التحقق الصوري باستخدام الكسوريات

*** سليمة باجي عبدالله**

الخلاصة

التخويل الصوري هو التحقق من اصالة الصورة بواسطة اكتشاف التغيرات الماكرة. في البحث تم استخدام العلامة المائية الهشة وذلك لتحقيق اغراض التخويل الصوري حيث خلقت عدة علامات مائية باستخدام خاصية التشابه الضمني (Self-Similarity) المتبعة في خورازمية الكسوريات (المستخدمة في ضغط الصور) ، أي ان العلامة المائية المخلوقة مكونة من خصائص الصورة وبالتالي لكل صورة علاماتها المائية الخاصة بها والتي تختلف عن العلامات المائية لاي صورة اخرى والتي من الصعب جداً أكتشافها بها. ومن ثم يتم اختيار أفضل علامة تحوي على افضل توزيع عشوائي للمناطق المتشابهة وغير المتشابهة وذلك من خلال تطبيق عدة اختبارات إحصائية للعشوائية .

بعد اختيار العلامة المائية يتم طمرها بالصورة بمعدل ثنائي واحد بكل قطعة(block) باستخدام LSB وعندها تكون لدينا صورة معلمة. القرار فيما لو ان الصورة تم التلاعب بها ام لا يكون من خلال استخلاص العلامة المطمورة ومقارنتها مع العلامة المعاد خلقها باستخدام خوارزمية الكسوريات أيضا.

في البحث تمكنا من اكتشاف التغيرات التي تحدث للصورة وكذلك تحديد المواقع التي حدث فيها التلاعب. من التطبيق العملي للبحث والنتائج المستخلصة نجد ان أخفاء العلامة المائية بالصورة لم يؤثر على وضوحية الصورة وذلك لان مايتأثر هو ثنائي واحد فقط (LSB) ذات التأثير الاقل.

الكلمات المفتاحية

التخويل – الكسوريات – التشابه الضمني – العلامة المائية–الاختبارات الاحصائية للعشوائية

---

* كلية الرافدين الجامعة

Image Authentication Based on Fractals

*** Salima Baji Abdullah**

Abstract

Image Authentication verifies the originality of an image by detecting malicious manipulations.

In this proposed system used fragile watermarks for image authentication by creating marks based on self-similarities blocks properties of fractals encoding , (used in image compression), The created marks are depend on the characteristics of the image, so each image has its own watermarks which defers from other watermarks of other images therefore its very difficult to discover the watermark.

Then we select the best one which has high randomization of similarity and dissimilarity of blocks through the use of statistical randomization tests. After the mark has been selected we hid it in blocks of image, one bit in each block by using LSB. Result of hiding is the marked image.

The decision on whether an image is altered or not can be made by extracting hidden mark and comparing it with recreated one.

In our research we can detect any change to an image as well as localizing the area that have been altered. From the practical application and the results of the algorithm, the quality of the watermarked image is very high because the watermark effect at most one LSB of one pixel in each block.

Key word

Authentication, fractals, self-similarity, watermarking, statistical randomization test

---

* **AL-RAFIDAIN UNIVERSITY COLLEGE**

1. introduction

The term 'authentication' has a wide range of meanings, as for example that of a specialist that decides whether a piece of art is authentic or not, whether a user can view or download that piece of art and finally the decision as to whether the content of an object is staying intact after its publication or transmission on the internet.

The ability to detect changes to digital image is very important for many applications such as news reporting, medical archiving, or legal usages [19]. More specific overview of a method design with the purpose to detect the alterations in digital image is given in [12].

Another need for image authentication arises in, for example electronic camera where a buyer purchases a digital image from a seller, and then the seller transmits the digital image to the buyer over the network. In the case the buyer wants to ensure that the received image is indeed the genuine image sent by the seller.

An early attempt in the field of image authentication was the method proposed in [22] where the checksums of digital images were calculated and in combination with a seal produced the watermark information that was responsible for the authentication. This work excited the idea of digital image authentication and many researchers approached the problem from different and more sophisticated ways. Another method that is both efficient and an easy to compute, was proposed by Yeung and Mintzer [16]. According to that method a binary logo is embedded in an image in order to detect possible alteration in the image and at the same time provide some information about the image owner.

Multimedia authentication techniques can be classified into three categories: complete authentication, robust authentication, and content authentication. Complete authentication refers to technique that consider the whole piece of multimedia data and do not allow any manipulation [15,16].

2. Watermarking

Multimedia integrity and authenticity can be guaranteed through the use of *digital signatures* and/or watermarks. A *digital signature* is a data string which associates a message (in digital form) with some originating entity [3]. Because the non-manipulation data are like generic messages, many existing message authentication technique can be directly applied. For instance digital signatures can be placed in the LSB of uncompress data, or the header of compressed data. Then, manipulations will be detected because the hash values of the alterd content bits may not match the information in the altered digital signature.

The use of watermarks instead of digital signature typically affords additional functionality by exploiting inherent properties of image content. Examples of such advantage are the capability for localization of manipulations made to the image and the direct embedding of the watermark in the image data[18]. It is worth mentioning that, both digital signatures and authentication watermarks are useful only for establishing the source of the image and detecting manipulations occurring after the signature/watermark has been inserted [8].

Authentication watermarks can be classified as either *fragile* or *semi-fragile*. Fragile watermarks, as the name implies, are designed to identify any alteration of the pixel values. In practice, fragile watermarks may be used for complete authentication[15,16].Semi-fragile watermarks, on the other hand, try to differentiate between content-preserving (nonmalicious) processes, e.g., compression, and malicious manipulations, e.g., removal of objects from a scene. Watermarks in this class are designed to withstand content-preserving operations, while detecting any malicious manipulations. A general overview of the digital

watermarking systems and methods is given in [23]. Various algorithms have been proposed for fragile watermarking [6,11,16,19] and semi-fragile watermarking [7,10,20].

3. Fractals

Fractals are a very promising technique for image compression have been successfully applied to the compression of one dimensional signals[13], two dimensional image[4,27], and three dimensions volume[26] by finding a fractal representation that models the original data as closely as possible, and storing the model instead of the original data. However, it has not been widely used because of its high computational complexity.

A practical block-based fractal image coding algorithm was first proposed by Jacquin and has received a great deal of attention as a new promising image compression technique. Fractal image coding is different from conventional image coding method and can obtain potential high compression ratio with acceptable image quality. It is based on iterated function system (IFS)[14] which successfully exploits the block wise self affine similarities between different parts of an image. Through affine transformation one sub-block of the original image can be approximated by another sub-block in the same image to some extent estimated by distortion measure. If the approximation error is small enough, the reconstructed image will be process based on fractal transform.

Barsley [14] suggests that perhaps storing image as collections of transformations could lead to image compression. His argument went from possibility to describe the image as a few parameters of affine transformation. If we have parameter of affine transformation we can produce the image. The machine produces self-similarity images called "Fractal"

3.1 Self-similarity

Sub set of fractals magnified, appear similar or identical to original fractals and to other subsets, this property is called Self-similarity [27]. The Self-similarity is the most important characteristic of fractals, and makes fractals independent of scaling. Thus there is no characteristic size associated with fractals. Figure (1) indicates a few of the self-similarities.

An image of a face in figure (2.a) does not contain self-similarity that can be found in the fractals in figure (1). The image does not appear to contain affine transformations of itself. But, in fact, this image does contain a different sort of self-similarity. Figure (2.b) shows sample regions which are similar at different scales: a portion of shoulder overlaps a region that is almost identical, and a portion of the reflection of the hat in the mirror is similar (after transformation) to a part of hat. The distinction from the kind of self-similarity in figure (2) is that rather than having the image be formed of copies of its whole self (under appropriate affine transformation), here the image will be formed of copies of properly transformed parts of itself. These transformed parts do not fit together, in general, to form an exact copy of the original image, and so we must allow some error in our representation of an image as a set of transformations. This means that the image we encode as a set of transformations will not be an identical copy of the original image but rather an approximation of it[21].

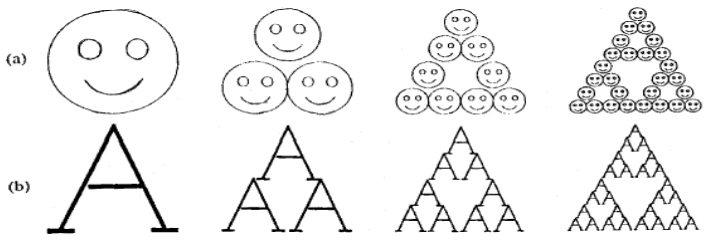

Figure 1:A copy machine that makes three reduced copies (1st copy) of the input image, and feed back 1st copy to copy machine to produce 2st copy and so on.

Figure (2): (a) Original 256×256 pixel Lenna image. (b) Self similar portions of

3.2 Iterated Function System (IFS)

According to the IFS theory, an object with self-similarity and there properties can be represented by a set of contractive mappings. Attractor is the desirable encoding object. However, for image coding, the contractive mapping W which maps the whole image onto itself may not always exist. Instead, the image is partitioned into many small objects and contractive mappings among the small objects are examined. This generalization is called partitioned iterated function system (PIFS) and is applied to image compression [1].

$$W(s) = \bigcup_{i=1}^{n} w_i(s) \ldots\ldots\ldots (1)$$

Real images are not self similar. At best we can find some small regions of a picture that show self similarity see figure(2). However, we can find parts of an image that are similar to other parts of the same image. This leads to the idea of the PIFS[11].

$$wi\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} ai & bi & 0 \\ ci & di & 0 \\ 0 & 0 & si \end{bmatrix} \times \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} ei \\ fi \\ oi \end{bmatrix} \qquad \ldots\ldots\ldots (2)$$

Where s_i represent the contrast, o_i represent the brightness of the transformation. The function w has to be contractive in all three directions: x,y and z. that transformation will be contractive when z distances are shrank by factor less than 1.

Each transform w_i that operate only on a sub-region of the image is referred to as domain block D_i . the image sub-regions to which the domain blocks are mapped are called range blocks R_i**.**

3.3 Affine transformation

An affine maps a plane to itself, mapping domain block into a range block is called affine transformation. It is convenient to separate the transformation given in equation (2) into two transformations. The first, controlling the special part of transformation is [27]

$$v(x,y) = \begin{bmatrix} ai & \text{bi} \\ ci & \text{ai} \end{bmatrix} \begin{bmatrix} \text{x} \\ \text{y} \end{bmatrix} + \begin{bmatrix} ei \\ fi \end{bmatrix} \quad \ldots\ldots (3)$$

The second controls the intensity transformation and is given by

$$u(z) = siz + oi \qquad \ldots\ldots\ldots\ldots (4)$$

Separating the affine transformation in this manner allows us to optimize the transformation for the contrast and brightness coefficients, s and o .Representing the image as a set of transformed sub-region dose not form an exact copy of the original image, but a close approximation of it.

The transformation w_i and domain blocks D_i are chosen such that the error, given by

$$d_{rms}(f,g) = \sqrt{\frac{1}{M \times N} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (f(x,y) - g(x,y)^2} \ldots\ldots\ldots\ldots (5)$$

for two square sub-regions of equal size, d and r, there is an s and an o that minimize [17]

$$S = \sum_{i=1}^{n} (sd_i + o - r_i)^2 .....(6)$$

where S is the squared error, d_i is the i^{th} pixel of d, and r_i is the i^{th} pixel of r. Solving for s and o given [5]

$$s = \frac{n\sum_{i=1}^{n} d_i r_i - \sum_{i=1}^{n} d_i \sum_{i=1}^{n} r_i}{n\sum_{i=1}^{n} d^2{}_i - (\sum_{i=1}^{n} d_i)^2} .....(7)$$

$$o = \frac{1}{n}[\sum_{i=1}^{n} r_i - \sum_{i=1}^{n} d_i] .....(8)$$

For these values of s and o,

$$S = [\sum_{i=1}^{n} r_i^2 + s(s\sum_{i=1}^{n} d_i^2 - 2\sum_{i=1}^{n} d_i r_i + 2o\sum_{i=1}^{n} d_i) + o(on - 2\sum_{i=1}^{n} r_i)].....(9)$$

$$d_{rms} = \sqrt{\frac{S}{n}} .....(10)$$

where *n* = size of blocks

Minimizing d_{rms} for a given domain-range pair in this manner gives the contrast and brightness coefficients for w_i. Once the s and o parameters have been determined, it is clear that the spatial transform coefficients ($a_i, b_i, c_i, d_i, e_i,$ and f_i) are uniquely defined from the size and position of R_i, the size and position of D_i, and one of the eight possible symmetries (see Figure 3) is used to map one region to the other.

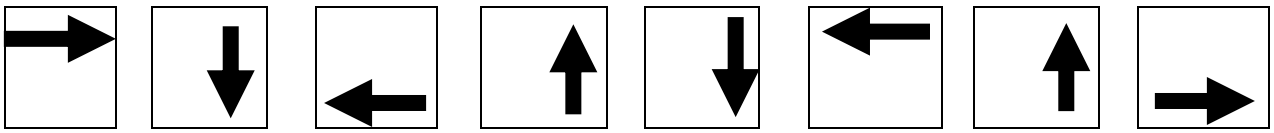

Figure 3: The eight possible symmetries of a square block. The symmetries consist of four 90^o rotations, a flip about the diagonal and four more 90^o rotations. The arrow shows the direction in which the data is scanned.

3.4 Basic Fractal Image Coding Algorithm

Fractal Image Coding is composed of three parts:

- Range block partition : The original image is partitioned into non-overlapping sub-blocks called rang blocks. The range blocks of size BxB ,R_k
- Domain block selection :Parts of original image are chosen to form the searching pool of domain blocks of size DxD (usually equal to 2Bx2B) ,D_k
- Mapping : Domain blocks are mapped to range blocks by an affine transformation as $W_k : D_k \to R_k$

The potential domain blocks can be designed by sliding a DxD window across the original image with horizontal and vertical step size of Δh and Δv respectively which typically equal to B or B/2.If the image is $M \times M$, then the number of such block are

$$(\frac{M-2B}{K} + 1) \times (\frac{M-2B}{K} + 1) \quad ..... \quad (11)$$

4. Statistical Tests

These tests check a random number stream for the uniformity of the stream and for correlations between numbers in the stream. If a given sequence was able to pass all of these tests within a given degree of significance (generally 5%), then it was judged to be, in their words "locally random". Kendall and Smith differentiated "local randomness" from "true randomness" in that many sequences generated with truly random *methods* might not display "local randomness" to a given degree — *very* large sequences might contain many rows of a single digit. This might be "random" on the scale of the entire sequence, but in a smaller block it would not be "random" (it would not pass their tests), and would be useless for a number of statistical applications.The following describes some of the tests[2].

- **Frequency (Monobits) Test**: The purpose of this test is to determine whether that number of ones and zeros in a sequence are approximately the same as would be expected for a truly random sequence. $x^2 = \frac{(n_0 - n_1)^2}{n}$ Where n = length of sequence, n_0 = number of 0's in sequence while n_1 is numbers of 1's. This test is success when $x^2 \le 3.84$
- **serial test**:did the same thing but for sequences of two digits at a time (00, 01,10,11), comparing their observed frequencies with their hypothetical predictions were they equally distributed. $x^2 = \frac{4}{n-1}\sum_{i=0}^{1}\sum_{j=0}^{1}(n_{ij})^2 - \frac{2}{n}\sum_{i=0}^{1}(n_j)^2 + 1)$ Where n = length of sequence , n_{00} = numbers of 00 appear , n_{10} = numbers of 10, n_{01} = numbers of 01 , n_{11} = numbers of 11 appear in sequence. This test is success when $x^2 \le 5.99$
- **Runs Test :**Where a run is an uninterrupted sequence of identical bits. A run of length k means that a run consists of exactly k identical bits and is bounded before and after with a bit of the opposite value. The purpose of the runs test is to determine whether the number of runs of ones and zeros of various lengths is as expected for a random sequence. In particular, this test determines whether the oscillation between such substrings is too fast or too slow. $t_0 = [\sum_{i=1}^{r_0}(r_{0i} - n/2^{2+i})^2 \times 2^{2+i}]/n$ $t_1 = [\sum_{i=1}^{n}(r_{1i} - n/2^{2+i})^2 \times 2^{2+i}]/n$

  r_{0i} is the run of zeros of length I ,while r_{1i} is the run of ones of length i. this test is success when half of run is (0's or 1's) of length 1 and quarter of length 2 and so on. For more details about thes tests and and about auto correlation and poker tests see[2,9].

5.Proposed system

In the proposed system, the fractal encoding technique as a tool to create marks.

5.1 sender

The general block diagram of the proposed system which create and embed mark is shown in figure (4).The search of a range block in all domain block and calculate the contrast, brightness and mean square error(s,o,MSE) is accomplished by using the equation(7,8,10) each time. When MSE is less than or equal to the predetermine minimum error between domain block and range block we add '1' to the mark else we add '0'. After creating mark we add new record to database file and save the mark and its range number in filed mark and rang_no, see figure (5) which shows structure of database file.

At end of matching all range block we get many marks one correspond each range block, the length of each mark is equal to the number of domain blocks, one bit correspond each domain block see the flowchart in figure (6) .

Then we apply statistical randomization tests to select best mark that have high randomized. We give 20 degree for each pass of test, so the total of passing all tests we put it in field, Degree_of_pass .

After selecting the best mark we select the range number that correspond to the selected mark and then hide them in image, 1 bit in the LSB of first pixel of each domain block, The process above is described in details in the following algorithms**,** where algorithem1 describe how to create marks from image, algorithem2 describe the selecting of best mark that have high degree of randomization depend on passing the statistical tests, while algorithem3 describes how to perform hiding of the selected mark and its range number in image.

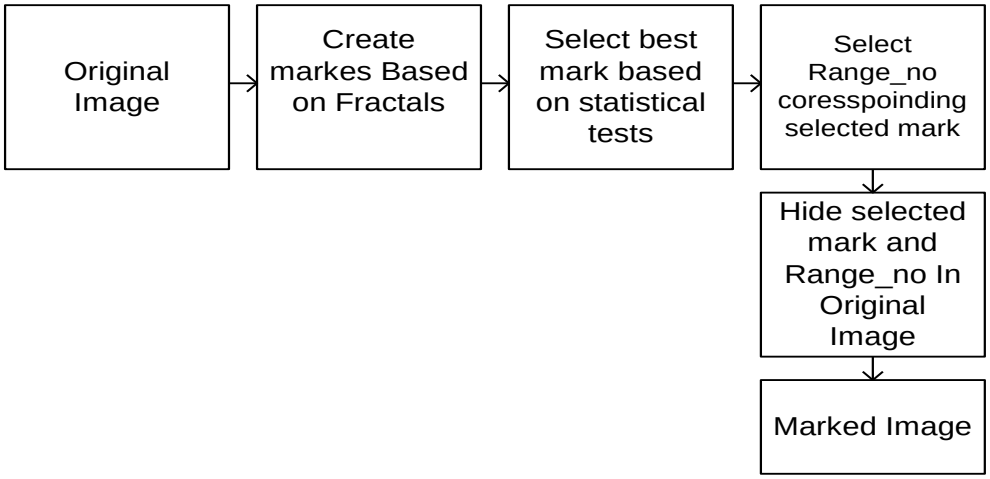

Figure (4) : General block diagram of sender

Field name	Type
Range_no	Numeric
mark	String
Serial	Numeric
frequency	Numeric
run	Numeric
poker	Numeric
Auto	Numeric
Degree_of_pass	Numeric

Figure (5) : structure of data base

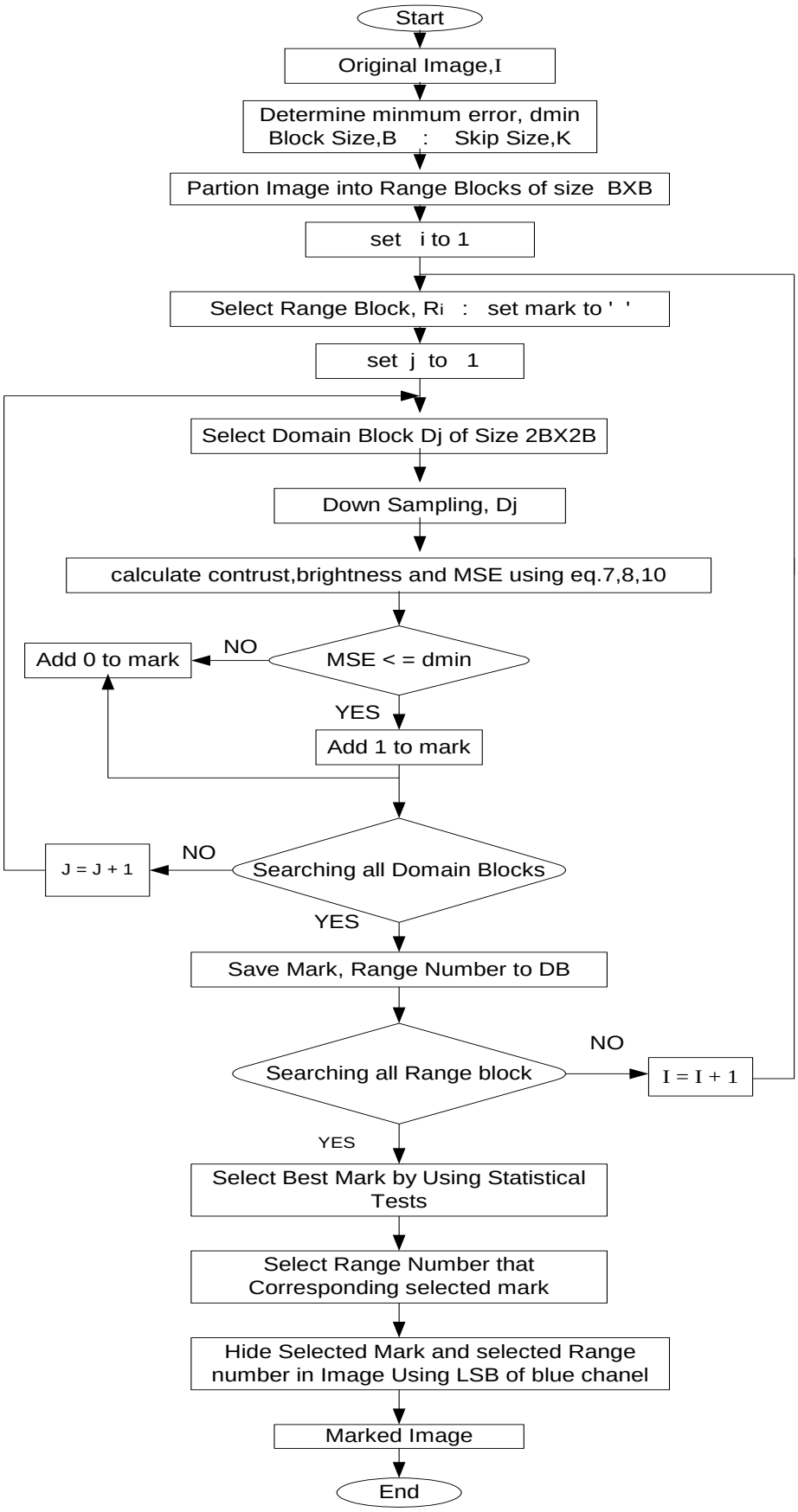

Figure (6): flowchart of sender of proposed system

Algorithm 1 (create marks)

Input	• Image(I) to be authenticate • Approximate minimum errors between range blocks and domain block, d_{min} • skip size,k • block size, B
Output	Data base file contain numbers of created marks equal to no. of range blocks in the image (I)

	Partition I into Rang blocks of size BxB
	D = no. of Domain blocks , using equation (11)
	R = no. of Rang blocks = imag_size/(block size * block size)
	For i = 1 to R
	Begin
	$Mark_i$ = ' ' /* clear mark */
	For j = 1 to D
	Begin
	Select (domain block, D_j of size 2Bx2B)
	Down sampling (D_j)
	Using equation (7,8,10) to
	calculate S,O, MSE(between D_j and Rang block, R_j)
	If MSE < d_{min} then
	add '1' to $Mark_i$
	Else
	add 'o' to $mark_i$
	End
	Add new record to D.B. File
	Set field of mark to $Mark_i$
	Set field of rang_no to i which is the number of R_i
	End

Algorithm-2 (select best marks)

Input	Data base file contain numbers of marks (from algorithm-1)
Output	• One mark • range no. which Corresponding of the selected mark
	n = no. of marks = no. of records = no. of range blocks
	For i = 1 to n begin Calculate statistical tests of randomize (frequency, run, serial, poker, autocorrelation) of $mark_i$ which lies in field mark of recod i /* give 20 degree for each pass test , and replace field Degree_of_pass with total values of tests */ end
	Search for record that have high randomization test, field Degree_of_pass
	Return mark which have high randomization test Return Range no. corresponding to the select mark

Algorithm-3(perform hide mark)

Input	• Image (I) to be authenticate • *Mark* (get from algorithm-2) • *Range number,r* (get from algorithm-2) • Block size and Skip size
Output	Market image

	D = no. of Domain blocks , using equation (11)
	i = 1
	For j = 1 to D
	Begin
	Selected domain block, D_j
	/* Selected one bit at sequence i from mark */
	Get 1bit from $mark_i$ and hide it in blue channel
	LSB of first pixel of domain block D_j
	i = i + 1
	end
	t = convert decimal to binary (*r*)+*'00000000'*
	L = length(t) : j = 1 : i = 1
	While L > 0 do
	Get 1bit from t_i and hide it in blue channel
	LSB of last pixel of domain block D_j
	i = i + 1 /* to get next bit to hide */
	L = L – 1
	j = j + 1 /* to get next domain block */
	Enddo

5.2 receiver

The general block diagram of the receiver is illustrated in figure(7). After receiving the image we extracted hidden mark one bit from the first pixel of each domain block and then extracted hidden rang number one bit from last pixel of each domain block until reaching the end mark '00000000', the stopping indicator, see the flowchart in figure(8) for more details.

Then we convert the binary number of range number to decimal and from that range number we create new mark . after that we compare extracted mark with the new recreated, if they are the same then this image is authenticate otherwise it is not. The process above is described in details in algorithms 4 for extracting hidden mark and range number and algorithms 4 for checking the authenticity.

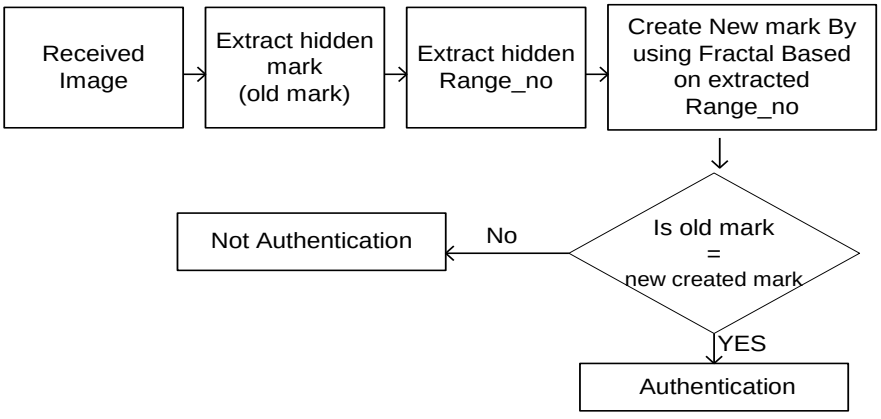

Figure (7) : General block diagram of receiver

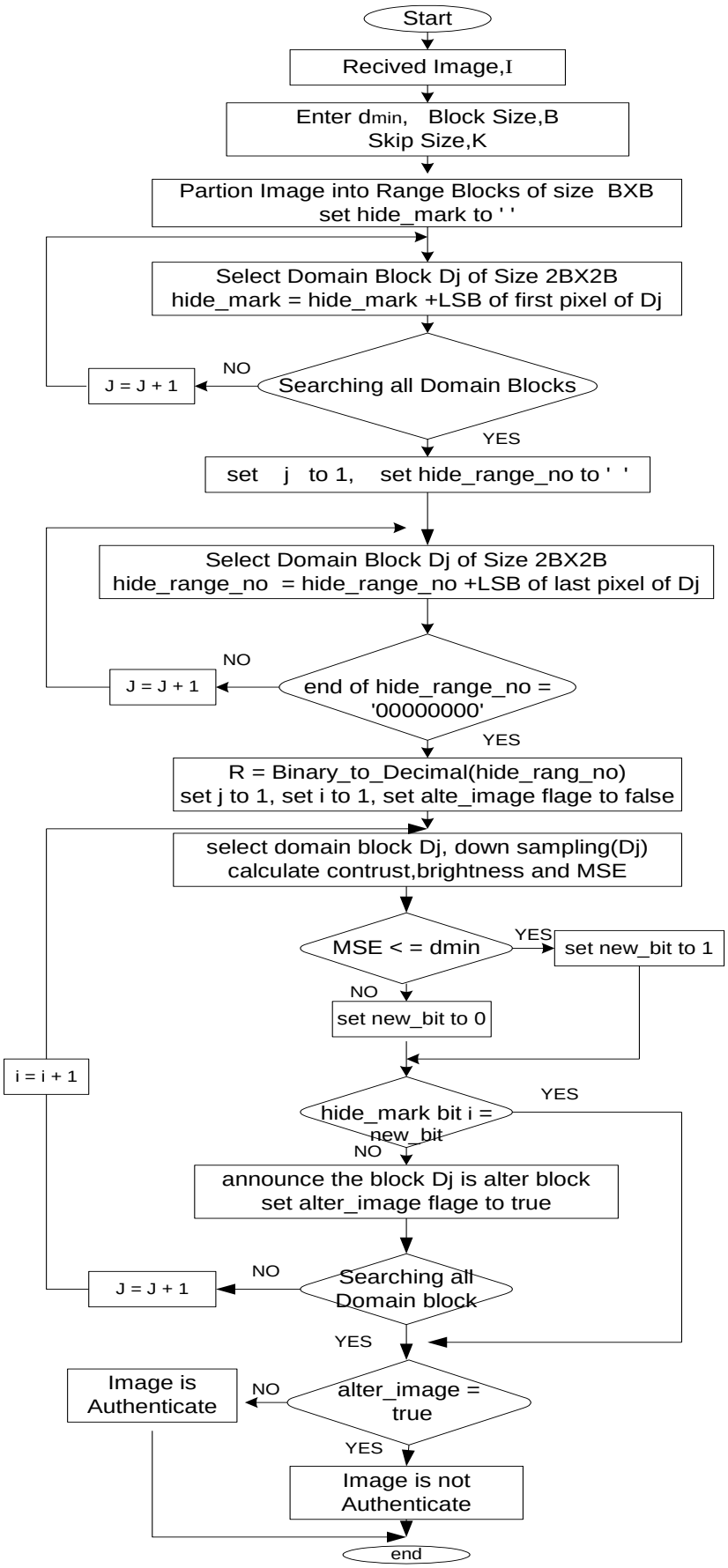

Figure (8): flowchart of receiver of proposed system

Algorithm-4(extracted mark and its range no.)

Input	• Image(I) to be check • Approximate minimum errors between range blocks and domain block, d_{min} • Skip size,k • Block size
Output	• Mark that hidden • Range number that hidden

	Partition I into Rang blocks of size BxB
	D = no. of Domain blocks using equation(11)
	/* extract hidden mark */
	Mark = ' ' : R = 1
	For j = 1 to D
	Begin
	Mark=Mark + 1 bit from LSB of blue channel of fist pixel of domain block D_j
	End
	/* extract hidden range number */
	R_on = ' ' : i = 1 : j = 1
	While R_no not end with *'00000000'* do
	R_no= R_no + 1 bit from LSB of blue channel of last pixel of domain block $D_{,j}$
	j = j + 1
	Enddo
	L = length(R_no)
	B_mark = substr(r_no,1,L-8)
	Range-no = convert binary to decimal (B_mark)

Algorithm-5(checking for authenticity)

Input	• Image(I) to be check • Approximate minimum errors between range blocks and domain block(d_{min}) • Skip size (k) • Mark get from algorithm-4 • Range_no, r get from algorithm-4
Output	Announce the received image is authentication or not
	Partition I into Rang blocks of size BxB D = no. of Domain blocks using equation(11) Set alter_image flag to false Set i to 1 /* bit position of mark */ For j = 1 to D Begin Set alter_block flag to false Select domain block(D_j)of size 2Bx2B Down sampling (D_j) compute equation (7,8,10) to get contrast, brightness,MSE (between D_j and Rang block, r_i) If MSE <= d_{min} and $Mark_i$ = 0 then set alter_block to true

	Else
	If MSE > d_{min} and $Mark_i$ = 1 then set alter_block to true
	If alter_block is true then
	Begin
	announce the domain D_j is alter
	Set alter_image flag to true
	End
	Set i to i + 1 / * to get next bit and check next domain block */
	Endfor
	if alter_image then announce image is not authenticate
	Else announce image is authenticate

5. Experimental Results

The tests are performed by taking a 256*256 true color Image for several skip size and block size of image, that produce different length of marks. The mean square error metric and PSNR are used to measure the visual distortion via the following equation

$$PSNR\,(\text{dB}) = 10\ \log_{10}\frac{255^2}{MSE}$$

which commonly used as pixel-based visual distortion metric[23] and this was used to measure the distortion between the original image and the watermarked image .

The quality of the marked image is very high because the watermark information affected at most one LSB's. The experimental result give PSNR measurements much higher than 40dB. These are considered as very acceptable, since the casual observer cannot notice any visual differences between the original image and the watermark image.

Size of block	Skip size	Size of R_block	No. R_block	Size of D_block	No. D_block	No. of similar blocks	No. of dissimilar blocks	Statistical test	PSNR
2	2	2×2	16384	4×4	16129	8018	8119	Very good	18.2
4	4	4×4	4096	8×8	3969	1050	2919	Good	25.7
8	8	8×8	1024	16×16	961	100	861	bad	42.5

6. Discussion and Conclusions

In this research the fractal encoding for compression is presented also we present some of statistical tests. In our proposed system we use fractals as a tool to create marks. Each image have its watermark depending on its characteristic, the quality of watermark image is remaining very good since the method affects not more one of the least significant bits of the blue channel. The security of the proposed system resides in 1) Approximate minimum errors,2) Block size and 3) Skip size.

For more security, we can encrypt the selected created mark before hidden it by using symmetric/asymmetric key , for more details see[3,25].

When the block size equal 2, the statistical test is very good but the length of mark is too long which lead to less value of PSNR, So we can use compress to improve the PSNR.

When the block size equal 8,we get best PSNR. Since the number of dissimilarity blocks (0) is more than similarity blocks (1), So the stream of bit is sparse string which can be compressed very efficiently, therefore we can compress it using oring Bits, variable-size code or Huffman coding see[24].

REFERENCES

1	”*Image Data compression*”, http://www.iee.et.tudresden ,1997.
2	"*Statistical-Randomness*"', http://www.enwikipedia/wiki
3	A. Menezes, P. van Oorchot, and S. Vanstone, *Handbook of Applied Cryptography*. Boca Raton, FL: CRC, 1997.
4	A.E. Jacquin."*Image coding based on fractal theory of iterated contractive image transformation* ". IEEE Transactions on Image Processing, 18-30, Jan. 1992
5	B. Grinstead, ”*Content-Based Compression of Mammograms*”, master thesis ,Chicago University of Electrical Engineering, 2001.
6	C. W. Wu, D. Coppersmith, F. C. Mintzer, C. P. Tresser, and M. M. Yeung, “*Fragile imperceptible digital watermark with privacy control*,”Proc. SPIE, *Security and Watermarking of Multimedia Contents I*, vol. 3657, Jan. 1999.
7	D. Kundur and D. Hatzinakos, “*Digital watermarking for telltale tamperproofing and authentication*,” *Proc. IEEE*, vol. 87, pp. 1167–1180, July 1999.
8	G. L. Friedman, “*The trustworthy digital camera: Restoring credibility to the photographic image*,” *IEEE Trans. Consumer Electron.*, vol. 39, pp. 905–910, Nov. 1993.
9	G.carter "Statistical Testes For Randomnoss",EISS,England,1989
10	J. Eggers and B. Girod, “*Blind watermarking applied to image authentication*,” in *Proc. IEEE ICASSP*, Salt Lake City, UT, May 2001.
11	J. Fridrich, M. Goljan, and A. C. Baldoza, “*New fragile authentication watermark for images*,” in *Proc. IEEE Int. Conf. Image Processing*, Vancouver, BC, Canada, Sept. 10–13, 2000.
12	J.Fridrich, "*Methods for temper Detection in Digital Image*", multimedia and security workshop at ACM multimedia 99, Orlando,FL,Usa,Oct.1999
13	M. F. Barnsey, J. H. Elton, and D.P. Hardin Recurrent "*Iterated function systems"*, Constructive approximation 1989
14	M. F. Barnsley, and S.G. Demko. Iterated function schemes and the gobal construction of fractals. In proceedings of the Royal society A399,pp. 243-275,1985
15	M. Wu and B. Liu, "*Watermarking For Image Authentication*", IEEE Proc. Of ICIP, Chicago, Oct. 1998.
16	M. Yeung and F. Mintzer, " *an invisible watermarking technique for image verification*", IEEE proc. ICIP, 97. Santa Barbara, California,1997.
17	N.Henk,”*Fractal Image Compression*”, http://www.Cwi.h/Cwi/projects/fractals, 1997.
18	P. Lamy, J. Martinho, T. Rosa, and M. Paula,”*Content-based Watermarking for Image Authentication*”, A.pfitzmann (Ed):IH’99.LNCS 1768, pp 187-198,2000.
19	P. W. Wong, “*A public key watermark for image verification and authentication*,” in *Proc. IEEE Int. Conf. Image Processing*, Chicago, IL, October 4–7, 1998, pp. 425–429.
20	S. Bhattacharjee and M. Kutter, “*Compression tolerant image authentication*,” in *Proc. IEEE Int. Conf. Image Processing*, Chicago, IL, Oct. 1998.
21	S.kumar,”*An Iintroduction To Image Compression*”,athttp://www.debugmode.com,22Oct 2001
22	S. Walton "*Image Authentication For A Sippery New Age*" Dr. Dobb's Journal of software tools for professional programmers, vol.20, Apr. 1995
23	S.Katzenbeisser and F.A.P. Petitcolas "*information hiding techniques for steganography and digital watermarking*" , Artech House,2000
24	Salomon D. ,*'Data Compression',Hamilto*,USA,1997
25	Seberry J. and piprzk J., "*cryptography: an introduction to computer security*", Prentice Hall,1988
26	Wayne O. Cachran, John C. Hart, and Patrick J. Flynn. *"Fractal Volume compression*", In IEEE Transactions on Visualization and computer Graphics 313-322, 1995
27	Yuval Fisher "*Fractal Image compression Theory and Application"*,Springer-Verlag, ISBN,1995