Wavelet Fractal Image Watermarking System (WFIWS)

Itimad Raheem Ali[1,]*, Ghazali Sulong[2]
[1,2]UTM VicubeLab, Faculty of Computer Science and Information Systems,Universiti Teknologi Malaysia,Johor Bahru, Malaysia

Abstract

In this paper, a robust watermarking method is presented. Using the discrete Haar wavelet transform, the proposed method exploits the large coefficients high frequency subband information to embed the invisible fractal watermarked image. The fractal technique is adopted for information preservation from unintended and malicious users. For a still color image, the fractals are created through the midpoint displacement method. By so doing, the original image is utilized for watermarking. The efficacy of the proposed method is in the fact that under JPEG2000 compression distortions, efficient watermarking is still achieved. For a secure and imperceptible watermarking, a fractal approach of watermark embedding is proposed and extracted from the original image. A fractal approach of watermark embedding is proposed and extracted from the original image under the degree of compression that makes the size of image decrease into (1:3). These efficacies of the proposed method make it appropriate to use like owner identification, proof of ownership, and transactional watermarks.

***Index Terms*:** Watermark, Fractal, Wavelet Transform, JPEG2000.

I. INTRODUCTION

Due to the tremendous growth in the sharing of digital multimedia files such as; still images, audio files, video files, and text files, over the Internet there is an increasing concern for copyright protection. The Internet end user cannot be specifically defined, like in the case of sending out a patient's document across the Internet to a particular user (let's say a field doctor), supposing that along the path to the document destination a malicious attack diverts the document to an unintended destination, the confidentiality of the document can be infiltrated. In such a case, the doctor might issue wrong diagnoses based on the altered medical information received. The advantages of the Internet in the aspect of accessibility, usage and fastness, makes it an acceptable means of sending and receiving information. However, this issue of unintended users having access to confidential files threatens copyright protection [1]. In most of the cases of malicious attacks on digital online files, the modifications made to the files are undetectable. Like a conflict scenario between two countries, a change in the video signal transmitted, where the video signal was tampered with by a malicious attack to indicate the imminence of a harmful object to a country's border, upon receiving that signal, by the opposing country, the implications of the misinformation might result to war between the two countries. Hence, there is the need to ensure that the confidentiality of vital documents traveling across the Internet, from one user to another, is protected.

Cryptography is one of the methods used for copyright protection. It is a technique for ensuring the protection of digital files. The files are secured by secrecy; the secret message is scrambled to the message in such a way that the message cannot be deciphered. However, the shortcoming of cryptography method is that, the supposed secrecy can be easily intercepted due to the obviousness of the encrypted message, which attracts suspicion on the document. For a military communication scenario, this unhidden message can be a problem [2]. In this paper, a robust digital watermarking is proposed, which is the process of inserting a digital signal or pattern into digital content [3], for digital files protection. In the proposed method the image is first divided into fractal by a midpoint displacement method and then by using the discrete Haar wavelet transform, the large coefficients high frequency subband information of the cover image is used to embed the invisible fractal watermark.

II. RELATED WORK

Over the years, various watermarking methods have been proposed. Q.M. Jonathan Wu, proposed a novel biometrics watermarking techniques using newly proposed mathematical transform namely, the fractional dual tree complex wavelet transform (FrDT-CWT) and singular value decomposition (SVD). This technique generated keys in the embedding process of a gray-scale watermark. The host image was first randomized by Hessenberg decomposition and a chaotic map followed by embedding in the FrDT-CWT domain by modifying the singular value of the randomized image. This technique confirmed the high security, efficiency and robustness in the security, attack and comparative analysis. The main benefit of this technique is that no one can obtain the keys without the information of the biometrics of the owner/user used. Another benefit is that it is not possible for a person to lose biometrics, and the biometrics information is difficult to falsify for stealing. Hence, the keys become unique, untraceable, and secure [4]. The contribution of Claudia Feregrino U., decreased in the distortion generated by the watermark embedding in the host image. This technique described a digital watermarking scheme based on fractal codification for 8-bit gray scale images; according to the watermark bit being embedded, it replaces range blocks by modified blocks.

The technique achieved a better robustness against JPEG attacks, decreased at 13.2 db in distortion and up to 50% improvement in Bit Correct Ration (BCR). The disadvantage of the utilization of embedding regions is that, the scheme may lose synchronization of the regions a geometric attack, adaptive disposition of blocks can be used to reach robustness against them [5]. Hazem, introduced a blind watermarking technique based on the wavelet-trees. This technique deals with the color pixel as one unit and exploits the significant features and relations between the color pixel components in the wavelet domain. The watermark is embedded by spreading it through the host image in such a manner that the inter-pixel robust relations carry the watermark bit sign with sufficient energy [6]. Gao proposed a novel reversible watermarking algorithm based on a chaotic system. This technique used chaotic system not only to search space of reversibility of the scheme, but also used to randomly select the position of watermarking embedding. This technique had stronger robustness against image compression [7]. Gaurav Bhatnagar, presented a scheme for dealing with the situation where the size of watermark is very big. Two watermarks are embedded in the host image to prevent ambiguity and enhance the security, where both the watermarks are visually meaningful image instead of Gaussian sequences. Moreover no attracter can extract the data without accessing the original image, and then the security of the proposed method lies in the original image [8]. To improve watermarking technology based on discrete wavelet transform and chaos theory was proposed by Zhang. This algorithm was effective and robust to common image processing operations and some geometric operation such as; JPEG compression, JPEG2000 compression filtering, Adding Gaussian noise and so on [9]. Ali Kadhim, came up with an approach that does not require the use of the original image for watermark extraction, they adopted the discrete wavelet transform features. Their method achieved robustness to compression ratio of 1:19 for JPEG2000 compressed image [1].

III. FRACTALS

Broken is the meaning of the word fractal which comes from the Latin adjective fractus. A more scientific approach of the term fractal is that it is a fragmented mathematical geometric shape that can be subdivided into smaller pieces that represent a reduced-size copy of the whole that is in contrast to fractal objects such as, mountains and coastline[10].

“Fractals are objects of any kind whose spatial form is nowhere smooth, hence termed "irregular", and whose irregularity repeats itself geometrically across many scales”, and the fractal as the B. Mandelbrot definition, is a rough or fragmented geometric shape that can be subdivided in parts, each of which is (at least approximately) a reduced/size copy of the whole. Mathematically, the fractal is a set of points whose fractal dimension exceeds its topological dimension. It is common in nature to notice objects that do not have well-defined geometric shapes, but appear to be constructed according to some simple mathematical rules, examples are found in mountains, coastline, volcanic soil, seashells, plant and clouds [11].

A. Random Midpoint Displacement Method

Random midpoint displacement method was introduced by Fouriner et al [12]. The mathematical, non-linear and linear fractals presented above are deterministic, that means repeating the transformations under the same starting conditions will always result in the same figure. The midpoint displacement method, however, belongs to the category of random fractals, such as, the fractals generated by the DLA-method, which in general produce more nature-like "random" objects. Then the principle of work of this project was presented as follows:

An initial square is subdivided into four smaller squares. Let us have four points $[x_0, y_0, f(x_{0,}y_0)],[x_1, y_0, f(x_1, y_0)],$ $[x_0, y_1, f(x_{0,}y_1)],[x_1, y_1, f(x_1, y_1)]$. In the first step, one vertex is added into the middle. The vertex is denoted by $\lfloor x_{1/2}, y_{1/2}, f(x_{1/2}, y_{1/2}) \rfloor$, where

$$x_{1/2} = \frac{1}{2}(x_0 + x_1)$$

$$y_{1/2} = \frac{1}{2}(y_0 + y_1)$$

$$f(x_{1/2}, y_{1/2}) = \frac{1}{4}(f(x_0, y_0) + f(x_1, y_0) + f(x_0, y_1) + f(x_1, y_1)) \quad .... (3.1.1)$$

This procedure is recursively repeated for each subsquare, then for each one of their descendants, and so on. The random number must be generated with Gaussian distribution [μ=0, σ=1] and in the i-th iteration step the variation σi has to be modified according to

$$\sigma^2{}_i = \frac{1}{2^{2H(i+1)}}\sigma^2 \qquad .... (3.1.2)$$

Where, H denotes Hurst exponent (1≤H≤2). From equation (3.1.2) it can be seen, that the first iteration has the biggest influence on the resulting shape of the surface and its influence on the others decreases.

In the second step, the points on the edges of Initial Square are calculated. Square by square is virtually rotated and the values are calculated as in the previous step.

In the next step, the square is virtually rotated back by 〖45〗^0and the first two steps are recursively applied on the four new squares, as mentioned above. This recursive process ends after a given number of iterations [11]. Fractal dimension D of surface is obtained by

D=3-H

An example of fractal terrain obtained with random midpoint displacement algorithm is shown in Figure 1.

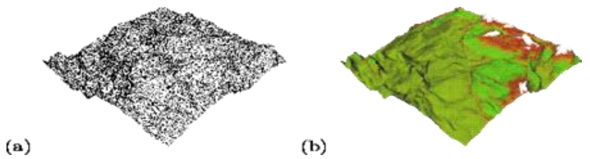


Fig. 1. Example of fractal terrain (a) Wire frame model (b) the same model textured

IV. PROPOSED WAVELET FRACTAL IMAGE WATERMARKING SYSTEM (WFIWS)

A. The Proposed Digital Watermarking Algorithm

The steps of the proposed watermarking algorithm consist of two modules:

a. Embedding Module.

b. Extraction Module.

The embedding module is used for embedding fractal true color image (watermark) inside a still cover true color image, then the result of this process is watermarked image. The overall operations of embedding and implementing algorithms are described in section (4.2). The extraction module is used for extracting fractal image from watermarked image. The whole processes are described in section (4.3). Sometimes, watermarking image might be attacked and effected in a digital method such as: compression JPEG, filters or added noise ... etc. So the proposed method has the ability to resist the attack and extract the fractal image from the watermarked image. The proposed watermarking algorithm consists of eleven stages represented in the scheme, as shown in Figure 2 and the stages algorithm:

Stage1:Generate the fractal image by a Midpoint Displacement Method at size (64×64).

Stage2: Convert the generated fractal image into a sequence of bits (0 & 1).

Stage3: Transform the sequence of bits into 2-level (± Factor).

Stage4: Select a cover image of type BMP.

Stage5: Apply Haar wavelet transform on to the cover image.

Stage6: Add the 2-level of fractal image into the coefficients of cover image.

Stage7: Apply the inverse Haar wavelet transform.

Stage8: Display watermarking image.

Stage9: Attack by JPEG2000 compression watermarking image, but the fractal image should not be affected

Stage 10: Extract the watermark image.

Stage 11: Display the fractal image.

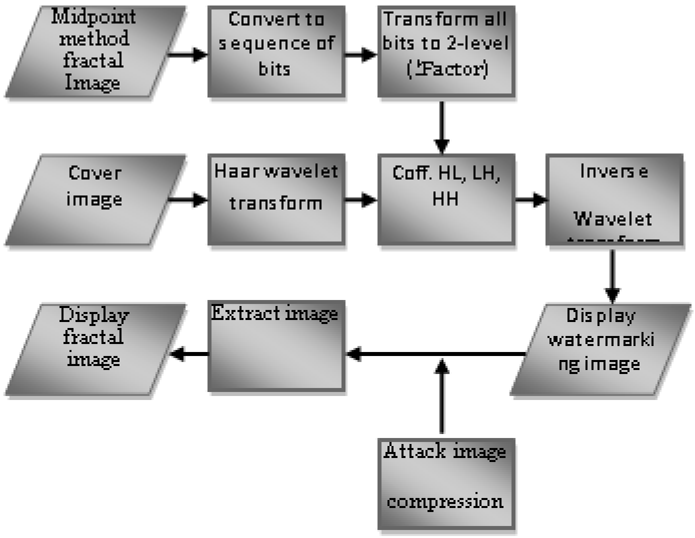

Midpoint method fractal Image
Convert to sequence of bits
Transform all bits to 2-level (±Factor)
Cover image
Haar wavelet transform
Coff. HL, LH, HH
Inverse Wavelet transform
Display fractal image
Extract image
Display watermarking image
Attack image compression

Fig. 2. General block diagram of (embedding and extracting) modules

B. Watermark Generation using Random Midpoint Displacement Method

This method is used to generate the fractal image used as a watermark; the principles of this method are as follows:

Step1: Apply Gaussian Random Generation algorithm to find values of all points in fractal image below:

Function GetRandomGeneraion Sum = 0; for I = 0 to 11 Sum = Sum + Rnd; Next I;

GetRandomGeneraion = (Sum/12-0.5)*12.00014555*Val;

End Function

Step2: An initial square of image, as shown in Figure 3.

F (Xl, Yl) = GetRandomGeneraion(l)

F(X1+W, Yl) = GetRandomGeneraion(l)

F (Xl, Y1+H) = GetRandomGeneraion(l)

F(X1+W, Y1+H) = GetRandomGeneraion(l)

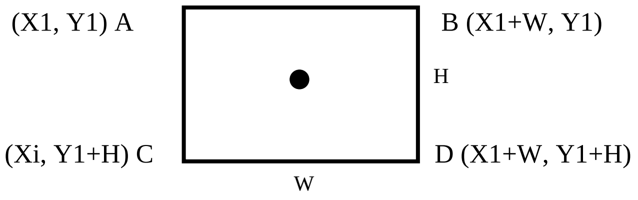


Fig. 3. Represent the middle Point algorithm.

Step3: One vertex is added into the middle. Its vertex is denoted by

Xc= X_1+W/2

Yc= Y_1+H/2

F(Xc, Yc) = (A+B+C+D)/4 + GetRandomGenerator

Step4: Step3 is repeated for each sub square with applied algorithm as below in:

```
While, Sx > 0 And Sy > 0
   Sx = Wt/2:Sy =Ht/2: Ys =0: Ye= Wt: Yc= Sy
      While Ye <= Hm
Xs = 0: Xe = Ht: Xc = Sx
While Xe <= Wm
A=F(Xs,Ys); B=F(Xe,Ys)
C=F(Xs,Ye); D=F(Xe,Ye)
E=(A+B+C+D)/4 + GetRandomGenerator (Std)
```

```
F(Xe,Ye)=E
F(Xe,Ys)=(A+B+E)/3 + GetRandomGenerator (Std)
F(Xe,Ye)=(A+B+E)/3 + GetRandomGenerator (Std)
F(XS,Ye)=(A+B+E)/3 + GetRandomGenerator (Std)
F(Xe,Yc)=(A+B+E)/3 + GetRandomGenerator (Std)
XS= Xe; Xe= Xe +Wt; Xe= Xe +Wt
Wend Ys = Ye; Ye= Ye + Ht; Ye = Ye + Ht
Wend Wt = Sx; Ht = Sy; Std = Std I Sqr(2 A (H * (I + 1)));
 I = I + 1;
Do.
```

Step5: Combine (Red, Green and Blue) components with the header of the image to construct a fractal image.

Step6: Fractal image’s results are displayed in step (5). This image includes:

a- Size equals (64*64).

b- Bitmap (BMP) image format.

c- Type of 24 bit.

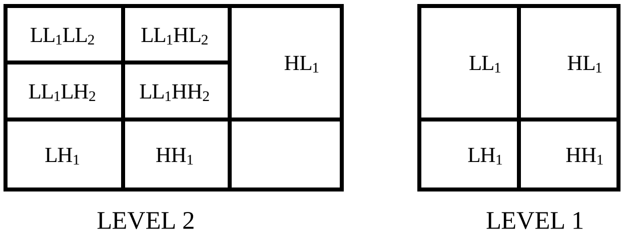


Fig. 4. Represent the coefficients for levels 1 and 2

C. *Embedded Module*

The embedding module is used to hide the fractal image which was constructed from applied Random Midpoint Displacement Algorithm and it is demonstrated by:

a. 24 bit color or true color.

b. Size of fractal image (64*64) pixels.

All bits = 64 * 64 *24 = (98304) bits.

c. The High_Dimensional factor's value is inserted which determines the characterizations’ image.

The embedded module uses the cover image demonstrated by:

a. 24 bit color or true color.

b. Size of Cover image greater than (218*218) pixels.

Generally, cover and fractal image are BMP pictures, which utilize RGB color model. They are divided into three components (Red, Green and Blue).

<u>Algorithmic Steps to construct watermarking image</u>

Step 1: Apply “Random Midpoint displacement method" to construct fractal image (64*64), it is described in section (4.2).

Step 2: Choose cover image and check it, it must satisfy the followings:

a- Bitmap image format (BMP).

b- 24-bit or true color type.

c- Size of image greater than (218*218) pixels.

Step 3: Input and check

Number of Levels (1-3), as in Figure 4.

Watermark image’s Name and it must be bitmap (BMP) format.

Magnitude factor's range is (±(1-50)), it will be minus value when the embedded bit equals '0' and plus value when the embedded bit equals '1'.

Step 4: Convert all components data (Red, green, blue) fractal image to sequence of bits ('0' and '1').

Number bits = width x height x 24
= 64 x 64 x 24
= 98304 bits.

Step5: Red component in first stage is taken from the cover image and execute Haar Wavelet Transform which contains "Low Pass Filter LPF and High Pass Filer HPF".

The results are four sub bands as follows:

a- Approximation image band Low Low (LL) component.

b- Horizontal image band High Low (HL) component.

c- Vertical image band Low High (LH) component.

d- Diagonal image band High High (HH) component.

The principle of the Haar wavelet transform is to calculate the average and the difference for values of neighboring data, the equations and algorithms below are implemented to find all the coefficients (LL, HL, LH and HH), as shown in Figure (5):

```
Int k, D, X, Y;
K =0; For (Y=0; Y<BMP.Height; Y+ 2)
D=0; For (X=0; X<BMP.Width; X+ 2)
D=0;
LI=BUFD[Y][X] + BUFD[Y] [X+l];
L2 BUFD[Y+l][X]+BUFD[Y+l][X+l];
LL[k][d]=(Ll+L2) /4;
LH[k][d]=(LI-L2) /4;
Hl=BUFD[Y][X] - BUFD[Y][X+l];
H2=BUFD [Y+1] [X]-BUFD [Y + 1] [X + 1];
HL[k][d]=(Hl+H2) /4;
HH[k][d]=(HI-H2) / 4 ;
D++;
k++;
```

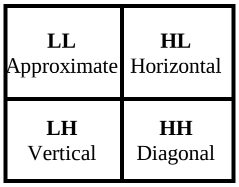


Fig. 5. Refers to all coefficients and concepts.

Step 6: Save bits of one component which resulted from step (4) in coefficients (HL, LH, HR) for the cover image as follows:

a- All bits mapped to values of magnitude factor, where as the bit equals '0' the sign of magnitude of factor is minus, and when it equals '1' sign is plus.

b- Choose the same location in (LH, HL, HR) coefficients for checking and embeddeding 3 bits and this means adding ((_-^+) magnitude factor) to any above coefficients.

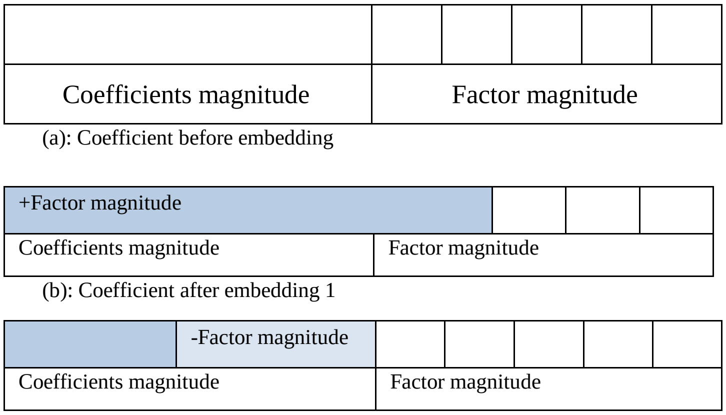
c- As shown in Figure (6), the addition or subtraction magnitude of factor to coefficients according to:

If embedded bit equals '1':

New coefficient = original coefficient + magnitude factor

If embedded bit equals '0':

New coefficient = original coefficient - magnitude factor



(a): Coefficient before embedding

(b): Coefficient after embedding 1

(c): Coefficient after embedding 0

Fig. 6. The wavelet coefficient value before and after adding magnitude factor '0' or '1'.

Step7: Rearrange coefficients (LL, HL, LH and HH) to construct a level image in the form of Gray scale for only Red component.

Step8: Execute Inverse Haar Wavelet Transform to replace Red component of watermarking image after embedding all bits from the essential Red component of fractal image as follows:

a- Distribute the coefficients (LL, HL, LH and HH) according to Figure 7:

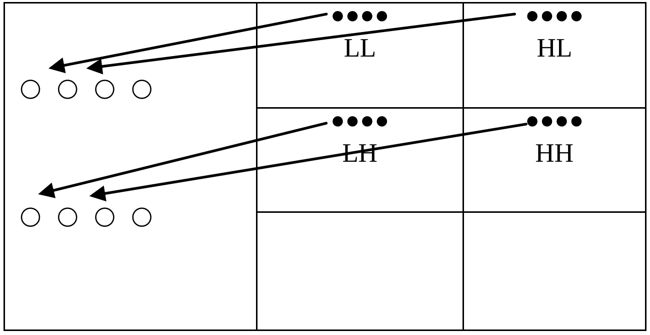


Fig. 7. The position of the set of Permutation the coefficients (LL, HL, LH and HB) in Inverse Haar Wavelet transform.

Get the value of the coefficient (LL) in location (X, Y), (HL) coefficients in location (X+l, Y), (LH) in location (X, Y+l), and (HR) in location (X+l, Y+l) for all data of image.

b- Apply the code below to calculate coefficients (ILL, IHL, ILH and IHH).

```
Int k, D, X, Y;
K=0; For(Y=0; Y<BMP.Height; Y+2)
D=0;
For(X=0; X<BMP.Width; X+2)
D=0;
 Ll=BUFD[Y][X]+BUFD[Y][X+l];
L2=BUFD[Y + 1] [X]+BUFD[Y+1] [X+1];
LL[k][d]=(Ll+L2) ; LH[k][d]=(LI-L2) ;
Hl=BUFD[Y][X] - BUFD[Y] [X+l];
H2=BUFD[Y + 1] [X]-BUFD[Y+1] [X+1];
HL[k] [d]=(Hl +H2) ; HH[k][d]=(HI-H2) ;
D++; k++;
```

c- Repeat step (8-a) to construct and save watermarked image.

Step 9: Repeat steps (6, 7 and 8) for components (Green and Blue) of the cover and the fractal image and construct Gray image of (Green and Blue) components.

Step10: After embedding the fractal image in coefficients of the cover image, combine (Red Green and Blue) components with the header of the image to construct watermarked image.

Step11: Display all images (cover, fractal, level, watermarking) in output.

Flow Diagram of the embedding watermark method

Figure 8 show the flow diagram of the embedding watermarking method.

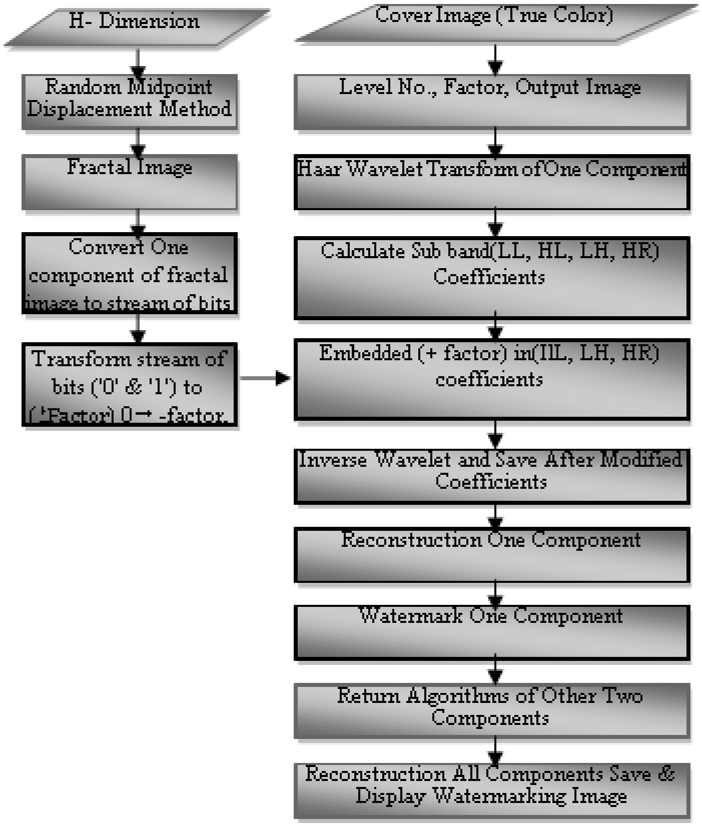

H- Dimension
Random Midpoint Displacement Method
Fractal Image
Convert One component of fractal image to stream of bits
Transform stream of bits ('0' & '1') to ('Factor) 0→ -factor.
Cover Image (True Color)
Level No., Factor, Output Image
Haar Wavelet Transform of One Component
Calculate Sub band(LL, HL, LH, HR) Coefficients
Embedded (+ factor) in(HL, LH, HR) coefficients
Inverse Wavelet and Save After Modified Coefficients
Reconstruction One Component
Watermark One Component
Return Algorithms of Other Two Components
Reconstruction All Components Save & Display Watermarking Image

Fig.8. Flow diagram of embedding watermark method.

D. Extraction Module

This module is used for extracting all data of the fractal image from the coefficients of watermarking image. The process of retrieving the fractal image depends on the calculations of the coefficients of the watermarked and cover image, then find the difference between same coefficient (HL, LH, HH) in the same location and suggest if if it is '0' or '1'. The steps of the watermark image extraction module are shown in Figure 6 and the stages are explained in the following steps:

Algorithm Steps Extraction Module

Step1: Choose and check the watermark and cover image, magnitude factor demonstrated by:

a- Bitmap image format (BMP).

b- 24-bit or true color type.
c- Size of image greater than (218*218).

Step 2: Take the Red component of watermarked image and execute Haar wavelet on the red data, then calculate the coefficients (LL W, HL W, LHW and HHW), it is explained in section (4.3.1, step 5).

Step3: Take the Red component of cover image and execute Haar wavelet on the image, then calculate the coefficients (LLC, HLC, LHC and HHC), it is explained in section (4.3.1, step 5).

Step4: Find the difference between two coefficients of the same location to these bands (IHLW with IHLC, ILHW with ILHC and IHHW with IHHC), when the difference is greater than zero the output bit is '1' else the output bit is '0'; the following code represents the process.

```
EndLoc=0;EE=0;
for(i=0;i<NDD;i++)
for(E=0;E<NWW;E++)
   Diff=ILHC[i][E]-ILHW[i][E];
if(Diff>0)Out[ii][EE]=l; else Out[ii] [EE]=0;
EE++; if(EE>=8) {EE=0; ii++;}
EndLoc++;
Diff=IHLC[i][E]-IHLW[i][E];
if(Diff>0)Out[ii][EE]=1; else Out[ii] [EE]=0;
EE++;if (EE>=8) EE=0;ii++;
EndLoc++;
Diff=IHHC[i][E]-IHHW[i][E];
   if(Diff>0)Out[ii][EE]=l; else Out [ii] [EE]=0;
   EE++;if(EE>=8) EE=0;ii++;
EndLoc++;
if (EndLoc>32768) E=NWW ;
i=NDD;
EndLoc=0;
```

Step 5: The sequence of bits resulted from step 4 are transformed to bytes, which is the reconstructed Red component of the output image.

Step 6: Repeat steps (2, 3, 4, and 5) in order to find the (Green and Blue) components of the output image.

Step 7: Combine the three components Red, Green and Blue with header image to construct output bitmap and true color image (64x64) and display.

Flow Diagram represents Extraction Watermarking Image

Figure 9 shows the flow diagram of extracted watermarking image.

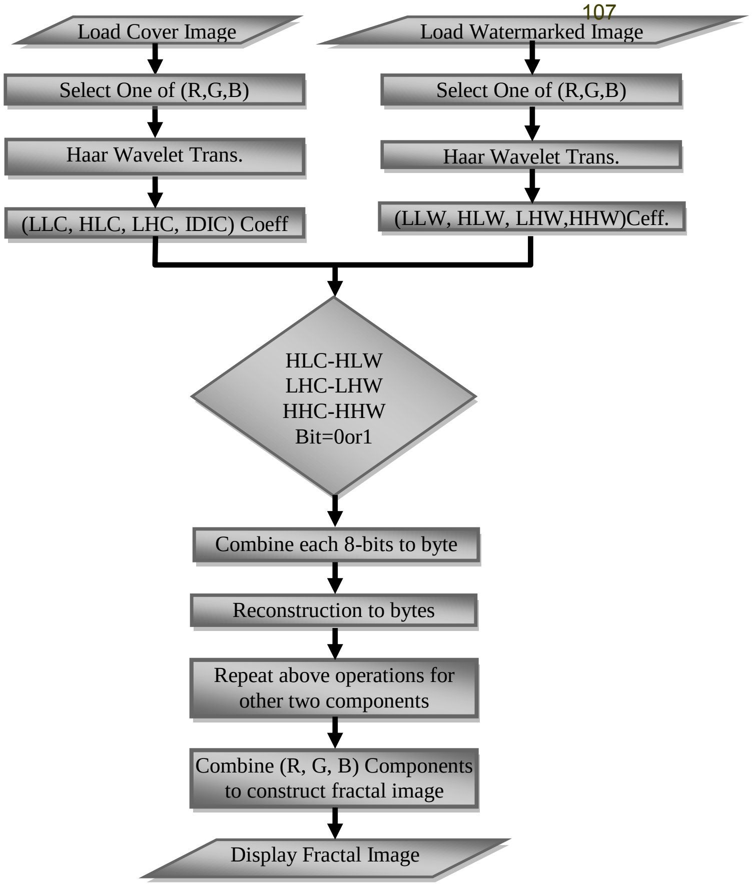


Fig. 9. flow diagram represents extracted watermarking image

V. PRACTICAL INVESTIGATION

Since, the objective is to look for a suitable decomposition scheme and transform type that can satisfy the following:

a- Large number of bits can be embedded.
b- Higher ratio of compression can be applied.
c- Less shift factor.

To investigate the proposed system, it should be tested by using standard measurements such as; Mean Square Error (MSE), Signal to Noise Ratio (SNR), Peak Signal to Noise Ratio (PSNR) and Mean absolute Error (MAE) also measure the number of bits of fractal image after extraction. When the shift factor increases, more degradation occurs in the image, which will increase the value of the MSE and decreases the value of PSNR; these measures were adopted in this study as an objective distortion measure,

$$MSE = \frac{\sum_{x,y}[O(x,y) - R(x,y)]^2}{W \times H} \qquad \text{.... (5.1)}$$

$$PSNR=10\log_{10}\frac{(255)^2}{MSE} \qquad \text{.... (5.2)}$$

When O(x, y) represents the value of the color component of the original image at pixel (x, y), and R(x, y) represents the corresponding value in the watermarked image. W & H represent the width and height of the image respectively. Generally, the values of PSNR above (38) db are visually satisfactory, even for the professionals. The best results occur when the quality of the compressed watermarked image is greater or equal to 88.0, Thr. Represent ±magnitude factor, comp. size represents the size of the image after JPEG2000 compression process, comp. ratio represents the ratio of compression to the tested image and the ratio of different bits of the fractal image that is affected after JPEG2000 compression.

A. Packet Decomposition with Haar Transform

The flow diagrams given in sections (4.3.2) and (4.4.2) show the steps of decomposition and reconstruction of an image using Haar transform. A set of images was tested by these diagrams, and those tested images are shown in Figure 10; tables (1) and (2), showing the results obtained from the system implementation for Lena image and baboon image.

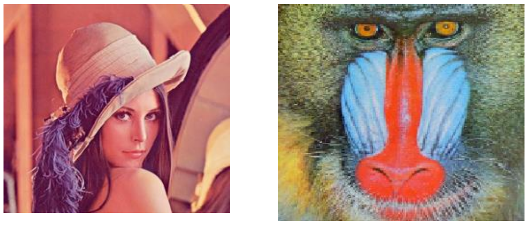


Fig.10. The tested images of type bitmap

TABLE I
LENA .BMP

Thr	Quality	Comp Size	Comp Ratio	MSE	SNR	PSNR	MAE	Ratio of diff. bits
1	100.0	95.1	1/2	0.70716	198.034535	49.6355	0.50008	0.000000
	90.0	95.1	1/2	0.70716	198.034535	49.6355	0.50008	0.000000
	88.0	64.0	1/3	1.30389	107.404350	46.9783	0.97702	0.195597
	87.0	34.4	1/5	2.00680	69.784340	45.1057	1.53925	0.382874
5	100.0	111.9	1/1	3.53473	39.619296	42.6472	2.49928	0.000000
	90.0	111.9	1/1	3.53473	39.619296	42.6472	2.49928	0.000000
	88.0	64.1	1/2	4.03135	34.738640	42.0762	3.22390	0.002228
	87.0	34.8	1/5	4.35143	32.183338	41.7444	3.39402	0.116323
7	100.0	117.6	1/1	4.94777	28.304358	41.1867	3.49809	0.000000
	90.0	117.6	1/1	4.94777	28.304358	41.1867	3.49809	0.000000
	88.0	64.0	1/3	5.43084	25.786733	40.7821	4.28121	0.000671
	87.0	34.7	1/5	5.74552	24.374190	40.537471	4.49355	0.071075
9	100.0	122.2	1/1	6.35976	22.020258	40.0963	4.49605	0.000000
	90.0	122.2	1/1	6.35976	22.020258	40.0963	4.49605	0.000000
	88.0	64.1	1/2	6.88746	20.333126	39.7502	5.40097	0.000224
	87.0	35.0	1/5	7.20292	19.442602	39.5557	5.62553	0.049032
15	100.0	132.2	1/1	10.5827	13.233195	37.8848	7.47817	0.000000
	90.0	132.2	1/1	10.5827	13.233195	37.8848	7.47817	0.000000
	88.0	64.0	1/2	11.0841	12.634599	37.6837	8.56017	0.000102
	87.0	34.9	1/5	11.4459	12.235740	37.5444	8.79596	0.034251

The result of Lena.bmp image

TABLE II
BABOON .BMP

Thr	Quality	Comp Size	Comp Ratio	MSE	SNR	PSNR	MAE	Ratio of diff. bits
1	100.0	126.1	1/1	0.725007	188.354823	49.527383	0.506083	0.008331
	90.0	126.1	1/1	0.725007	188.354823	49.527383	0.506083	0.008331
	88.0	64.0	1/3	2.438895	55.991972	44.258873	1.888250	0.339742
	87.0	34.9	1/5	4.327701	31.554515	41.768231	3.351344	0.446737
5	100.0	133.3	1/1	3.550099	38.466119	42.628399	2.513855	0.000936
	90.0	133.5	1/1	3.550099	38.466119	42.628399	2.513855	0.000936
	88.0	64.1	1/2	4.447118	30.707196	41.650017	3.607646	0.037699
	87.0	34.6	1/5	5.413058	25.227609	40.796377	4.257589	0.263794
7	100.0	137.0	1/1	4.962119	27.520202	41.174132	3.513585	0.000936
	90.0	137.1	1/1	4.962119	27.520202	41.174132	3.513585	0.000936
	88.0	64.0	1/2	5.797819	23.553428	40.498157	4.692128	0.012889
	87.0	34.8	1/5	6.570111	20.784810	39.955076	5.202159	0.192332
9	100.0	140.2	1/1	6.375207	21.420248	40.085861	4.512812	0.000936
	90.0	140.2	1/1	6.375207	21.420248	40.085861	4.512812	0.000936
	88.0	63.9	1/3	7.193615	18.983294	39.561331	5.799413	0.006989
	87.0	34.7	1/5	7.953101	17.170475	39.125439	6.314280	0.147217
15	100.0	148.2	1/1	10.607926	12.873253	37.874499	7.501394	0.000936
	90.0	148.2	1/1	10.607926	12.873253	37.874499	7.501394	0.000936
	88.0	64.1	1/2	11.438070	11.938947	37.547276	9.110357	0.001190
	87.0	34.9	1/5	12.200839	11.192551	37.266907	9.690923	0.072286

the result of baboon.bmp image

VI. THE RESULT STATEMENTS

Here, some "bold statements" are presented, which constitute with the proposed robust watermarking method. These statements are considered to be essential in determining a robust watermarking method, Figure 11 shows as an example, the result of Hiding image, and the level of Haar wavelet transform. The statements are given:

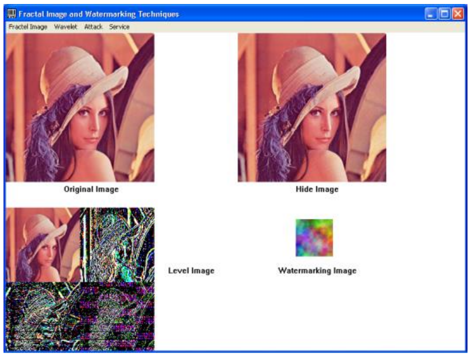


Fig. 11. Hiding image

1- When the value of the shift factor (Thr.) is suitable to the modification with the use of the JPEG2000, a robust watermark ensues.

2- When the shift factor (Thr.) increases, the Peak Signal-to-Noise Ratio (PSNR) decreases, which degrades the watermarked image substantially.

3- With an increased shift factor (Thr.), the quality of the image degrades.

4- Applying Wavelet Haar transform guarantees more bits embedding with modulation factor value that is robust against JPEG2000 compression.

5- Using the Midpoint displacement method, an image of any size can be generated with mean equals to zero and standard deviation equals to one.

6- A method is considered robust until the size of the compressed image decreases by (1:3) of the original image.

7- The bits distortion can occur within a magnitude factor in the range (± (1-50).

8- The range of the magnitude factors must not be greater than or less than the magnitude of the JPEG2000 compression.

VII. Conclusions

A robust watermarking method that utilized the large coefficients high frequency subband information of the discrete Haar wavelet transformed image to embed the invisible fractal watermark is proposed. It is shown, through experiments, that the proposed method can achieve minimal distortions for the watermarked image for a JPEG2000 compression size 64 Kb with a ratio of (1:3) at 88.0 quality, compression size =64.0 Kb when the original size =192.1 Kb then the distorted bits is very little or underprivileged. While the compression ratio increases, the number of survived embedded fractal watermark bits is decreases. The distortion bits are increased when quality and compression size decreases. In subsequent research, the proposed method will be used to determine the modified area of the cover image from the Tamper detection and the robustness of the proposed method against other types of attacks will be evaluated. For future work can use other types of transform and can suggest algorithm to survive other type of the cover image not only (.Bmp) images.

Acknowledgment

This research is supported by the Ministry of Science and Technology (MOSTI) and collaboration with Research Management Center (RMC), Universiti Teknologi Malaysia (UTM). This paper is financial supported by E-Science Grant Vot. No.: R.J130000.7911.4S026.

References

[1] [1] A. Kadhem, “Watermark Applications in Color Images Using Wavelet Transforms,” Baghdad University, 2004.

[2] [2] J. Mohammad, “Watermarking Robustness Algorithms Using Error Correcting Codes on Compressed Image,” Baghdad University, 2009.

[3] [3] T. K. Ingemar J. Cox, Matthew L. Miller, Jeffrey A. Bloom, Jessica Fridrich, Digital Watermarking and Steganography, Second Edi. United States of America: Densise E. M. Penrose, 2008.

[4] [4] G. Bhatnagar and Q. M. Jonathan Wu, “Biometrics inspired watermarking based on a fractional dual tree complex wavelet transform,” Future Generation Computer Systems, vol. 29, no. 1, pp. 182–195, Jan. 2013.

[5] [5] P. A. Hernandez-Avalos, C. Feregrino-Uribe, and R. Cumplido, “Watermarking using similarities based on fractal codification,” Digital Signal Processing, vol. 22, no. 2, pp. 324–336, Mar. 2012.

[6] [6] H. M. Al-Otum and N. A. Samara, “A robust blind color image watermarking based on wavelet-tree bit host difference selection,” Signal Processing, vol. 90, no. 8, pp. 2498–2512, Aug. 2010.

[7] [7] Q. Gu and T. Gao, “A novel reversible robust watermarking algorithm based on chaotic system,” Digital Signal Processing, vol. 23, no. 1, pp. 213–217, Jan. 2013.

[8] [8] G. Bhatnagar, Q. M. J. Wu, and B. Raman, “A new aspect in robust digital watermarking,” Multimedia Tools and Applications, Apr. 2011.

[9] [9] L. Zhang, X. You, and Y. Hu, “Image-Adaptive Watermarking Algorithm to Improve Watermarking Technology for Certain Water Marking Based on DWT and Chaos,” AISC Springer- Verlag Berlin Heidelberg, vol. 104, pp. 43–49, 2011.

[10] [10] N. S. Lam, “Fractal Analysis,” International Encyclopedia of Human Geography, Elsevier Ltd, no. Louisiana State University, Baton Rouge, LA, USA, p. Pages 263–270, 2009.

[11] [11] N. V. V.P. Dimri, R.P. Srivastava, Ed., Fractal Models in Exploration Geophysics Applications to Hydrocarbon Reservoirs. Elsevier B.V., 2012, pp. 1–165.

[12] [12] L. C. A. Fournier, D. Fussel, “Computer Rendering of Stochastic Models,” Communications of the ACM, vol. 25, pp. 371–384, 1982.

[13] E. E. Reber, R. L. Michell, and C. J. Carter, “Oxygen absorption in the Earth’s atmosphere,” Aerospace Corp., Los Angeles, CA, Tech. Rep. TR-0200 (420-46)-3, Nov. 1988.

[14] (Handbook style) *Transmission Systems for Communications,* 3rd ed., Western Electric Co., Winston-Salem, NC, 1985, pp. 44–60.

[15] *Motorola Semiconductor Data Manual,* Motorola Semiconductor Products Inc., Phoenix, AZ, 1989.

[16] (Basic Book/Monograph Online Sources) J. K. Author. (year, month, day). *Title* (edition) [Type of medium]. Volume (issue). Available: http://www.(URL)

[17] J. Jones. (1991, May 10). Networks (2nd ed.) [Online]. Available: http://www.atm.com

[18] (Journal Online Sources style) K. Author. (year, month). Title. *Journal* [Type of medium]. Volume(issue), paging if given. Available: http://www.(URL)

[19] R. J. Vidmar. (1992, August). On the use of atmospheric plasmas as electromagnetic reflectors. *IEEE Trans. Plasma Sci.* [Online]. *21(3).* pp. 876–880. Available: http://www.halcyon.com/pub/journals/21ps03-vidmar

Itimad Raheem Ali is currently a PhD candidate in Computer Science at the Department of Computer Graphics and Multimedia, Universiti Teknologi Malaysia. She received the MSc in Computer Science (2006) interest with image processing, and BSc degree in College of Computer Science (1999) from Al-Nahrain Universiti. Her research interest includes real-time computer graphics, facial animation, character motion, natural interaction, lip synchronization, Eye movement, facial expression, Emotions, virtual reality and simulation.