A Novel Digital Watermarking Technique Based on ISB (Intermediate Significant Bit)

Akram M. Zeki and Azizah A. Manaf

***Abstract*—**Least Significant Bit (LSB) technique is the earliest developed technique in watermarking and it is also the most simple, direct and common technique. It essentially involves embedding the watermark by replacing the least significant bit of the image data with a bit of the watermark data. The disadvantage of LSB is that it is not robust against attacks. In this study intermediate significant bit (ISB) has been used in order to improve the robustness of the watermarking system. The aim of this model is to replace the watermarked image pixels by new pixels that can protect the watermark data against attacks and at the same time keeping the new pixels very close to the original pixels in order to protect the quality of watermarked image. The technique is based on testing the value of the watermark pixel according to the range of each bit-plane.

***Keywords*— *Watermarking, LSB, ISB, Robustness*.**

I. Introduction

In a digital image watermarking system, information carrying watermark is embedded in images. Ideally, it should be no perceptible difference between the watermarked and original image, and the watermark should be difficult to remove or alter without the degradation of the host image [1] [2]. A watermark usually is a binary sequence representing a serial number or credit card number, a logo, a picture or a signature. It is used to prove the copyright or ownership. Digital watermarking has become a significant topic of computer science due to the increasing popularity of the Internet and the essential need of data security [3].

In general, watermarking scheme consists of: watermark embedding and watermark extracting. Embedding watermarking into the image is performed usually by modifying the image characteristics such as luminance values or transform domain coefficients. Selection of the coefficients depends on perceptual criteria as well as on a key instrumented permutation to increase the security and robustness of the system. Embedding can be done in an image dependent / independent additive manner or by some substitution mechanisms. It is often necessary to utilize Human Visual System (HVS) models for adaptively embedding the watermark. This can reduce the impacts of the modifications on image quality or for the same visual quality a much stronger watermark can be embedded [4].

II. Bit-Plane Method

A bit-plane of digital images is a set of bits having the same position in the respective binary numbers. In grey scale image representation, there are 8 bit-planes: the first bit-plane contains the set of the most significant bits MSB and the 8th bit-plane contains the least significant bits LSB. The set in between i.e. from 2nd to 7th bit-planes are intermediate significant bits ISB, as shown in Fig. 1.

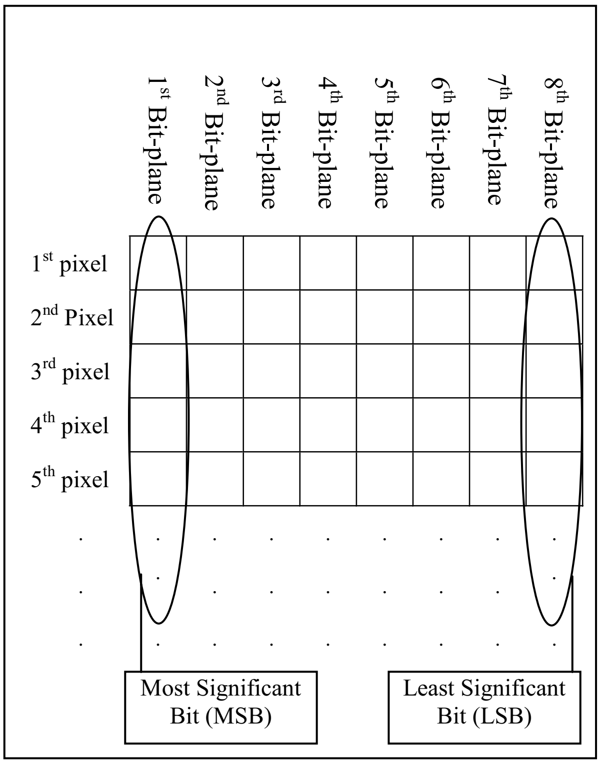


Fig. 1 A bit-plane of digital images

The value of each bit of the 8 bit-plane can be presented by 2^{n-1}, where n is order of the plane starting from 1 to 8. i.e: $(2^0 + 2^1 + 2^2 + 2^3 + 2^4 + 2^5 + 2^6 + 2^7) = (1 + 2 + 4 + 8 + 16 + 32 + 64 + 128) = 255$. The maximum value that can fit in 8 bits is 255 and the minimum value is 0. Any modification to the 8th bit-plane will change the pixel value by ±1, the 7th bit-plane by ±2, the 6th bit-plane by ±4, the 5th bit-plane by ±8, the 4th bit-plane by ±16, the 3rd bit-plane by ±32, the 2nd bit-plane by ±64, and the 1st bit-plane by ±128. As a result, if the changed value is small (such as in 8th bit-plane), the image quality is kept high. While a big changed value (such as 1st bit-plane) causes the image quality to be highly degraded.

III. The Research Approach

In this study, watermarking technique based on intermediate significant bit (ISB) has been developed in order to improve the robustness and the quality of watermarked image. This study uses one bit-plane to

Akram M. Zeki is with the Faculty of Computer Science and Information System, University Technology Malaysia, UTM Skudai, 81310, Johor, Malaysia (akramzeki@yahoo.com).

Azizah A. Manaf is with the College of Science and Technology, University Technology Malaysia, City Campus, Jalan Semarak, 54100 Kuala Lumpur, Malaysia (azizah07@citycampus.utm.my).

embed the watermark object into a selected bit-plane. The next step after selecting one bit-plane for embedding is finding ranges of the chosen bit-plane. The length of the range L is 2^{n-1} (L is the maximum value of each range – the minimum value of the range + 1) and the number of ranges in each bit-plane are 256 / L. We can notice that in each range the bit changes between 0 and 1. The number of ranges for the first bit-plane are 2 only, as follows [0:127] and [128:255]. In other words, the bit in the first range is 0, while the bit in the second range is 1 and the length of each range of the first bit-plane is 128. For the second bit-plane there are 4 ranges as follows: [0:63] [64:127] [128:191] [192:255] and the length of the ranges is 64, and so on, as shown in Table I.

TABLE I
RANGES OF EACH BIT-PLANE WITH THE LENGTH

Bit -Plane	Length of the ranges	Number of ranges	Ranges
1	128	2	[0:127] [128:255]
2	64	4	[0:63] [64:127] [128:191] [192:255]
3	32	8	[0:31] [32:63] … [192:223] [224:255]
4	16	16	[0:15] [16:31] … [224:239] [240:255]
5	8	32	[0:7] [8:15] … [240:247] [248:255]
6	4	64	[0:3] [4:7] … [248:251] [252:255]
7	2	128	[0:1] [2:3] … [252:253] [254:255]
8	1	256	[0] [1] … [254] [255]

During the embedding there is no changing to the pixel if the same bit will be embedded. In other words, the same range will be selected to locate the watermarked pixel (i.e. embedded bit is 1 and the selected bit of the original pixel value was 1 too, or embedding 0 if the original pixel value was 0). But if the selected bit of the original binary pixel is not the same as the embedded one, the previous or next range will be selected. The new range will be determined depending on its distance to the original pixel value. However, selecting nearest range to the original pixel will improve the quality of watermarked image.

The best watermarking robustness can be obtained if the pixel is in the middle of ranges, so that any change on the pixel by attacks will not affect the selected bit. While if the pixel value is located in the edges of ranges, any small change by attacks will move the pixel from a range to the next or previous range. In this case, the selected bit will be changed from 0 to 1 or vice versa. Due to this change, the watermark cannot be extracted. Meanwhile the best watermarked image quality can be obtained if the watermarked pixel (after embedding) is very close to the original pixel. So in case the embedded bit is not the same as the original bit, the best watermarked image quality can be obtained by moving the pixel to the location in the edge of ranges towards the original pixel P.

In this study we will try to find best pixel value in between the middle and the edge of the range that can survive against different types of attacks and at the same time keeping minimum image distortion. This is done by positioning the watermarked pixel away from the edge of the range.

Assume that the bias value (X) is at least the distance from the position of the watermarked pixel P` to the edge of the range (which is more close to the original pixel). That means if the distance from the pixel to the edge of the range is greater than the bias value, then the position of the pixel will not change. While if the distance from the pixel to the edge of the range is smaller than the bias value, then the position of the pixel will change to be as far as the bias value. So, the bias value (X) $\in$ [0, L/2 – 1].

In other words, the minimum of the bias value is 0 and the maximum of the bias value in each bit-plane is L/2 – 1. If the bias value (X) = (L/2 – 1), the watermarked pixel P` will be located in the middle of the range. While if the bias value (X) = 0, the watermarked pixel P` will be the same as the original pixel, this is in case original bit is the same as the embedded one. And P` will be located in the edge of the range if the original bit is not the same as the embedded one. Table II shows different bit-planes with different bias values (X).

TABLE II
THE RANGE OF THE BIAS VALUE (X) FOR DIFFERENT BIT-PLANES.

Bit -Plane	L	X
1	128	[0 - 63]
2	64	[0 - 31]
3	32	[0 - 15]
4	16	[0 - 7]
5	8	[0 - 3]
6	4	[0 - 1]
7	2	[0]
8	1	[0]

From the above table, we can notice that the robustness will not be improved in 7^{th} and 8^{th} bit-plane by using this method, because there are no range middle values. So the watermarked pixel will be located in the edges of the range. Finally we should notice that the extracting stage of the proposed method is very easy, direct extracting from the chosen bit-plane will give the watermark object.

IV. Embedding Process

In this study watermark embedding in all bit-planes will be done with all possible bias values (X) (for every embedding the robustness and the quality of watermarked image will be measured).

To improve the security of the system, the watermark object is encrypted using the shared key using the Data Encryption Standard (DES) algorithm. For this purpose, the watermark data is scrambled in order to ensure additional security. One of the best methods has been used here to encrypt the embedding position and determine the pixel bit, for embedding in the host image, is called the Random Pixel Manipulation Technique [5].

To verify the robustness of the proposed method, normalized cross correlation NCC will be used for this purpose [6] [7]. NCC is an important performance parameter in any extracting module. NCC is defined in (1). Where W(x,y) is the original watermark image and W'(x,y) is the extracted watermark image.

$$NCC = \frac{\sum_x \sum_y W(x,y)W'(x,y)}{\sum_x \sum_y [W(x,y)]^2} \quad (1)$$

To study the quality of watermarked image, the peak signal to noise ratio PSNR will be used for this purpose. In general, a processed image is acceptable to the human eyes if its PSNR is greater than 30 dB [8]. The larger the PSNR, the better is the image quality. The PSNR is defined as given by (2).

$$PSNR = 10\log_{10}\left(\frac{255^2}{\frac{1}{m\times n}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}(\alpha ij - \beta ij)^2}\right)dB \qquad (2)$$

αij is the pixel of the cover image in which the coordinates are (i, j) and βij is the pixel of the watermarked image in which coordinates are (i, j). (m, n) is the size of the cover image and watermarked image.

It shall be assumed that the optimum value should in any case have PSNR not less than 30. Any embedded set which will have PSNR is less than 30 will be ignored. The best NCC value will be chosen as a best embedding status. The proposed method can be presented by few steps as follows:

Step 1: Select the bit-plane one by one, from 0 to 8.

Step 2: Find the length of the range of the selected bit-plane by $L = 2^{n-1}$ (n for 1st bit-plane is 8 while for 8th bit-plane is 1).

Step 3: Create table ranges of the selected bit-plane (Number of ranges is 256 / L).

Step 4: For each range, two pixels are to be found: the maximum value of the range and the minimum value of the range, so that L = maximum value – minimum value +1.

Step 5: Each range is divided into two equal groups; the length of each group is (L/2).

- In case the original bit is equal to the embedded bit, the distance d, between the original pixel and the nearest edge of the range, is to be found and the following steps are done:
  - If the original pixel is in the left-hand group, d = Original Pixel P – minimum pixel value of the range.
    - If (X ≤ d), the watermarked pixel P` = P.
    - Otherwise, the watermarked pixel P` = minimum pixel value of the range + X.
  - If the original pixel is in the right-hand group, d = maximum pixel value of the range – Original pixel P.
    - If (X ≤ d), the watermarked pixel P` = P.
    - Otherwise, the watermarked pixel P` = maximum pixel value of the range – X.
- In case the original bit is different from the embedded bit, the following steps are to be done:
  - If the original pixel is in the left-hand group, or in last range, the watermarked pixel P` = maximum pixel value of the previous range – X.
  - If the original pixel is in the right-hand group, or in the first range, the watermarked pixel P`= minimum pixel value of the next range + X.

Step 6: Calculate the PSNR and NCC for each embedding.

Step 7: If PSNR ≥ 30 db find the highest NCC and save the status which is considered the best status for embedding.

V. Results and Analysis

The watermark object (logo image) which is in grey scale level image is shown in Fig. 2. It contains 90 × 90 pixels and will be embedded within the host grey scale level image which is shown in Fig. 3, and containing 256 × 256 pixels. The embedding done with all bit-planes and few different types of attacks have been applied to the watermarked image in order to test the robustness of the proposed technique.

Fig. 2 Grey scale logo with 90 × 90 pixels

Fig. 3 Grey scale host image with 256 × 256 pixels

The first attack used here is JPEG, which is type of compression that minimizes the amount of colour information in the image that is actually not distinguishable by the human eye. The quality of the compression used is 85.

The second attack is blurring the image, this type of filter that softens an image and makes it looks blurred. This filter returns a circular averaging filter (pillbox) within the square matrix of side 2 × radius+1 [9]. The radius used here is 1.

The third attack is Gaussian filter [10], which is designed to pass a step function with zero overshoot and minimum rise time; this filter returns a rotationally symmetric Gaussian low pass filter of size [3 3]. It has been used here with standard deviation sigma (0.5).

The forth attack is Wiener filter, the goal of this filter is to filter out noise that has corrupted a signal. Wiener method is based on statistics estimated from a local neighbourhood of each pixel of size m-by-n. To estimate the local image mean, standard deviation in this study (m and n) is 3. Wiener filter estimates the additive noise power before doing the filtering [11]; the noise in this study is 0.001.

The fifth attack used here is Speckle noise; this method adds multiplicative noise to the image with mean 0 and variance -in this study- 0.01 [11].

Finally geometric transform attacks such as Rotation and Scaling have been tested here, rotation with angle 1° has been applied and scaling with 50% from the original image is applied. Obviously the robustness to rotation and scaling can only be achieved after re-rotating or re-scaling the image to its original size.

The peak signal to noise ratio (PSNR) and the normalized cross correlation (NCC) for the proposed method for all possible bias values have been calculated after applying chosen attacks as shown in Table III, IV, V, VI, VII, VIII, IX and X for all bit-planes 1^{st}, 2^{nd}, 3^{rd}, 4^{th}, 5^{th}, 6^{th}, 7^{th}, and 8^{th} respectively.

Notice that the PSNR has been calculated after the embedding the data and before applying any attacks. Although some attacks improve the quality of the image such as: filtering and compression, some others destroy the image such as: blurring and noise.

TABLE III
THE PSNR AND NCC FOR THE PROPOSED METHOD FOR THE 1ST BIT-PLANE

X	PSNR	NCC (JPEG)	NCC (Blurring)	NCC (Gaussian)	NCC (Winner)	NCC (Speckle)
0	16.13647	0.795478	0.744931	0.7323	0.7056	0.779114
1	16.00184	0.842855	0.749759	0.7484	0.8473	0.805124
2	15.86849	0.880796	0.758538	0.7586	0.9298	0.844149
3	15.73641	0.90722	0.77309	0.7755	0.9675	0.872815
4	15.6056	0.928738	0.786227	0.7887	0.9843	0.906456
5	15.47602	0.944807	0.799774	0.803	0.989	0.938923
6	15.34768	0.961884	0.815191	0.8168	0.9921	0.970004
7	15.22055	0.972511	0.82719	0.8324	0.993	1
8	15.09462	0.978956	0.84503	0.8522	0.9942	1
9	14.96986	0.985035	0.865521	0.873	0.9959	1
10	14.84627	0.988114	0.879178	0.8858	1	1
11	14.72382	0.99132	0.889336	0.8977	1	1
12	14.6025	0.994532	0.902111	0.91	1	1
13	14.4823	0.996114	0.916023	0.9248	1	1
14	14.36318	0.997196	0.930629	0.935	1	1
15	14.24515	0.997998	0.937078	0.939	1	1
16	14.12817	0.998393	0.940395	0.9438	1	1
17	14.01224	0.99891	0.944081	0.9504	1	1
18	13.89733	0.998948	0.946434	0.9551	1	1
19	13.78344	0.999326	0.953067	0.9614	1	1
20	13.67054	0.999553	0.959391	0.9656	1	1
21	13.55862	0.999828	0.96494	0.9707	1	1
22	13.44765	0.999859	0.969446	0.9739	1	1
23	13.33764	0.999903	0.972865	0.9759	1	1
24	13.22855	0.999896	0.974027	0.9784	1	1
25	13.12037	0.999888	0.975839	0.9806	1	1
26	13.0131	0.999992	0.978075	0.9834	1	1
27	12.9067	0.999992	0.979552	0.9857	1	1
28	12.80117	1	0.981525	0.9879	1	1
29	12.69649	1	0.983731	0.9897	1	1
30	12.59264	1	0.985815	0.9913	1	1
31	12.48962	1	0.988688	0.9934	1	1
32	12.38739	1	0.990442	0.9945	1	1
33	12.28595	1	0.992826	0.9959	1	1
34	12.18528	1	0.994241	0.9963	1	1
35	12.08536	1	0.994502	0.9967	1	1
36	11.98619	1	0.994972	0.9973	1	1
37	11.88775	1	0.995754	0.9975	1	1
38	11.79002	1	0.995827	0.998	1	1
39	11.693	1	0.995935	0.9979	1	1
40	11.59667	1	0.99628	0.9982	1	1
41	11.50102	1	0.996645	0.9984	1	1
42	11.40605	1	0.997328	0.9985	1	1
43	11.31174	1	0.997388	0.9986	1	1
44	11.21808	1	0.997833	0.9987	1	1
45	11.12507	1	0.997954	0.9989	1	1
46	11.03269	1	0.997904	0.9993	1	1
47	10.94094	1	0.997902	0.9995	1	1
48	10.84981	1	0.998106	0.9996	1	1
49	10.75928	1	0.998338	0.9997	1	1
50	10.66936	1	0.998509	0.9999	1	1
51	10.58002	1	0.998673	1	1	1
52	10.49128	1	0.998808	1	1	1
53	10.40311	1	0.999242	1	1	1
54	10.31551	1	0.999349	1	1	1
55	10.22848	1	0.99942	1	1	1
56	10.14201	1	0.999882	1	1	1
57	10.0561	1	0.999947	1	1	1
58	9.97073	1	0.999955	1	1	1
59	9.885903	1	0.99995	1	1	1
60	9.80161	1	0.999948	1	1	1
61	9.717845	1	0.999954	1	1	1
62	9.634601	1	0.999954	1	1	1
63	9.551874	1	0.999954	1	1	1

TABLE IV
THE PSNR AND NCC FOR THE PROPOSED METHOD FOR THE 2ND BIT-PLANE

X	PSNR	NCC (JPEG)	NCC (Blurring)	NCC (Gaussian)	NCC (Winner)	NCC (Speckle)
0	24.06413	0.711555	0.645592	0.635	0.5851	0.697784
1	23.75336	0.785238	0.702612	0.707	0.6442	0.75555
2	23.44878	0.842088	0.755879	0.7629	0.7319	0.810577
3	23.15018	0.876803	0.799364	0.8048	0.7955	0.864797
4	22.85737	0.909485	0.832591	0.835	0.8536	0.896109
5	22.57019	0.931049	0.853794	0.8608	0.8958	0.92045
6	22.28847	0.946609	0.874192	0.8846	0.9197	0.941476
7	22.01203	0.959485	0.889573	0.8969	0.9359	0.955118
8	21.74073	0.969397	0.899595	0.9102	0.9515	0.969498
9	21.47442	0.977538	0.91829	0.918	0.9654	0.979595
10	21.21296	0.983219	0.922482	0.9241	0.9943	0.992713
11	20.95622	0.98888	0.925283	0.9319	0.9952	1
12	20.70405	0.992618	0.932704	0.9406	1	1
13	20.45631	0.993836	0.938843	0.948	1	1
14	20.21286	0.996574	0.945634	0.953	1	1
15	19.97358	0.99799	0.953053	0.9564	1	1
16	19.73833	0.998516	0.958725	0.96	1	1
17	19.50699	0.998988	0.960425	0.9617	1	1
18	19.27942	0.999589	0.962984	0.9669	1	1
19	19.05551	0.999911	0.963277	0.9698	1	1
20	18.83511	0.999923	0.964398	0.9745	1	1
21	18.61813	0.999984	0.965228	0.9754	1	1
22	18.40444	1	0.967169	0.9765	1	1
23	18.19394	1	0.968555	0.9773	1	1
24	17.98652	1	0.973079	0.9783	1	1
25	17.78207	1	0.982227	0.9793	1	1
26	17.58049	1	0.983041	0.9798	1	1
27	17.38168	1	0.983431	0.9823	1	1
28	17.18553	1	0.983483	0.984	1	1
29	16.99196	1	0.983458	0.9845	1	1
30	16.80087	1	0.983453	0.9872	1	1
31	16.61218	1	0.983453	0.9885	1	1

TABLE V
THE PSNR AND NCC FOR THE PROPOSED METHOD FOR THE 3RD BIT-PLANE

X	PSNR	NCC (JPEG)	NCC (Blurring)	NCC (Gaussian)	NCC (Winner)	NCC (Speckle)
0	31.0378	0.748387	0.754281	0.7545	0.7202	0.760441
1	30.3148	0.800633	0.795909	0.8016	0.7515	0.815772
2	29.62676	0.841893	0.824179	0.8319	0.7935	0.850114
3	28.97074	0.877778	0.844959	0.8578	0.8291	0.887326
4	28.34418	0.902601	0.880603	0.8828	0.865	0.917557
5	27.74483	0.922822	0.888579	0.8966	0.8902	0.937661
6	27.17071	0.9387	0.901801	0.911	0.903	0.956092
7	26.62007	0.954276	0.910795	0.9215	0.9162	0.968579
8	26.09127	0.964553	0.91888	0.9296	0.9269	0.984737
9	25.58283	0.972665	0.92555	0.9411	0.9471	0.996588
10	25.09338	0.980038	0.931065	0.9532	0.9903	0.9991
11	24.62168	0.986049	0.934879	0.9601	0.9933	0.99957
12	24.16658	0.989564	0.960694	0.9634	1	0.999761
13	23.72701	0.991286	0.963537	0.9674	1	1
14	23.30198	0.993667	0.963251	0.9717	1	1
15	22.89059	0.995542	0.963181	0.9759	1	1

TABLE VI
THE PSNR AND NCC FOR THE PROPOSED METHOD FOR THE 4TH BIT-PLANE

X	PSNR	NCC (JPEG)	NCC (Blurring)	NCC (Gaussian)	NCC (Winner)	NCC (Speckle)
0	0	37.23247	0.697749	0.726566	0.7358	0.7012
1	1	35.77832	0.752892	0.773111	0.7791	0.7374
2	2	34.46706	0.799402	0.821054	0.8221	0.7715
3	3	33.27344	0.844238	0.844937	0.8493	0.7952
4	4	32.17838	0.875453	0.865212	0.874	0.8295
5	5	31.16731	0.902997	0.876462	0.9005	0.8579
6	6	30.229	0.9203	0.9108	0.9115	0.8885
7	7	29.35279	0.922757	0.916648	0.9259	0.9062

TABLE VII
THE PSNR AND NCC FOR THE PROPOSED METHOD FOR THE 5TH BIT-PLANE.

X	PSNR	NCC (JPEG)	NCC (Blurring)	NCC (Gaussian)	NCC (Winner)	NCC (Speckle)
0	42.41361	0.597743	0.654393	0.6589	0.6177	0.620673
1	39.78235	0.666284	0.721876	0.7269	0.6631	0.659024
2	37.5967	0.726554	0.774336	0.7856	0.6895	0.686901
3	35.72715	0.766078	0.837519	0.8244	0.7232	0.708491

TABLE VIII
THE PSNR AND NCC FOR THE PROPOSED METHOD FOR THE 6TH BIT-PLANE.

X	PSNR	NCC (JPEG)	NCC (Blurring)	NCC (Gaussian)	NCC (Winner)	NCC (Speckle)
0	0	47.17567	0.574057	0.630067	0.6198	0.583
1	1	42.71523	0.649816	0.703711	0.7033	0.6252

TABLE IX
THE PSNR AND NCC FOR THE PROPOSED METHOD FOR THE 7TH BIT-PLANE.

X	PSNR	NCC (JPEG)	NCC (Blurring)	NCC (Gaussian)	NCC (Winner)	NCC (Speckle)
0	51.23908	0.551679	0.604441	0.5862	0.5495	0.569641

TABLE X
THE PSNR AND NCC FOR THE PROPOSED METHOD FOR THE 8TH BIT-PLANE.

X	PSNR	NCC (JPEG)	NCC (Blurring)	NCC (Gaussian)	NCC (Winner)	NCC (Speckle)
0	51.20492	0.561493	0.595022	0.5637	0.5346	0.557159

In addition to above tables the rotation has been applied after every embedding and normalized cross correlation has been tested for all choices, and the result for all embedding was 0.7995. The same testing has been done after applying scaling; the NCC for all embedding was about 0.8092. In other words for these two attacks the NCC values has not been improved when the watermarked pixel moved from the minimum of the bias value (X) to the maximum of the bias value (X) (at the middle of the range).

For the other attacks the results show that the best robustness (biggest NCC) can be obtained with the maximum the bias value (X) but the worst one when the bias value (X) is minimum. From the results we can notice also that the 2nd bit-plane is better than 1st bit-plane when the bias value (X) is minimum. This is because at the 1st bit-plane there were two ranges only and both of them at the edge of the range (when the pixel is located at the edge of the range and the embedded bit differs from the original one, the watermark pixel has only one option located in).

From the results, we can notice that the best watermarked image quality (highest PSNR) was when the bias value (X) is minimum (X = 0, nearest pixel to the original), while the worst one when the bias value (X) is maximum. To select the best embedding status, we consider undetected watermarked image when the PSNR is greater than 30db [12]. Any PSNR value can be chosen by the user, because undetected watermarked image depends on the type of the host image either smooth or texture, and usually the distortion on the smooth images can be noticed by human eye more than texture image areas [13], [14], so the PSNR which are more than 30db are taken.

By simple comparison of every embedding experiment for which PSNR is greater than 30db, the greater NCC will be chosen as a best embedding status. From the above tables we can notice that the best normalized cross correlation was in 4th bit-plane when the bias value (X) was 6. The watermarked image at this level is shown in Fig. 4.

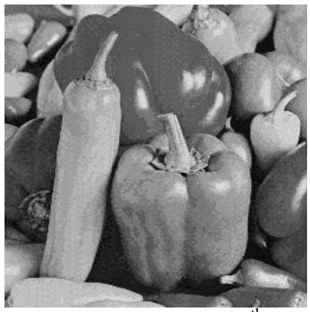


Fig. 4 Watermarked image for the 4th bit-plane for the proposed method with the bias value (X) = 6.

To compare the proposed method with LSB method, the PSNR and NCC have been calculated for LSB method within all bit-plane of the chosen host images after applying chosen attacks as shown in Table XI.

TABLE XI
PSNR AND NCC FOR LSB METHOD WITHIN ALL BIT-PLANES AFTER APPLYING DIFFERENT TYPES OF THE ATTACKS.

Bit-plane	PSNR	NCC JPEG	NCC Blurring	NCC Gaussian	NCC Winner	NCC Speckle
1	8.749611	0.990715	0.944379	0.948909	0.999453	0.993509
2	15.27354	0.951596	0.922206	0.927808	0.989779	0.960868
3	21.01023	0.877564	0.861839	0.869224	0.937693	0.889876
4	27.16578	0.786123	0.800031	0.814711	0.810812	0.804133
5	33.1107	0.65944	0.724851	0.744905	0.667343	0.645563
6	39.1325	0.562295	0.646805	0.670961	0.599686	0.589905
7	45.2193	0.549319	0.577275	0.594592	0.557862	0.572397
8	51.20492	0.546522	0.567865	0.562747	0.53516	0.567465

The rotation and scaling have been applied to above embedding and NCC has been tested for all choices. The results of both attacks are the same as the above results so the NCC for rotation is 0.7995 and for scaling is 0.8092. Hence we conclude that the proposed method in this research can't improve the robustness against geometric transform attacks like rotation and scaling.

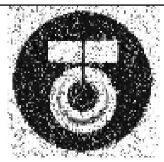
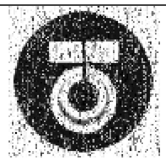
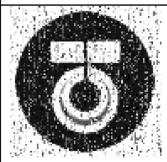
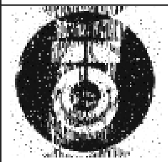
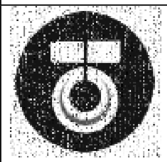
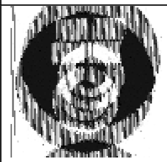
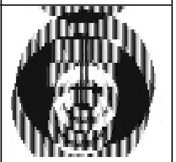
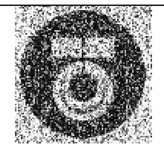
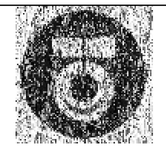
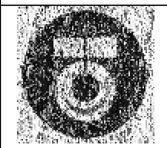
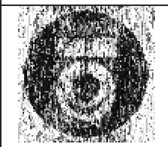
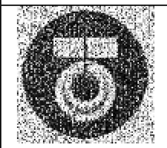
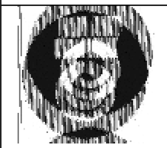
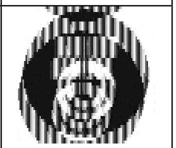
For other attacks, we can see from the results that for all bit-planes the robustness of the LSB method is better than the proposed method at the minimum bias value (X) but it is worst than the proposed method at the maximum bias value (X). It can be seen also for 7th bit-plane and 8th bit-plane (least significant bit) the quality of extracted logo will not be improved nor affected by using the proposed method because the lengths of the ranges 7th and 8th bit-planes are 2 and 1 respectively. So the watermarked pixel is already in the edges of the range and these ranges don't have middle values.

Regarding the quality of watermarked image, we can see that for the 1st bit-plane to the 7th bit-plane the watermarked image qualities by using the proposed method are better than the LSB method whether the bias value (X) is maximum or minimum. While for the 8th bit-plane (least significant bits), the quality has not been improved nor affected by using this method because the length of the range is 1. So the watermarked pixel has no other choice to move in the range and this value is already considered the nearest value to the original pixel value. From the results we can notice also that the acceptable image quality for the LSB method was at the 5th bit-plane while for the proposed method was at the 4th bit-plane when the bias value (X) equals 6.

The extracted logo (watermark 1) from the different host images (when embedding in the 4th bit-plane with bias value = 6) is displayed in Table XII.

TABLE XII

The extracted logo (watermark 1) from the different host images when embedding within 4th bit-plane at X = 6

Host	JPEG	Blurring	Gaussian	Wiener	Speckle	Rotation	Scaling
Proposed Method							
LSB Method							

In order to prove the above result, more samples of host images have been used for embedding as shown in Fig. 5, the selected watermark logo has been embedded within the 4th bit-plane of the host images, the PSNR and NCC have been measured with all possible bias values from 0 to 7 as shown in Table XIII for host 2 and Table XV for host 3.

Host 2 Host 3

Fig. 5 Grey scale host images with 256 × 256 pixels.

TABLE XIII

THE PSNR AND NCC FOR THE 4TH BIT-PLANE OF THE HOST 2.

X	PSNR	NCC (JPEG 70)	NCC (Blurring)	NCC (Gaussian)	NCC (Winner)	NCC (Speckle)
0	37.42594	0.663488	0.691445	0.698097	0.711266	0.743068
1	35.96755	0.718753	0.727455	0.741954	0.758124	0.806056
2	34.65274	0.772147	0.772357	0.784091	0.800193	0.85972
3	33.45586	0.817115	0.794016	0.808013	0.833765	0.905449
4	32.35761	0.856196	0.81089	0.83096	0.869729	0.926617
5	31.34309	0.881363	0.822679	0.85131	0.891881	0.941138
6	30.40038	0.899963	0.853466	0.866833	0.921045	0.951933
7	29.51966	0.911523	0.858512	0.87805	0.935416	0.951818

TABLE XV

THE PSNR AND NCC FOR THE 4TH BIT-PLANE OF THE HOST 3.

X	PSNR	NCC (JPEG 70)	NCC (Blurring)	NCC (Gaussian)	NCC (Winner)	NCC (Speckle)
0	37.1483	0.696117	0.744099	0.747873	0.671389	0.740201
1	35.67559	0.778842	0.810719	0.81824	0.712075	0.801931
2	34.3537	0.834044	0.861787	0.870255	0.740892	0.855158
3	33.15449	0.874584	0.887748	0.900579	0.75553	0.905956
4	32.05719	0.902233	0.905217	0.920152	0.790651	0.944097
5	31.04604	0.925196	0.913922	0.942387	0.81637	0.963153
6	30.10879	0.939509	0.943015	0.949099	0.856248	0.973784
7	29.23579	0.948898	0.946055	0.959664	0.877583	0.982908

The above tables prove the above result, that the best robustness can get with undetectable watermarked images (PSNR ≥ 30db) in the 4th bit-plane at the bias value = 6.

The contribution from the above results is that the robustness for the proposed method is better than the LSB method. Although the difference is not so much but this result gives new threshold value that can replace the classic LSB method.

Finally we must mention that the robustness of the proposed method has not been improved against geometry transform attacks which do not change the value of the pixel but they transform the pixel to another location. To solve this problem repeating the watermark embedded in a special form may be done for this purpose.

VI. Conclusion

This study tested the location of the watermark pixel according to the range of each bit-plane, so if the watermarked pixel is in the middle of the range then any effect on the pixel by attacks will be difficult to move the selected bit to another range. While if the pixel value is located in the edges of ranges, any small change by attacks will move the pixel from a range to another, and the watermark cannot be extracted. This study was trying to find best pixel value in between the middle and the edge of the range that can protect the watermark object from different types of attacks and at the same time keeping minimum distortion of watermarked image. This was done by positioning the watermarked pixel away from the edge of the range. In this study we found that the best extracted logo from undistorted watermarked image was in the 4th bit-plane where the distance from the edge of the range to the position of the watermarked pixel was 6. The contribution of this research is replacing the classic least significant bits (LSB) technique by a new technique called intermediate significant bits ISB, which improves the robustness and maintains the quality of watermarked images. The threshold value for the best embedding status has been found based on ISB.

References

[1] Voyatzis, G and Pitas, I. "Protecting Digital-Image Copyrights: A Framework", IEEE Trans. On Computer Graphics and Application, Volume. 19, No. 1, pp. 18-24, 1999.

[2] Chen, Wen-Yuan. "A Comparative Study of Information Hiding Scheme Using Amplitude, Frequency and Phase Embedding", PhD thesis. National Cheng Kung University, Tainan, Taiwan, 2003.

[3] Chin, C. C. Guei, M. C.hen, Min-Hui Lin. Information hiding based on search-order coding for VQ indices. Pattern Recognition Letters 2004.

[4] Ahmet B. Watermarking Capacity Improvement by Low Density Parity Check Codes. Master thesis. Bogazici University. Turkey. 2002

[5] Venkatraman, S., Abraham, A., and Paprzycki, M. (2004). Significance of Steganography on Data Security. Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC'04). IEEE Computer Society.

[6] Maity, S. P. Kundu, M. K. "Robust and Blind Spatial Watermarking In Digital Image", Proc. 3rd Indian Conf. on Computer Vision, Graphics and Image Processing (ICVGIP '2002), pp. 388 -393, Ahmedabad, India, 16-18th December, 2002.

[7] Parameswaran, L. Anbumani, K. "A Robust Image Watermarking Scheme using Image Moment Normalization", Transactions on Engineering, Computing and Technology. ISSN 1305-5313, Vol. 13 May, 2006.

[8] Chen, T. H. Horng, G and Wang, S. H. "A Robust Wavelet-Based Watermarking Scheme Using Quantization and Human Visual System Model". Pakistan Journal of Information and Technology, 2 (3): 213-230, 2003. ISSN 1682-6027.

[9] MathWorks, Inc. www.mathworks.com, last accessed on February 2007.

[10] Yuan, Y. B. Vorburger, T.V. Song, J. F. Renegar, T. B. A Simplified Realization for the Gaussian Filter in Surface Metrology. International Colloquium on Surfaces, Chemnitz (Germany), Jan. 31 - Feb. 02, 2000.

[11] McAndrew, Alasdair. Introduction to Digital Image Processing with Matlab, Thomson Course Technology, 2004, ISBN. 0534400116.

[12] Bennour J., Dugelay J-L. and Matta F. "Watermarking Attack: BOWS contest". Proceeding of SPIE Feb 2007.

[13] Jain, A. K. and Uludag, U. "Hiding Fingerprint Minutiae in Images", Proc. AutoID 2002, 3. Workshop on Automatic Identification Advanced Technologies, pp. 97-102, Tarrytown, New York, USA, March 14-15, 2002.

[14] Wu, D. C. and Tsai, W. H. A steganographic method for images by pixel-value differencing. Pattern Recognition Letters, 24, 1613–1626. 2003.