Journal of Information Hiding and Multimedia Signal Processing ©2017 ISSN 2073-4212
Ubiquitous International Volume **8**, Number **3**, May **2017**

Reversible Data Hiding Scheme based on Fractal Image Coding

Wen-Hua Ren
Department of Computer and Information Technology
Zhejiang Police College
Hangzhou, 310053, P.R. China
renwenhua999@163.com

Xiang Li and Zhe-Ming Lu*
School of Aeronautics and Astronautics
Zhejiang University
Hangzhou, 310027, P. R. China
*Corresponding author: zheminglu@zju.edu.cn

Received July 2016; revised December 2016

Abstract. *In this paper, a novel reversible data hiding scheme based on fractal image coding is proposed. As a novel theory, fractal theory has drawn more and more attention in recent years. It has been applied to many fields such as physics. Fractal theory is also used in image compression. It can utilize the self-similarity of an image to reduce data redundancy. Like other image coding methods, fractal image coding has also been combined with data hiding. Some schemes have already been proposed. But most of the existing schemes do not emphasize the reversibility of the scheme. This letter proposes a reversible data hiding scheme based on fractal image coding. In fractal image coding, the position of each domain block in the domain block pool depends on the sliding step size of a window. The step size can be set to even. As a result, the coordinate values of the upper left corner of all domain blocks in the pool are even. As the parameters of a range block's fractal codes, the coordinate values of the best matching domain block are also even. So we can utilize odd-even feature of the coordinate values to embed data. In this scheme, the coordinate values of the upper left corner of the best matching domain block in the fractal codes of an image are slightly modified for embedding data. The experiment results prove that the proposed scheme is effective.*

Keywords: Information hiding, Reversible data hiding, Fractal image coding, Iterated function system(IFS).

1. **Introduction.** The concept of fractal was proposed by Mandelbrot [1]. As a novel nonlinear science theory, fractal theory has obtained rapid development. It has been used in many fields such as physics and biology. In 1988, Barnsley and Sloan [2] proposed a scheme based on fractals generated by iterated function system (IFS) to compress images. The scheme introduced fractal theory to the field of image compression. But in their scheme, image coding process needs human intervention. So the non-automatic coding scheme is difficult for practical applications. Later, Jacquin [3] proposed a block-based fractal image compression scheme based on partitioned iterated function system (PIFS). In his scheme, fractal image coding can be automatically done by computer. It is a great boost to the development of fractal image coding. Later, some new fractal image coding methods were proposed. For example, Fisher [4] proposed a good fractal coding scheme

based on quad-tree partition. Fractal image coding has many good properties such as high compression ratio and fast decoding. It has gained more and more attention in recent years.

With the development of the Internet, network information security has become more and more important. Reversible data hiding is a new technique in the field of information security[5, 6]. It can embed some important data into digital image files. Then the sender can transfer these files to the specified receiver on the Internet. The receiver can extract the hidden data from the files by specific data extraction algorithm. Other people are difficult to find the hidden data. Even though they find the existence of hidden data, they do not know how to extract the data from the files. So reversible data hiding can enhance the security of transferred information greatly. In addition, reversible data hiding technique demands data embedding process is reversible or lossless. In other words, after extracting the data from a stego image, the image must be restored to its original state. Namely, data embedding can not affect the normal use of the cover image. Some reversible data hiding schemes for other compressed image types have already been proposed. Li et al. [7] proposed a reversible data hiding scheme for block truncation coding (BTC) compressed images. Chu et al. [8] proposed a reversible scheme for vector quantization (VQ) compressed images. Hu et al. [9] proposed a reversible scheme for JPEG compressed images.

With regard to fractal compressed images, some data hiding schemes based on fractal image coding have already been proposed. In 1996, Puate and Jordan [10] proposed a fractal image compression scheme to embed a digital signature into an image. In their scheme, the domain block pool is divided into two square local searching regions (LSR) around the range block. The signature is embedded into the range blocks according to which local searching region the best matching domain block belongs to. Zhao et al. [11] proposed a different local searching region partition scheme. Two local searching regions are positioned alternately by columns in the pool. Davern and Scott [12] also proposed a data hiding scheme. But they divide the domain block pool in a different way. The domain block pool is split into two halves. Then the watermark is embedded into the selected range blocks based on which half the best matching domain block belongs to. These schemes can embed data into images, but they do not emphasize the reversibility of the scheme. Li and Wang [13] proposed a reversible watermarking scheme. In this scheme, they classify eight kinds of isometric transformations into two classes. So a digital watermark bit can be embedded according to which class the best isometric transformation belongs to.

In the paper, we propose a novel reversible data hiding scheme based on fractal image coding. The above schemes select to embed data by classifying the fractal codes of an image. However, our scheme takes a different approach. We decide to embed data by modifying the fractal codes. The fractal codes of a range block include the coordinate values of the upper left corner of the best matching domain block, best isometric transformation index, contrast scaling factor and luminance offset. In the process of constructing domain block pool, the position of each domain block in the pool depends on the sliding step size of a window. If the step size is even, the coordinate values of the upper left corner of all domain blocks including the best matching domain block are even. So we can take advantage of odd-even feature to modify the coordinate values for embedding data. In data extraction process, a data bit can be extracted from a coordinate value according to whether the coordinate value is even or not. In our scheme, each range block can embed two data bits. In the schemes mentioned above, each range block can embed one bit. So the embedding capacity of our scheme is higher. In addition, our scheme is reversible. The experiment results demonstrate that the proposed scheme is effective.

The rest of the paper is organized as follows. Section 2 introduces fractal image coding in detail. The proposed reversible data hiding scheme is presented in Section 3. The experiment results are given in Section 4. Finally, we draw a conclusion in Section 5. Complex systems widely exist in natural, social, biological, engineering, and many other fields. It is one of the most challenging issues to research the modeling problem of complex systems, and it has an important significance for deeply analyzing and scientifically understanding the internal relationship between system structures, functions and dynamics. The complex network theory provides a new way of thinking and perspective for the research of complex science, especially complex systems.

2. Fractal Image Coding.

Fractal image coding is a new image compression technique. It utilizes the self-similarity of an image to compress image. Iterated function system (IFS) theory is the mathematic foundation of fractal image coding. In fractal image coding, Jacquin's block-based coding algorithm is very classical. Our reversible data hiding scheme is also based on this image coding algorithm. So we will introduce this classical fractal image coding algorithm next. Fractal image coding includes two parts: image encoding and image decoding. We will introduce fractal image encoding first.

2.1. Fractal Image Encoding.

For an original image I with the size of $N \times N$, it will be divided into two different sizes of square blocks first. The smaller non-overlapping blocks of size $B \times B$ are called range blocks. The number of range blocks n_R is equal to $(N/B)^2$. The bigger blocks of size $2B \times 2B$ are called domain blocks. A domain block pool can be obtained by sliding a window of size $2B \times 2B$ within the image. The initial position of the window is the upper left corner of the image, namely (0, 0). Then it will move to next position by steps of either ph pixels horizontally or p_v pixels vertically. p_h is usually equal to p_v. Let p denote the sliding step size of the window. In application, p is usually set to B or $2B$. The domain blocks are allowed to be overlapped. The number of domain blocks n_D is equal to $((N-2B)/p+1)^2$.

Next, for each range block $R_i (i = 1, 2, \ldots, n_R)$, we need to search the domain block pool to find a best matching domain block $D_j (j = 1, 2, \ldots, n_D)$ which is closest to range block R_i after affine transformation w_i, namely $R_i \approx w_i(D_j)$. The affine transformation wi is made up of spatial contractive transformation S_i, isometric transformation F_i and grayscale transformation G_i.

Firstly, the domain blocks must shrink to the size of range blocks by spatial contractive transformation S_i. The spatial contraction operation can be formulated as follows:

$$d'_{i,j} = (d_{2i,2j} + d_{2i+1,2j} + d_{2i,2j+1} + d_{2i+1,2j+1})/4 \tag{1}$$

where $d_{i,j}$ and $d'_{i,j}$ denote the pixel value at location (i, j) in domain block and contractive domain block respectively.

Secondly, isometric transformation F_i is applied to all contractive domain blocks. There are eight kinds of isometric transformations including identity, four kinds of orthogonal reflections and three kinds of rotations around center of block.

At last, grayscale transformation G_i is performed on the domain blocks. It includes contrast scaling and luminance shifting.

After these transformations, we can obtain all transformed domain blocks D'_j. Then compare range block R_i with all transformed domain blocks D'_j in the pool. We select mean square error (MSE) as the inter-block distortion measure. Compute the MSE value between range block R_i and each block D'_j. Find the minimum MSE value in all obtained MSE values. The domain block D_j corresponding to the minimum MSE value is the best matching domain block with range block R_i. The position information of the best matching domain block and the parameters of the corresponding affine transformation wi constitute the fractal codes of range block R_i. The fractal codes of block R_i consist of the coordinate values (x_i, y_i) of the upper left corner of the best matching domain block D_j, the best isometric transformation index k_i, contrast scaling factor s_i and luminance offset o_i.

In this way, the fractal codes of all range blocks can be obtained. So the original image I can be represented by these fractal codes. These fractal codes can be regarded as compressed bit-stream of the image I. The data size of these fractal codes is very small. We can store and transfer these fractal codes instead of image data. So it can reduce image data redundancy greatly.

2.2. **Fractal Image Decoding.** Next we will introduce fractal image decoding. The decoding process is simpler than encoding process. It is a simple iteration process. Select an arbitrary image of same size with the original image I as the initial image. The decoded image I' can be obtained by iterating the affine transformation wi denoted in the fractal codes on the initial image.

3. **The Proposed Scheme.** In this section, the proposed reversible data hiding scheme based on fractal image coding is presented. Most of the existing data hiding schemes embed data by confining searching region of domain block in the pool. So the matching domain block is just local optimal, not global optimal. The matching domain block found in the specific region is not the best matching one for a range block. It may have some impact on the visual quality of the decoded image. In addition, such schemes are usually irreversible. In decoding, the original decoded image can not be obtained for lack of original global optimal fractal codes. In our scheme, we adopt different method to embed data. Through the above introduction of fractal image coding, we can know that a domain block pool is obtained by sliding a window of size $2B \times 2B$ with step size p across the image. So the position of each domain block in the pool depends on the step size p. If p is even, the coordinate values of the upper left corner of the domain blocks in the pool are all even. So we can use odd-even feature to embed data. Moreover, it is simple to realize blind extraction based on odd-even feature. So we take advantage of this feature to embed data by modifying the coordinate values of the best matching domain block in the fractal codes of an image.

3.1. **Data Embedding.** The detailed data embedding process is described as follows:

(1) Perform fractal image encoding on the original image I. In fractal image encoding, make sure that the sliding step size p is set to even. So the original fractal codes of the image can be obtained.

(2) In order to realize blind extraction, we must embed some auxiliary information to let the decoder know which range block data embedding ends in. We select the number of hidden valid characters l as the auxiliary information. The auxiliary information and the valid information constitute the hidden data bit sequence. Let w_j denote the bit sequence $(j = 1, 2, \ldots, m)$. The bits of auxiliary information l should be embedded before the bits of valid information.

(3) For range block R_i, the coordinate values in the fractal codes of block R_i are defined as x_i and $y_i (i = 1, 2, \ldots, n_R)$. In block R_i, two data bits will be embedded. If $w_j = 1$, x_i is added by 1. If $w_j = 0$, x_i remains unchanged. In the same way, w_{j+1} is embedded into y_i. All hidden data bits will be embedded into the fractal codes of range blocks in order of range block index i. If the slight modification leads to the sliding window in the domain block pool is beyond the image region in direct image decoding based on the modified fractal codes, we need to take some special measures to deal with the rare case. For the

pixels beyond the image region, we can fill them with their adjacent pixels according to the correlation between adjacent pixels in the image.

(4) After embedding all data bits, new fractal codes of the original image I can be obtained.

3.2. **Data Extraction.** The data extraction process is simple. It is just the reverse of data embedding. The detailed data extraction process is presented as follows:

(1) Extract the hidden data bits according to whether the coordinate values of the obtained fractal codes are even or not. If a coordinate value is even, bit '0' is extracted. Otherwise, bit '1' is extracted. In this way, extract the data bits in order of range block index i until all hidden data bits have been extracted.

(2) In order to realize the reversibility of the scheme, we need to restore the obtained fractal codes to their original state. Check the coordinate values of the obtained fractal codes. If a coordinate value is odd, the value will be subtracted by 1. Otherwise, the value remains unchanged. So we can get the original fractal codes of image I. The original decoded image corresponding to image I can be obtained according to the original fractal codes. So the impact of data embedding on the decoded image can be removed.

4. **Experiments and Analysis.** In this section, some experiments are given to demonstrate the effectiveness of the proposed scheme. We select grayscale image 'Lena' of size 256×256 as the test image. The data to be embedded is a string of English characters. The size of range block is $B \times B$. The size of domain block is $2B \times 2B$. The value of B is a power of 2. In fractal image coding, B is an important parameter. In our scheme, the embedding capacity of the scheme depends on B. The smaller is B, the higher is the embedding capacity. In addition, the value of B is also related to the visual quality of the decoded image. Next we will study the relationship between the value of B and the visual quality of the decoded image. In this experiment, the sliding step size p is set to $2B$. The decoded images corresponding to $B = 4, 8, 16$ respectively are shown in Fig.1. From Fig.1, we can clearly see that the visual quality of the decoded image decreases with the increase of B. In order to measure the visual quality of the decoded image accurately, we choose peak signal noise ratio (PSNR) between the original image and the decoded image as the metric. Larger PSNR means better visual quality of the decoded image. The results are shown in Table 1.

TABLE 1. Results of PSNR values when B changes

B(pixel)	PSNR(dB)
4	24.49
8	22.55
16	17.75

From Table 1, we can see that PSNR decreases with the increase of B. But B is not the smaller the better. If B is equal to 2, image encoding time will be very long. So we have to find a balance between visual quality of the decoded image and image encoding time. According to our experiments, the visual quality of the decoded image and image encoding time are both acceptable when $B = 4$.

Next we will study the relation between embedded data size and PSNR of the decode image.

Let $PSNR_1$ denote the PSNR value between the original decoded image generated by the original fractal codes and the stego decoded image generated by the modified fractal codes which hide data. Let PSNR2 denote the PSNR value between the original

FIGURE 1. Original image and the decoded images.(a)original image;(b) decode image ($B = 4$);(c) decoded image ($B = 8$);(d)decoded image ($B = 16$)

TABLE 2. Results of PSNR values when data size changes

Data size(bit)	PSNR1(dB)	PSNR2(dB)
0	∞	∞
512	30.82	∞
1024	25.43	∞

decoded image and the restored decoded image obtained after extracting data. We embed different amounts of data bits into the original fractal codes to observe the change of these parameters. B is set to 4 and p is set to 8 here. The results are shown in Table 2. From Table 2, we can see that $\mathtt{PSNR}_1$ decreases with the increase of embedded data size. So we need to balance embedded data size and visual quality of the stego decoded image in application. In the experiments, all hidden data can be extracted correctly. The bit error rate is 0. In addition, we can find that the values of $\mathtt{PSNR}_2$ are all infinity. It proves that our data hiding scheme is reversible.

5. **Conclusions.** In this paper, a new reversible data hiding scheme based on fractal image coding is proposed. In this scheme, we set the step size of the sliding window to even. So the coordinate values of all domain blocks including the best matching domain block are even. The odd-even feature of the coordinate values can be used for embedding data. So we modify the coordinate values of the best matching domain block in an image's

fractal codes to embed data. The experiments have proved that the proposed scheme is effective.

Acknowledgment. This work was partially supported by the National Natural Science Foundation of China under grant 61171150, the Nature Science Foundation of Guangdong Province (2015A030310172), and the Science and Technology Plan Projects of Shenzhen (JCYJ20150324140036830, GJHZ20160226202520268).

REFERENCES

[1] B. Mandelbrot, The Fractal Geometry of Nature, San Francisco, CA: Freeman, 1982.

[2] M. Barnsley and A. D. Sloan, A Better Way to Compress Images, *Bytes*, vol. 13, pp. 215–233, 1988.

[3] A. E. Jacquin, Fractal image coding: A review, *Proceedings of the IEEE*, vol. 81, pp. 1451–1465, 1993.

[4] Y. Fisher, Ed., Fractal Image Compression: Theory and Application, Springer-Verlag, New York, 1994.

[5] S. W. Weng, J. S. Pan and L. D. Li, Reversible data hiding based on an adaptive pixel-embedding strategy and two-layer embedding, *Information Sciences*, vol. 369, no. 10, pp. 144–159, 2016.

[6] S. W. Weng, Y. J. Liu, J. S. Pan and N. Cai, Reversible data hiding based on flexible block-partition and adaptive block-modification strategy, *Journal of Visual Communication and Image Representation*, vol. 41, no. 11, pp.185–199, 2016.

[7] C. H. Li, Z. M. Lu and Y. X. Su, Reversible data hiding for BTC-compressed images based on bit-plane flipping and histogram shifting of mean tables, *Information Technology Journal*, vol. 10, no. 3, pp. 1421–1426, 2013.

[8] D. H. Chu, Z. M. Lu and J. X. Wang, A high capacity reversible information hiding algorithm based on difference coding of VQ indices, *ICIC Express Letters, Part B: Applications*, vol. 3, no. 4, pp.701–706, 2012.

[9] Y. J. Hu, K. Wang and Z. M. Lu, An improved VLC-based lossless data hiding scheme for JPEG images, *Journal of Systems and Software*, vol. 86, no. 8, pp.2166–2173, 2013.

[10] J. Puate and F. Jordan, Using fractal compression scheme to embed a digital watermark into image, *Proceedings of SPIE Photonics East Symp.*, Boston, MA, Nov.18-22,1996.

[11] Y. Zhao, Q. Ma, B. Z. Y, Digital watermark based on LIFS, *The Proceedings of ICIP'2000*, vol. 2, pp. 1281–1284, August 2000.

[12] P. Davern and M. Scott, Fractal based image steganography, *Information Hiding: First International Workshop*, no. 1174, pp. 279–294, Springer Lecture Notes. Cambridge, U. K, 1996.

[13] C. H. Li and S. S. Wang, Digital watermarking using fractal image coding,*IEICE Transactions on Fundamentals*, vol. E83-A, no. 6, pp. 1286–1288, 2000.