \frac{0.00046}{2} = 4.23323.$$

Watermark extraction:

$$n = \left\lfloor \frac{f^w}{\Delta} \right\rfloor = \left\lfloor \frac{4.23323}{0.001} \right\rfloor = 4233$$

$$d = f^w - n \Delta = 4.23323 - 4233 \times 0.001 = 0.00023$$

$$w^* = n - 2^b \left\lfloor \frac{n}{2^b} \right\rfloor = 4233 - 2 \times \left\lfloor \frac{4233}{2} \right\rfloor = 1$$

Data recovery:

$$f^* = 2^b \left\lfloor \frac{n}{2^b} \right\rfloor \Delta + 2^b d = 2 \times \left\lfloor \frac{4233}{2} \right\rfloor \times 0.001 + 2 \times 0.00023 = 4.23246.$$

2.2 The improved difference expansion technique (IDE) [14]

2.2.1 Watermark embedding

Given a real number D , an embedding position p (a negative integer which indicates the number of digits after the decimal point), an embedding strength s which may be different from the above embedding strength b in Section 2.1, the embedding process is as follows.

Step 1: According to the embedding position p , regulate the position of the decimal point of D , and perform a s -bit right-shift to D .

$$D_1 = (D \times 10^{-p}) / 2^s. \tag{6}$$

Step 2: Obtain the integer part I and the decimal part F of D_1 according to

$$D_1 = I + F, \tag{7}$$

$$I = LTrim(D_1), \quad F = RTrim(D_1). \tag{8}$$

Here, $LTrim(A)$ represents a function to trim the integer part of A and $RTrim(A)$ represents a function to trim the fractional part of A .

Step 3: Embed the watermark $w \in \{0, 1, \dots, 2^s - 1\}$ according to

$$I^w = I \times 2^s + w \times \text{sign}(D), \quad (9)$$

where $\text{sign}(D)$ is the sign function of D .

Step 4: Calculate the watermarked data D^w according to

$$D^w = (I^w + F) \times 10^p. \quad (10)$$

2.2.2 Watermarking extraction and data recovery

Step 1: Acquire the watermarked data D^w . Regulate the position of the decimal point of the host data according to

$$D_1^w = D^w \times 10^{-p}. \quad (11)$$

Step 2: Calculate the integer part I_1^w and the decimal part F_1^w of D_1^w , then extract watermark w^* according to

$$D_1^w = I_1^w + F_1^w, \quad (12)$$

$$w^* = I_1^w \% 2^s \times \text{sign}(D_1^w). \quad (13)$$

Step 3: Recover the host data according to

$$D^* = (\lfloor I_1^w \rfloor \times 2^{-s} \times \text{sign}(I_1^w) + F_1^w) \times 2^s \times 10^p. \quad (14)$$

For example, given $D=100.32412$, $s=1$, $p=-4$, $w=1$, the parameters can be calculated as follows.

Watermark embedding:

$$D_1 = (D \times 10^{-p}) / 2^s = (100.32412 \times 10^4) / 2 = 501620.6$$

$$I = LTrim(D_1) = 501620, \quad F = RTrim(D_1) = 0.6$$

$$I^w = I \times 2^s + w \times \text{sign}(D) = 501620 \times 2 + 1 = 1003241$$

$$D^w = (I^w + F) \times 10^p = (1003241 + 0.6) \times 10^{-4} = 100.32416.$$

Watermark extraction:

$$D_1^w = D^w \times 10^{-p} = 100.32416 \times 10^4 = 1003241.6$$

$$I_1^w = 1003241, F_1^w = 0.6$$

$$w^* = I_1^w \% 2^s \times \text{sign}(D_1^w) = 1003241 \% 2 \times 1 = 1.$$

Data recovery:

$$D^* = (\lfloor |I_1^w| \times 2^{-s} \rfloor \times \text{sign}(I_1^w) + F_1^w) \times 2^s \times 10^p = (\lfloor 1003241 \times 2^{-1} \rfloor \times 1 + 0.6) \times 2 \times 10^4 = 100.32412.$$

3 The combined IQIM and IDE technique

Given an original quantization coefficient f (in the m^{th} quantization interval), there may be a rounding error in the watermarked quantization coefficient f^w because of the storage precision limitation in computer, which may cause f^w to fall in the $(m+1)^{th}$ quantization interval. Since f^w is not in the same quantization interval as f , errors may occur at watermark extraction and data recovery stages. In order to reversibly extract the watermark and recover the host data, Reference [13] gives the following condition which must be satisfied

$$5 \cdot 2^b \cdot 10^{-(q+1)} \leq r < 2^b (\Delta - 5 \cdot 10^{-(q+1)}). \tag{15}$$

where q (a positive integer) is the data storage precision, and the other symbols are defined as Section 2.1.

Because of the above condition, not all quantization coefficients f (responding to the coordinate values in graphic) can be embeddable in the scheme. Therefore, to extract the embedded data and recover the original values, one needs to know which coordinate values have been selected for the embedding watermarks. Without the knowledge, their scheme may not work. However, such problem has not been considered in [13]. Here, we propose an improved watermark scheme with a combination of the two above methods, in which every vertex can be guaranteed to be embeddable. Furthermore, our scheme has greatly increased the embedding capacity of [13, 14].

3.1 Watermark embedding

Given a real number X as host data, two embedding strength b and s , a quantization step Δ , the embedding process is as follows(all the same symbols are defined as above).

Step 1: Calculate the ordinal number of the quantization interval m and the relative distance r of X in the m^{th} quantization interval as described in IQIM technique.

$$m = \left\lfloor \frac{X}{2^b \Delta} \right\rfloor, r = X - 2^b m \Delta. \tag{16}$$

Step 2: Take the relative distance r as the real number D defined in Section 2.2.1 and embed a watermark w_1 ($w_1 \in \{0, 1, \dots, 2^s - 1\}$) in r using IDE technique, then obtain the watermarked relative distance r^w (corresponding to D^w in Section 2.2.1). This is the first embedding process.

Step 3: Embed a watermark w_2 ($w_2 \in \{0, 1, \dots, 2^b - 1\}$) in the quantization coefficient X for the second embedding process according to

$$X^w = 2^b m \Delta + w_2 \Delta + \frac{r^w + k}{t}. \quad (17)$$

where X^w is the watermarked quantization coefficient, t and k are two positive real constants. The values of t and k will be given in Section 4.1.

3.2 Watermark extraction and data recovery

The process of watermark extraction and data recovery is as follows.

Step 1: Obtain the watermarked data X^w , and calculate the ordinal number of the quantization cell n and its relative distance d according to

$$n = \left\lfloor \frac{X^w}{\Delta} \right\rfloor, \quad d = X^w - n \Delta. \quad (18)$$

Step 2: w_2 can be extracted according to

$$w_2 = n - 2^b \left\lfloor \frac{n}{2^b} \right\rfloor, \quad w_2 \in \{0, 1, \dots, 2^b - 1\}. \quad (19)$$

Step 3: Calculate the watermarked relative distance r^{w*} (corresponding to r^w in Step 2 of the above embedding process) according to

$$r^{w*} = d \times t - k. \quad (20)$$

Step 4: Take r^{w*} as the watermarked data D^w defined in Section 2.2.1 and extract the watermark w_1 using IDE technique, then recover r^{w*} to r^* (corresponding to r in Step 1 of the above embedding process).

Step 5: Recover the host data X^* according to

$$X^* = 2^b \left\lfloor \frac{n}{2^b} \right\rfloor \Delta + r^*. \quad (21)$$

The proof of the reversibility is described in Section 4.2.

4 The requirements and proof of reversibility

4.1 Requirements of reversibility

Given the data precision g , an embedding strength b , a quantization step Δ and the original quantization coefficient X in the m^{th} quantization interval, the watermarked quantization

coefficient X^w , to guarantee the requirement of reversibility, Reference [13] gives the following conditions

$$2^b m \Delta - 5 \cdot 10^{-(q+1)} \leq X^w < 2^b (m + 1) \Delta - 5 \cdot 10^{-(q+1)}. \tag{22}$$

Combined with Eq. (17), we obtain

$$0 \leq w_2 \Delta + \frac{r^w + k}{t} < 2^b \Delta - 5 \cdot 10^{-(q+1)}. \tag{23}$$

Thus, we obtain

$$5 \cdot t \cdot 10^{-(q+1)} - k \leq r^w < t \left(\Delta - 5 \cdot 10^{-(q+1)} \right) - k. \tag{24}$$

From the above inequality, we finally manage

$$k > k_0, \quad \frac{2^b \Delta + (2^s - 1) \times 10^p + k}{\Delta - 5 \cdot 10^{-(q+1)}} < t \leq \frac{(1 - 2^s) \times 10^p + k}{5 \cdot 10^{-(q+1)}}. \tag{25}$$

where $k_0 = \frac{5 \Delta 2^b \cdot 10^{-(q+1)} + (2^s - 1) \Delta \cdot 10^p}{\Delta - 10^{-q}}$.

The detailed derivation process for Inequality (25) is provided in [Appendix](#).

When the values of k and t satisfy the requirement, regardless of the rounding error every vertex can be guaranteed to be embeddable in our scheme.

4.2 Proof of reversibility

According to the above Inequality (24), it can be deduced that $0 < \frac{r^w + k}{t} < \Delta$.

According to Eq. (18), we have

$$n = \left\lfloor \frac{X^w}{\Delta} \right\rfloor = \left\lfloor 2^b m \Delta + w_2 \Delta + \frac{r^w + k}{t} \right\rfloor = 2^b m + w_2 \tag{26}$$

$$d = f^w - n \Delta = \frac{r^w + k}{t} \tag{27}$$

According to Eqs. (20) and (27), $r^{w*} = d \times t - k = r^w$. As the extraction process of Step 4 in Section 3.2 has been proved to be reversible in Reference [14], the watermarked relative distance r^{w*} can be recovered exactly as the same as the original relative distance r ; that is $r^* = r$.

According to Eq. (26), we can obtain $\left\lfloor \frac{n}{2^b} \right\rfloor = \left\lfloor m + \frac{w_2}{2^b} \right\rfloor$. For $w_2 < 2^b$, then $\left\lfloor \frac{n}{2^b} \right\rfloor = m$.

Finally, we have $X^* = 2^b \left\lfloor \frac{n}{2^b} \right\rfloor \Delta + r^* = 2^b m \Delta + r = X$.

The proof is completed.

5 Reversible watermark for 2-D CAD engineering graphics

5.1 Watermark embedding

For a 2-D engineering graphics G , assuming all vertices can be denoted as $V = \{V_1, V_2, \dots, V_i, \dots, V_N\}$ (N is the total number of vertices in G), where $V_i(V_{ix}, V_{iy})$ represents a vertex at coordinates

(V_{ix}, V_{iy}) . Given a random binary sequence $W = \{w_1, w_2, \dots, w_n\}$ (n is the total number of bits) as watermark information, the embedding process is described as follows.

- Step 1: Choose two reference vertices V_p and V_q from V with a secret key K , and calculate the distance between the two vertices, $u = \sqrt{(V_{px} - V_{qx})^2 + (V_{py} - V_{qy})^2}$.
- Step 2: Compute the distance between any vertex V_i and $V_p (i \neq p, q)$, i.e., $s_i = \sqrt{(V_{ix} - V_{px})^2 + (V_{iy} - V_{py})^2}$, then calculate the ratio $R_i = s_i / u$. In such a way, a ratio set can be obtained from V , i.e., $R = \{R_1, R_2, \dots, R_M\}$, where $M = N - 2$.
- Step 3: Use the combined IQIM and IDE hiding technique described in Section 3.1 to embed watermark W into the ratio set R , and obtain the watermarked ratio set R^w .
- Step 4: Given the ratio set $R^w = \{R_1^w, R_2^w, \dots, R_M^w\}$. The watermarked coordinates of the 2-D vertex can be calculated according to

$$\begin{cases} V_{ix}^w = V_{px} + \frac{s_i^w}{s_i} (V_{ix} - V_{px}) \\ V_{iy}^w = V_{py} + \frac{s_i^w}{s_i} (V_{iy} - V_{py}) \end{cases}, \tag{28}$$

where $s_i^w = R_i^w \times u$ ($i \neq p, q$).

Replace the original data with the watermarked coordinates and obtain the watermarked 2-D CAD engineering graphics G^w .

5.2 Watermark extraction and data recovery

The process of watermark extraction and data recovery is as follows.

- Step 1: Get all the vertices of the watermarked 2-D engineering graphics G^w and select the two reference vertices V_p and V_q with the key K , then calculate the ratio data set $R^w = \{R_1^w, R_2^w, \dots, R_M^w\}$.
- Step 2: Use the combined IQIM and IDE hiding technique described in Section 3.2 to extract watermark information W^* .
- Step 3: Calculate the correlation coefficient NC between W^* and the original watermark W , i.e.,

$$NC = \frac{\sum_{i=1}^n w_i \times w_i^*}{\sqrt{\sum_{i=1}^n (w_i)^2} \times \sqrt{\sum_{i=1}^n (w_i^*)^2}}. \tag{29}$$

- Step 4: If the value of NC is equal to 1.0, use the combined IQIM and IDE embedding technique described in Section 3.2 to recover the original ratio set $R = \{R_1, R_2, \dots, R_M\}$.
- Step 5: According to Eq. (28), calculate the original vertex set V , and recover the original 2-D CAD engineering graphics G .

6 Experimental results and analysis

The experiments in this section are carried out on a PC with CPU P4 2.0 GHz, RAM 1 GB, Win7 Professional, AutoCAD 2007 and Visual C++ 6.0. Fifty different 2-D CAD engineering graphics (in DWG format) are used.

In the experiments, the parameters are chosen as follows: embedding strength $s=b=1$, storage precision $q=6$, quantization step $\Delta=0.001$, embedding position $p=-4$, watermark sequence $W=$ “Chongqing University”. With the above values, we can calculate and choose that $k=1.02 \times 10^{-4}$, $t=2.5$ according to the analyses in Section 4.1.

Here, the root mean square error (*RMSE*) is exploited to measure the objective quality of the watermarked graphics,

$$RMSE = \frac{1}{N} \|V - V^w\| \tag{30}$$

where V and V^w are the corresponding vertices in the original 2-D CAD engineering graphics and its watermarked counterpart, and N is the total number of vertices.

6.1 Analysis of reversibility

In the experiment, 50 different 2-D CAD engineering graphics are used to check the *RMSE* between the original graphics and the recovered graphics. The results are listed in Table 1.

As can be seen from Table 1, the average *RMSE* between the original graphics and the recovered graphics is about 3.81202×10^{-14} , which can satisfy the reversibility requirement for 2-D CAD engineering graphics. Additionally, one of the 50 test graphics is shown in Fig. 1 as well as its watermarked and recovered counterpart. The distortion between the recovered graphics and the original graphics is so small that it is almost impossible to distinguish the difference.

6.2 Analysis of imperceptibility

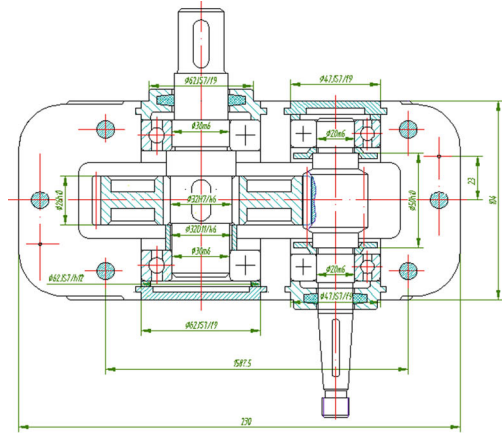
In this section, a set of experiments are done to compare the imperceptibility of the watermarked graphics among the proposed scheme and the scheme in [13, 14]. In the experiments, the embedding capacity is set to be 1 bit/vertex for all those schemes. As seen in Table 2, the *RMSE* values of the watermarked graphics using the proposed watermark scheme are smaller than those using the methods in [13, 14]. It indicates that under the same embedding capacity, our scheme has a better imperceptibility.

According to the principle of our proposed scheme, the distortion inflicted on the graphics is closely related to the quantization step Δ and embedding strength b (s). A first set of

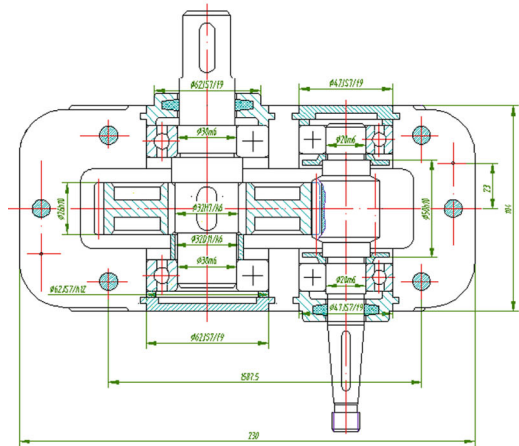
Table 1 The *RMSE* values between the original graphics and the recovered graphics

ID of graphics	Number of vertex	RMES($\times 10^{-14}$)
G1	8182	3.05214
G2	4903	0.54080
G3	18240	2.12368
G4	10115	0.10328
G5	39288	0.82100
Average of 50 graphics	10254	3.81202

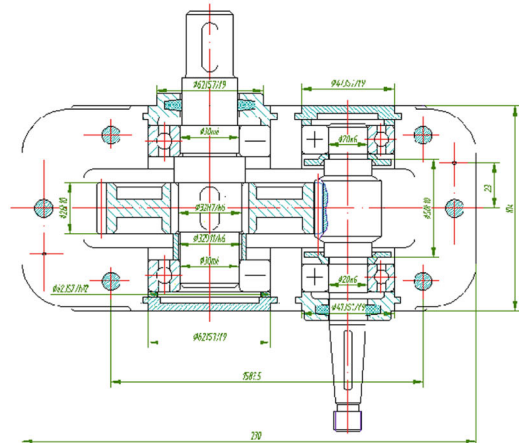
Fig. 1 An example of watermark embedding and data recovery



(a) The original graphics.



(b) The water marked graphics.



(c) The recovered graphics.

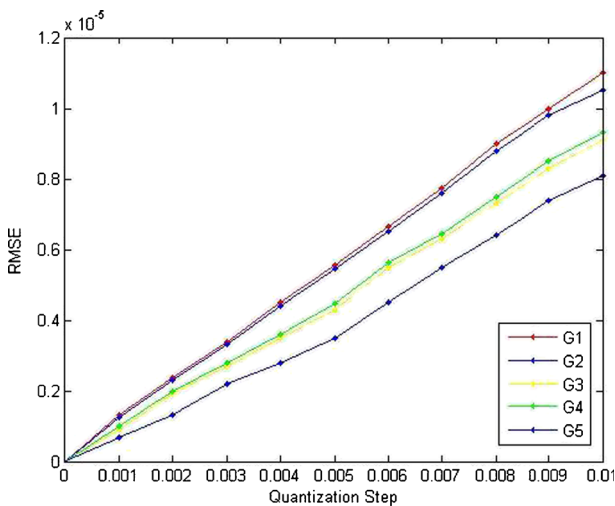
Table 2 The comparison of RMSE between the original graphics and the watermarked graphics with different schemes

ID of graphics	Scheme in [14] ($\times 10^{-3}$)	Scheme in [13] ($\times 10^{-6}$)	Proposed Scheme ($\times 10^{-6}$)
G1	7.31520	1.01851	0.65218
G2	0.06213	1.02312	0.62452
G3	3.01237	0.80324	0.50013
G4	0.51862	0.82150	0.53128
G5	0.21388	0.64012	0.39105
Average of 50 graphics	3.63214	0.81287	0.53162

experiments have been conducted to evaluate the relation between the imperceptibility and the quantization step Δ . Different quantization steps are used to embed a watermark into the 2D CAD engineering graphics G1~G5, and the RMSE between the original graphic and the watermarked counterpart is calculated. The results are illustrated in Fig. 2, which indicates that the imperceptibility of the watermarked graphics is decreased when the quantization step is increased. A second set of experiments have been done to evaluate the relation between the imperceptibility and the embedding strength b ($s=b$). Different embedding strengths are used to embed a watermark into the graphics G1~G5 and the RMSE between the original graphic and the watermarked counterpart is calculated. The results are illustrated in Fig. 3, which indicates that the imperceptibility of the watermarked graphics is decreased when the embedding strength is increased.

6.3 Analysis of capacity

According to the principle of the reversible watermark scheme, the watermark can be embedded into all the vertices except two reference vertices and each vertex can be embedded twice.

**Fig. 2** The relation between imperceptibility and quantization step

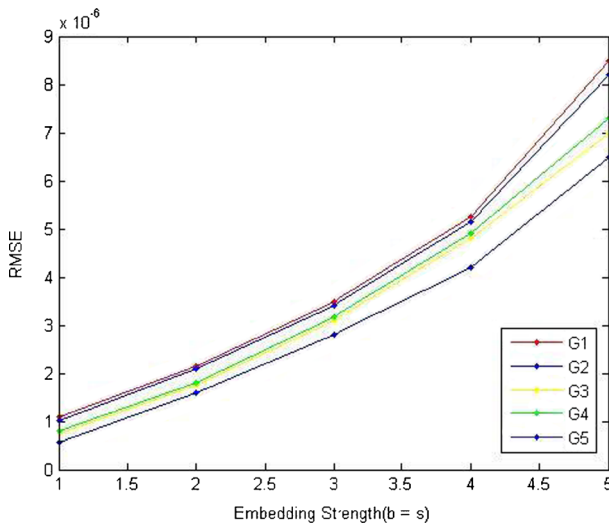


Fig. 3 The relation between imperceptibility and embedding strength

One set of experiments are conducted to compare the embedding capacity with other schemes under the similar transparency. Since the imperceptibility of Scheme [14] is much bigger than that of Scheme [13] and our proposed scheme, we only choose Scheme [13] for capacity comparison with our scheme. In the experiments, we embed two watermarks into each vertex in our proposed scheme, while the scheme in [13] embeds as many watermarks as possible when it achieves similar distortion to our scheme. Table 3 displays the result which indicates that under the similar transparency the embedding capacity of the proposed scheme is much higher than that in scheme [13].

6.4 Analysis of robustness

Translation, rotation, and scaling are typical operations in applications for 2D CAD engineering graphics. Robustness against these operations is very important for a reversible watermark scheme. Experiments have been done to evaluate the robustness against translation, scaling and rotation operations for 2D CAD graphics. Here, the correlation coefficient (NC) is used to indicate the correlation between the original watermark and the extracted watermark. As can be seen from the NC values listed in Table 4, the proposed scheme is robust against translation, rotation, scaling and their combination.

Table 3 The comparison of watermark capacity of different schemes (bit/vertex)

ID of graphics	Scheme in [13]	Proposed Scheme
G1	1.1225	2.0000
G2	1.1202	2.0000
G3	1.1126	2.0000
G4	1.1189	2.0000
G5	1.0988	2.0000
Average of 50 graphics	1.1175	2.0000

Table 4 NC value after different operations on the watermarked 2-D CAD engineering graphics

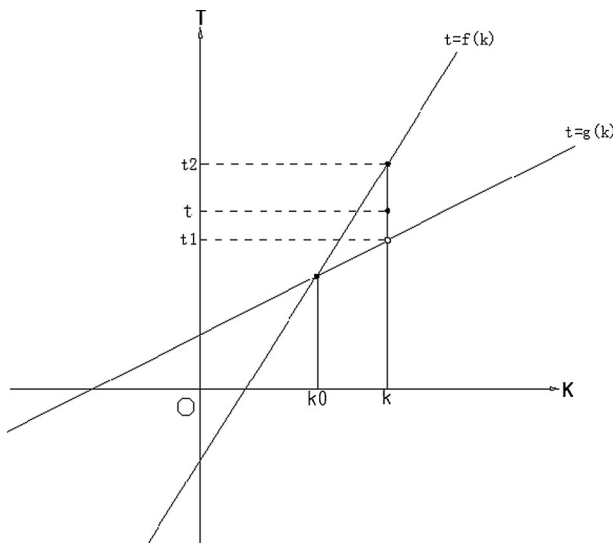
ID of graphics	Translation (100,50)	Scaling ($\rho=2$)	Rotation (30°)	RST combination
G1	1.00000	1.00000	1.00000	1.00000
G2	1.00000	1.00000	1.00000	1.00000
G3	1.00000	1.00000	1.00000	1.00000
G4	1.00000	1.00000	1.00000	1.00000
G5	1.00000	1.00000	1.00000	1.00000
Average of 50 graphics	1.00000	1.00000	1.00000	1.00000

What's more, since the watermark is embedded into the ratio of two relative distances, our scheme is robust against those operations regardless of quantization step or embedding strength. We have also done two set of experiments. One is to evaluate the influence that the quantization step Δ has on robustness in which we use different quantization steps. The other is to evaluate the influence that the embedding strength b ($s=b$) has on robustness in which we use different embedding strengths. All the results are the same as shown in Table 4.

6.5 Analysis of other performance

In this section, we discuss other performance of our proposed scheme including the time complexity and security. The time complexity of the two algorithms in Scheme [13] and [14] are both $O(N)$ (N is the number of vertices in a 2D engineering graphics). Since our proposed scheme combines the two methods in a linear way, we can deduce that our algorithm also has a linear time complexity $O(N)$.

The security of the proposed scheme mainly depends on two factors: the secret key K for selecting reference vertices and the quantization step Δ . Since the quantization step Δ is a real number, it can be used as part of the secret keys to improve the security of the algorithm. If the

**Fig. 4** A geometric method to determinate parameters k and t

quantization step for extraction process is not exactly the same as the one used in the embedding process, the extracted watermarks will be entirely different from the real ones. At the same time, the secret key K is used to determinate the two reference vertices V_p and V_q . Without knowing them, one cannot know where the watermarks have been embedded. Therefore, both of the two keys will greatly improve the security of the proposed scheme.

6.6 Other discussion

Because of the merging of IQIM with IDE techniques, there are some constraints existing in our scheme. In one case where the two reference vertices are exchanged, the content of a graphic will not be changed. However, the extracted watermark will be different from the original watermark, which will cause a false alarm for the integrity of the graphic.

7 Conclusion

In this paper, a reversible watermark scheme which combines IQIM [13] and IDE [14] techniques is proposed. It can solve the embedding-limitation problem existing in [13] so that every vertex of the 2-D CAD engineering graphics can be utilized for embedding watermark. Moreover, the combined scheme achieves high watermark capacity by embedding twice every round with good imperceptibility. In addition, the proposed scheme is robust against translation, rotation, scaling and their combination. Nevertheless, a false alarm will still occur if a swap between the reference points takes place. In our future work, further investigations will be made to improve the robustness for other operations in applications including modifications which do not change the content of graphics.

Acknowledgment The work was funded by the Natural Science Foundation Project of CQ CSTC (Grant No. 2011jjjq40001).

Appendix

In this part, we present the process to determine the values of t and k in detail as a supplement to Section 4.1.

Since we embed watermark in the relative distance r as described in step 2 of Section 3.1, we can calculate the range of the watermarked relative distance r^w as follows.

According to the Eqs. (6)~(7), we get $r=(I+F) \times 2^s \times 10^p$, and according to the Eq. (10), we get $r^w=(I^w+F) \times 10^p$, then

$$\begin{aligned} r^w-r &= (I^w + F) \times 10^p - (I + F) \times 2^s \times 10^p \\ &= (I \times 2^s + w + F) \times 10^p - (I + F) \times 2^s \times 10^p \\ &= (w + (1-2^s) \times F) \times 10^p. \end{aligned} \tag{31}$$

Given $w \in \{0, 1, \dots, 2^s - 1\}$ and $F \in [0, 1)$, we obtain $r^w - r \in (1 - 2^s, 2^s - 1) \times 10^p$. With $r \in [0, 2^b \times \Delta)$, we can get $r^w \in [(1 - 2^s) \times 10^p, 2^b \Delta + (2^s - 1) \times 10^p]$.

By substituting the range of r^w into Eq. (24), we can obtain

$$\begin{cases} (1-2^s) \times 10^p \geq 5 \cdot t \cdot 10^{-(q+1)} - k \\ 2^b \Delta + (2^s - 1) \times 10^p < t(\Delta - 5 \cdot 10^{-(q+1)}) - k \end{cases} \tag{32}$$

Then, it follows that

$$\frac{2^b \Delta + (2^s - 1) \times 10^p + k}{\Delta - 5 \cdot 10^{-(q+1)}} < t \leq \frac{(1 - 2^s) \times 10^p + k}{5 \cdot 10^{-(q+1)}}. \quad (33)$$

To solve the above inequalities, two linear equations are firstly denoted as

$$t = g(k) = \frac{k}{\Delta - 5 \cdot 10^{-(q+1)}} + \frac{2^b \Delta + (2^s - 1) \times 10^p}{\Delta - 5 \cdot 10^{-(q+1)}}, \quad (34)$$

$$t = f(k) = \frac{k}{5 \cdot 10^{-(q+1)}} + \frac{(1 - 2^s) \times 10^p}{5 \cdot 10^{-(q+1)}}. \quad (35)$$

We use a geometric method shown in Fig. 4 which considers the equation geometric meaning. To guarantee Inequality (33) solvable, the slope of Line $t = f(k)$ must be smaller than that of Line $t = g(k)$, that is $\frac{1}{\Delta - 5 \cdot 10^{-(q+1)}} < \frac{1}{5 \cdot 10^{-(q+1)}}$, then we obtain the range which Δ must satisfy is $\Delta > 10^{-q}$.

Let $f(k_0) = g(k_0)$, then we obtain X-coordinate of the intersection between the two lines

$$k_0 = \frac{5 \Delta 2^b \cdot 10^{-(q+1)} + (2^s - 1) \Delta \cdot 10^p}{\Delta - 10^{-q}}. \quad (36)$$

As seen in Fig. 4, when k is larger than k_0 , any value of t within $(t_1, t_2]$ can satisfy Inequality (33).

So finally we manage to obtain the range of the two parameters

$$k > k_0, \quad \frac{2^b \Delta + (2^s - 1) \times 10^p + k}{\Delta - 5 \cdot 10^{-(q+1)}} < t \leq \frac{(1 - 2^s) \times 10^p + k}{5 \cdot 10^{-(q+1)}}. \quad (37)$$

where $k_0 = \frac{5 \Delta 2^b \cdot 10^{-(q+1)} + (2^s - 1) \Delta \cdot 10^p}{\Delta - 10^{-q}}$.

Therefore, as long as t and k meet the above condition, Inequality (33) can be satisfied. Then our watermark scheme will be reversible.

References

- Alattar AM (2004) Reversible watermark using the difference expansion of a generalized integer transform. *IEEE Trans Image Process* 13:1147–1156
- Chaumont M, Puech W (2009) A high capacity reversible watermarking scheme. In: *Proc. SPIE, Visual Communications and Image Processing*. pp 72571H-9
- Coltuc D, Chassery JM (2007) Very fast watermarking by reversible contrast mapping. *IEEE Signal Proc Lett* 14:255–258
- Fridrich J, Goljan M, Du R (2001) Invertible authentication. In: *Proc. SPIE, Security and Watermarking of Multimedia Contents*. pp 197–208
- Fridrich J, Goljan M, Du R (2001) Distortion-free data embedding for images. In: *Proc. 4th Information Hiding Workshop*. pp 27–41
- Hong W, Chen TS, Chang YP, Shiu CW (2010) A high capacity reversible data hiding scheme using orthogonal projection and prediction error modification. *Signal Process* 90:2911–2922

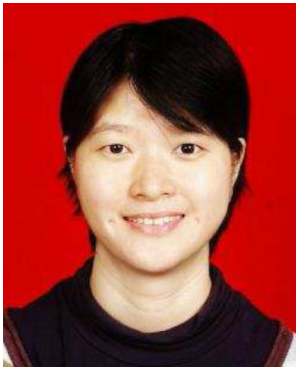
7. Honsinger CW, Jones P, Rabbani M, Stoffel JC (2001) Lossless recovery of an original image containing embedded data, U.S. Patent 6278791
8. Ji HM, Yang XQ, Zhang CM, Gao XF (2010) A new reversible watermarking of 3D models based on ratio expansion. In: Proc. 2010 3rd International Congress on Image and Signal Processing (CISP). pp 3899–3903
9. Kwon KR, Jang BJ, Lee EJ, Huh Y (2004) Copyright protection of architectural CAD drawing using the multiple watermarking schemes. In: IEEE international conference on multimedia and expo. ICME. pp 871–874
10. Lee H, Dikici C, Lavoue G, Dupont F (2011) Joint reversible watermarking and progressive compression of 3D meshes. *Vis Comput* 27:781–792
11. Lee SH, Kwon KR (2010) CAD drawing watermarking scheme. *Digit Signal Process* 20:1379–1399
12. Peng F, Guo RS, Li CT, Long M (2010) A semi-fragile watermarking algorithm for authenticating 2D CAD engineering graphics based on log-polar translation. *Comput Aided Des* 42:1207–1216
13. Peng F, Lei YZ (2011) An effective reversible watermarking for 2D CAD engineering graphics based on improved QIM. *Int J Digit Crime Forensic* 3:53–69
14. Peng F, Lei YZ, Long M, Sun XM (2011) A reversible watermarking scheme for two-dimensional CAD engineering graphics based on improved difference expansion. *Comput Aided Des* 43:1018–1124
15. Tian J (2003) Reversible data embedding using a difference expansion. *IEEE Trans Circ Syst Video Technol* 13:890–896
16. Tsai P, Hu YC, Yeh HL (2009) Reversible image hiding scheme using predictive coding and histogram shifting. *Signal Process* 89:1129–1143
17. Voigt M, Yang B, Bush C (2004) Reversible watermarking of 2D-vector data. In: Proc. Multimedia and Security Workshop. pp 160–165
18. Wang XT, Shao CT, Xu XG, Niu XM (2007) Reversible data-hiding scheme for 2-d vector maps based on difference expansion. *IEEE Trans Inf Forensic Secur* 2:311–320
19. Wang C-T, Yu H-F (2012) High-capacity reversible data hiding based on multi-histogram modification. *Multimed Tools Appl* 61:299–319
20. Zhao H, Wang H, Khurram Khan M (2011) Statistical analysis of several reversible data hiding algorithms. *Multimed Tools Appl* 52:277–290
21. Zhu AP, Zhang CM, Yang XQ, Gao XF (2010) Reversible watermarking of 3D mesh models using prediction-error expansion. In: Proc. 2010 3rd International Congress on Image and Signal Processing (CISP). pp 1171–1175



Di Xiao received the Ph. D. degree in computer software and theory from Chongqing University, Chongqing, China in 2005. From 2006 to 2008, he has done postdoctoral research at Chongqing University. From 2008 to 2009, he has been a visiting scholar funded by the Chinese government at the Department of Computer Science, New Jersey Institute of Technology, USA. At present, he is a professor at College of Computer Science, Chongqing University, China. His research interests include image processing, chaos based cryptography, image and graphics watermarking, etc. He is a member of IEEE and ACM.



Shulei Hu received the Bachelor's degree in Engineering from Yangtze University of China, Hubei, China in 2011. Currently, he is pursuing his Master degree in computer science from College of Computer Science, Chongqing University, China. His research interests include reversible watermarking, graphics watermarking, etc.



Hongying Zheng received the Ph. D. degree in computer software and theory from Chongqing University, Chongqing, China in 2007. At present, she is an associate professor at the College of Computer Science, Chongqing University, Chongqing, China. Her research interests cover image processing, image and graphics watermarking, etc. More than 20 academic papers have been published since 2004.