Fast Image Steganography Using Affine Mapping

Loay E. George * Ahmed H. Ali **

* Computer Science Department\ College of Science\University Of Baghdad

** Computer Science Department\ Studies, Planning and Follow up Directorate\Ministry Of Higher Education and Scientific Research

Receiving Date: 2011/6/12 - **Accept Date:** 2011/10/4

Abstract

One of the most important characteristics of using fractal image coding in hiding images is the encoding process that involves very long time complexity. In this paper new hiding system is proposed that hide secret image (color or gray) into cover image (color) with reducing the time of encoding process 10 times less than classical systems that use block indexing as descriptor with fractal coding. Affine transform is used in matching process to find the closer cover block for each secret block and least significant bit method to embed the coefficient affine transform. The results indicated that the encoding time is actually reduced and hiding rate reach to 100% and the quality of extracted secret image is still within the acceptable level.

المستخلص

يعتبر عامل الوقت من الخصائص المهمة التي تميز عملية ترميز الصور الكسورية والذي يكون طويل جدا، لذا يقترح هذا البحث نظام اخفاء جديد يقوم باخفاء الصورة السرية (ملونة او رمادية) في الصورة الغطاء (ملونة) حيث يقوم هذا النظام بتقليل الوقت الذي تحتاجة عملية التشفير الى 10 مرات اقل من الأنظمة التقليدية (والتي وتستخدم عمليه ترميز الصور الكسرية فقط) باستخدام طريقة (فهرسه البلوك). تم استخدام التتحويل الأفيني في عملية المطابقة لايجاد اقرب بلوك من بلوكات الصورة الغطاء لكل بلوك من بلوكات الصورة السرية وطريقة اخفاء اصغر بت في عملية اخفاء معاملات التحويل واظهرت النتائج الى ان وقت الترميز تم انقاصة فعليا وان نسبة الاخفاء قد نصل الى 100 % بدون تغيير على كفاءة الصورة السرية.

Keywords: FBC (Fractal Block Indexing), PIFS (Partition Iteration Function System), (CATs), Contractive Affine Transformations FIC (Fractal Image Coding)

Introduction

Image coding based on fractal theories and techniques are recently developed and received a great deal of attention [1]. Jacquin proposed the (FBC) scheme to automatically convert an image into a partitioned iteration function system, which is a set of (CATs) [2]. The small range block R and large domain block D used in CAT both are partitioned from an image. The coefficients represent the best-match transformation which minimizes error. A number of papers on fractal image encoding have been published since the pioneering idea of Barnsley and Sloan in 1988 [3]. All attempts to speed-up PIFS encode method dealt with encoding the gray-scale images (or images consist of only one color plane) [3]. Many researchers studied the implementation of FIC on color images in ways different than that followed in JPEG or JPEG 2000 [4,5]. Also many studies apply moment features in FIC to speeding color and gray image compression by using different method [6, 7].

Proposed System

Proposed system hides secret image (color or gray) into cover image (color) that use affain mapping with block indexing as descriptor with fractal coding.

Block Indexing

In this paper we adopt block indexing techniques that are used to reduce exhaustive search in matching process. The blocks are classified into classes depending on equation (1) each class have several blocks:

$$F = \left| \frac{m_{01}^2 - m_{10}^2}{m_{01}^2 + m_{10}^2} * N \right| \qquad \dots (1)$$

Where

F= {0 . . 100}

N={50,100,200 ,…,800}

F class and N no. of classes

$$m_{01} = \sum_{y=0}^{blocksize-1} \sum_{x=0}^{blocksize-1} x \times image(x, y) \qquad \dots.(2)$$

$$m_{10} = \sum_{y=0}^{blocksize-1} \sum_{x=0}^{blocksize-1} y \times image(x, y) \qquad \dots.(3)$$

Affain Mapping

Affine transform is a set of coefficients that used to build the constructed image from the cover image so that each block in secret image can be obtained from blocks of cover image by applying the suitable rotation and reflection operations after partitioning the secret image into blocks of fixed size [8].

For each cover block an approximation must be obtained from equation (4)

$$s \approx L c + F \qquad \dots (4)$$

Where

s scale block

c cover block

L scale

F offset

The computation needs the following steps

-For each *s* block make the suitable symmetry and compute an optimal approximation as follow:

- Calculate the values of scaling (*L*) and offset (*F*) coefficients by using the least square optimization as equation (5),(6)

$$L = \frac{n\,(\sum_{x,y} c_i s_i) - (\sum_{x,y} c_i)(\sum_{x,y} s_i)}{n(\sum c_i^2) - (\sum_{x,y} c_i)^2} \qquad \ldots(5)$$

And

$$F = \frac{1}{n}(\sum_{x,y} s_i) - n(\sum_{x,y} c_i) \qquad \ldots(6)$$

Where s is secret block and c cover block and n no. of pixels.

- use the quantized coefficients (L) and (F) to compute the error $Err(c,s)$ equation(7):

$$Err^2 = \frac{1}{n}[\sum_{x,y} s^2 + L(L\sum c_i^2 - 2\sum_{x,y} c_i s_i + 2F\sum_{x,y} c_i) + F(Fn - 2\sum_{x,y} s_i)] \qquad \ldots(7)$$

- Find the c blocks with minimal error.
- The output is the set of quantized coefficient (L and F) and symmetry index and the c blocks.

The symmetry matching done as follow:

-identity

- Reflection around Y-axis
- Reflection around X-axis
- 180^o rotation.
- Reflection around the diagonal Y=X
- 90^o rotation
- 270^o rotation
- Reflection around the diagonal Y=-X

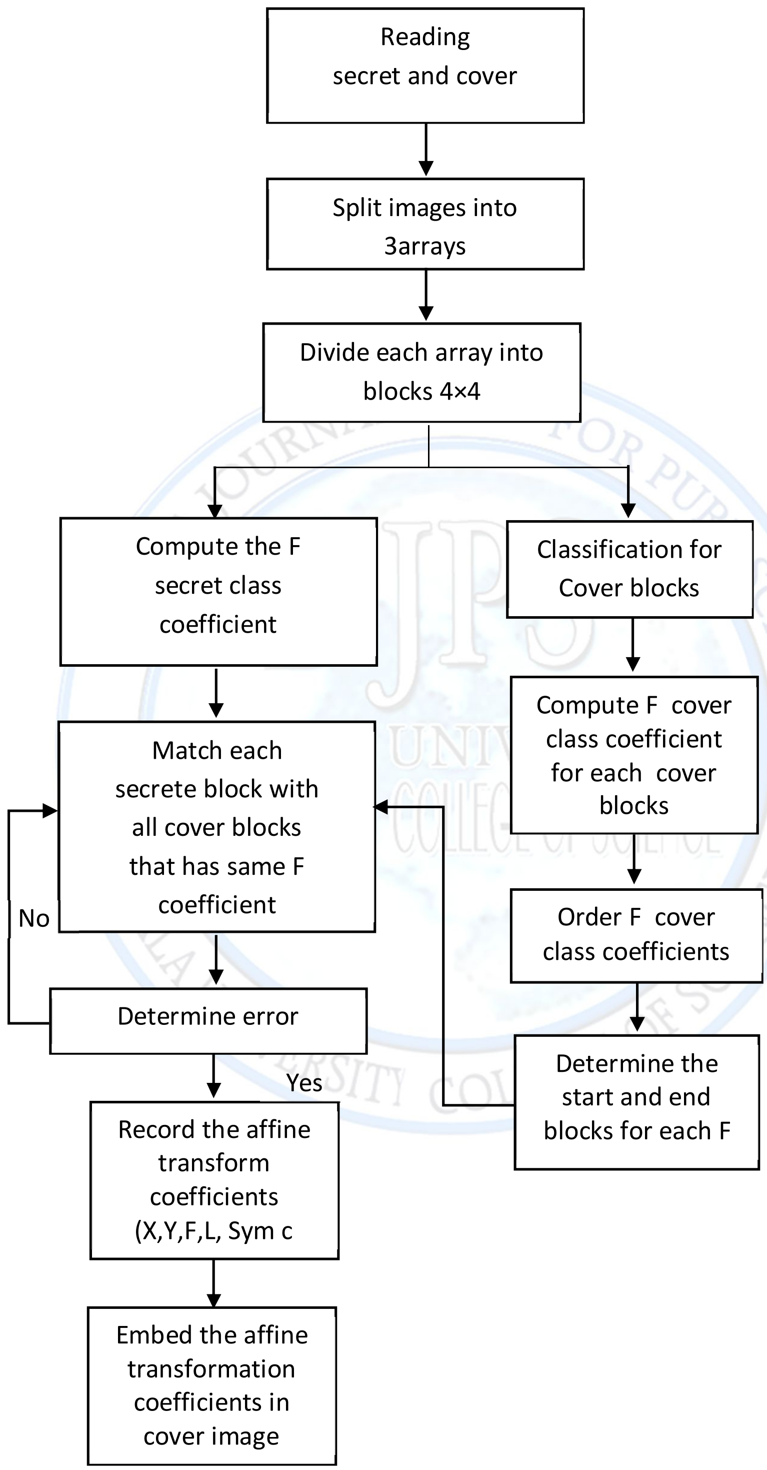

Figure (1) The Proposed System

Encoding process

Encoding process begin by reading secret_ images, divide the images into 3 arrays, and into blocks (4x4) then we use block indexing method to improve (speed up) the secret search task. Instead of compare all cover blocks with each affine transformed secret block, we need only to test the cover blocks whose F class values are similar to that of the tested secret block. To implement this idea the following block indexing algorithm is built as shown in Figure (1):

1. Load BMP images (Secret and Cover) and put it in (R,G,B) array (three 2D arrays).
2. For each cover block:
   a. Compute F cover class values (equation 1).
   b. Store the position *(Xs, Ys)* of the cover block and its F cover class values in array *T*.
2. Sort the records of the array *T* in ascending order according to their F value.
3. Establish start and end of each block of records that have same F cover class value.
4. for each secret block:
   a. Determine F secret class values (equation 1).
   b. depend on the array *T,* match only the cover blocks whose *Fs* values equal to F cover class. In each matching instance determine the *L* and F and *Err* (equations 5,6,7) for all possible symmetry cases (sym=0,1,.. .,7).
   c. Compare the result *(Err)* of each matching instance with the minimum Err registered during the previous matching instances. If *Err* is smaller then put its value in minimum Err register values of (L,F, Sym, Xs , Ys).
   d. In the case that the new registered minimum *Err* is less than error between matched blocks then stop the search process, and output the set (L,F,Sym, Xs , Ys) as best match, and go to step (4e).
   e. Otherwise, start test another cover blocks until either the registered minimum error become less than (Err_{max}) or all the cover blocks are tested.
   f. Output the set of IFS code (L,F,Sym, Xs , Ys) for the tested secret block.
5. After the affine transformation coding of all secret blocks, apply the coding method to encode the sequence of affine transformation coefficients.

Extraction process

In this process we built the secret image from cover image by:

1- For each layer of cover image:

a- Retrieve the array of coefficient from cover image by collect the least significant bit from each byte from cover image.

b- Apply the coefficient on cover block to obtain secret block depend on the equation (4).

c- Put the secret block in the right position in the reconstruction image (secrete image).

2- Merge the 3 layers of the reconstructed image to built the secrete image.

Test Result

The proposed methods are tested on cover image (Lena image 256 x 256, 24 bits) and secret image (Lena image 128 x 128, 8 or 24 bits) Figure (2),(3). The size of secret blocks is 4 x 4 pixels; the search step size is 1, the number of bits allocated for the scaling and offset factor is 13 bits, and for the symmetry factor 3 bits. So the hiding rate would be 100%. All methods were programmed using visual basic 6.0 and implemented on a Dell laptop with Intel Core 2 Due 2.0 GHz CPU.

Figure (1) **Figure (2)**

Cover Image **Secret Image**

The results listed in Table (1,2,3,4,5) illustrate the effect of using step size, block length and no. of classes only the error threshold (*Err*) as a stopping search condition with time of encoding (T_E), *(MSE)* and *(PSNR)* as a descriptor to measure the quality of the secret image.

$$MSE = \frac{1}{w \times h} \sum_{x,y} (I_{x,y} - I`_{x,y})^2$$

$$PSNR = \frac{w \times h \times 255^2}{(I_{x,y} - I`_{x,y})^2}$$

Where *w, h* is height and width of the image.

I, I` represent pixel in two image.

Step size	Time in Minute	MSE	PSNR
1	83.85	9	385
5	5.5	11	37.6
8	2.2	11.6	37.4
10	1.4	12	37.3

Table (1) Effect of step size on classical system with block length=4 and hiding rate 0.35

Step size	Time in Minute	MSE	PSNR
1	6.86	10	38
5	0.45	12.8	37
8	0.2	13.4	36.8
10	0.14	13.7	36.7

Table (2) Effect of step size on proposed system where block length=4 and no of classes 100 and hiding rate 0.35

Block length	Time in Minute	Hiding rate	MSE	PSNR
4	1.4	0.35	12	37.3
5	1.1	0.5	20	35
6	1	0.8	19.3	35.2
7	0.8	1.08	20.2	35

Table (3) Effect block length on classical system with step=10

Block length	Time in Minute	Hiding rate	MSE	PSNR
4	0.14	0.35	13.7	36.7
5	0.11	0.5	20.8	34.9
6	0.1	0.8	20.6	34.9
7	0.08	1.08	21	34.8

Table (4) Effect of block length on proposed system with step size=10 and no of classes=100

No of Classes	Time in Minute	MSE	PSNR
50	0.15	13.2	36.9
100	0.14	13.7	36.7
300	0.1	14.5	36.4
500	0.06	14.7	36.4

Table (5) Effect number of classes on proposed system with step block length=4 and size=10 and hiding rate=0.35

Conclusion

As seen in the result that the classical system using FIC only uses: step size 10, hiding rate (0.35) and the required time(1.4) minute, *PSNR is* (37.5) dB as shown in table (1) while the proposed system uses: step size 10, hiding rate(0.35) and the required time (0.14) minute, *PSNR is* (36.7) dB as shown in table (2) this indicates that the encoding time is reduced by 10 times and hiding rate reach to 100% using 1-least significant bit in hiding method and the quality of extracted secret image is still within the acceptable level.

References

1. Fisher Y. " Fractal Image Compression: Theory and Application", Springer-Verlag, New York, 1995.
2. Jacquin A.E. "Fractal image coding: a review," Proceedings of the IEEE, vol. 81, pp. 1451-1465, October 1993.
3. Jacquin, A.," *Image Coding based on a Fractal Theory of Iterated Contractive Image Transformations"* IEEE trans. Image processing, vol. 1, pp.18-30, 1992.
4. Hurtgen, B., Mols, P., and Simon, S., "Fractal Transforms Coding of Color Images", Visual Communications and Image Processing, SPIE 94, Chicago, pp.1683-1691, 1994.
5. Zhaohuili, Zhao, L, Soma, Y. ,"Fractal Color Image Compression", IEEE, 0-7695-08782/00, 2000
6. Loay A. George, "IFS coding for zero-mean image blocks", Iraqi J. Sci., vol. 47, no. 1, pp. 190-194, 2006.
7. Loay E. George, and Eman A. Al-Hilo, "Speeding-up Fractal Colored Image Compression Using Moments Features", ICCCE08, International Conference on Computer and Communication Engineering, International Islamic University Malaysia (IIUM), Kuala Lumpur, Malaysia, ISBN: 978-1-4244-1691-2, pp. 1303-1307, 2008.
8. Saud Kakil, "Image in Image Hiding System Using Affain Transform", University of Sulaimani/ College of Science, 2009.