Research Journal of Applied Sciences, Engineering and Technology 7(8): 1643-1654, 2014
DOI:10.19026/rjaset.7.444
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: June 11, 2012 Accepted: July 01, 2013 Published: February 27, 2014

Research Article

Digital Image Watermarking With Random Selection of Watermark Insertion Having Adaptive Strength

[1]G.S. Kalra, [2]R. Talwar and [3]H. Sadawarti
[1]School of Electronics Engineering, Lovely Professional University, Jalandhar, Punjab, India
[2]Swift College of Engineering and Technology,
[3]RIMTIET, Affiliated to Punjab Technical University, Punjab, India

Abstract: We have presented an algorithm of digital image watermarking for gray scale images which we implemented in frequency domain. Before inserting the watermark, we added the Hamming codes row wise as well column wise. Two encryption techniques were implemented on the ECC inserted watermark for its security. The pixel position for inserting the watermark was calculated using starting row and column number for that 8×8 block. Pixel embedding strength is calculated using criteria that low frequency is robust in general signal processing attacks, thus choosing less value to be embedded and vice-versa. Results show that the watermarking algorithm is robust against common signal processing attacks. The algorithm is tested against multiple attacks also.

Keywords: DCT, DWT, error correcting codes, frequency domain, watermarking

INTRODUCTION

The tremendous growth of internet resulted penetrating in the remote areas. It is even present where the person find hard to reach. Everyone needs latest images, audio files or video files and they are getting it free of cost on the internet. The original producer of the file even doesn't know that the file created by him/her is available for free through internet and even if knows, nothing can be done. Then the need arises for a method so that the actual producer can prove that the file belongs to him/her. There are many solutions for this problem as far images are concerned like Steganography, cryptography and digital watermarking. In digital watermarking, a specific code or mark is embedded permanently inside a cover multimedia file which remains within that cover invisibly or visibly even after decryption process. The embedding of watermark may be robust or fragile depending on the application. Also embedding can be in spatial domain or frequency domain. Generally, for robustness, embedding is done in the frequency domain so that the watermark could be scattered in a range of frequencies which are not noticeable by the human visual system. The watermark is a sequence of recognizable sequence of bits, a copyright mark or can be a image. Many techniques in spatial domain as well as in frequency domain using DCT, DWT and DFT exist.

LITERATURE REVIEW

Assigning codeword for the purpose of fingerprinting digital data were implemented (Dan and James, 1998) successfully, e.g., software, documents, music and video. Different codes were used for fingerprinting different files. These codes can be compared with certain other copies and some bits can be detected to get it changed. To secure those binary bits, the use of dual binary Hamming codes was introduced (Domingo-Ferrer and Herrera-Joancomarti, 2000). Hamming error correction technique was used for image watermarking using wavelet transform (Hongtao *et al*., 2002) in color images but the embedding results in low quality of image with PSNR below acceptable 30 dB. In an investigation (Terzija *et al*., 2002), watermark was encoded with three different error correction techniques which are Hamming, BCH and RS code. Following the idea of the JPEG standard, the encoded watermark was embedded by a method based on DWT. For some applications, it was proved that Hamming code to be the best ECC because with Hamming code there are fewer bits to be embedded. Similar analysis of ECC in spatial domain watermarking (Limin *et al*., 2003) shows no effect on bit error rate. ECC were also used effectively to store the history of the patient in the medical images (Nayak *et al*., 2004). For embedding the watermark, adaptive pixel coordinates can be considered depending on the

Corresponding Author: G.S. Kalra, School of Electronics Engineering, Lovely Professional University, Jalandhar, Punjab, India

local pixel selection criteria but it will create the de-synchronization between the Hamming codes (Solanki *et al*., 2004; Man and Jian-Guo, 2009). Embedding of ECC considered being extra payload which is not actually a watermark, but only used to correct the watermark after extraction. Another technique used multiple watermarks embedding (Jayalakshmi *et al*., 2008) in the same image file for better security but reduces the perceptual quality of the image. As selection of local criteria for embedding watermark is not suitable for ECC, M-ary phase modulation (Yongqing *et al*., 2008; Kung and Troung, 2006) was introduced. In this model, they proposed a distributed watermark in some range of DCT coefficients. In these algorithms multi bit embedding was done which results in more damage, if certain part of image is corrupted. Pixel comparison technique (Shijun *et al*., 2008) was also presented in which pixel embedding results in the change of pixel in comparing to the neighboring pixel. The major disadvantage of this technique was that it is very immune to histogram equalization attack. Blind watermarking techniques in frequency domain are able to survive the common signal processing attacks for the color images (Zhang *et al*., 2009) also. BCH (Cika, 2009) and RS (Abdul *et al*., 2009; Guyeux and Dahi, 2010) codes employed in frequency domain provide more security to the watermark but for the burst errors and also the extra bits added are more which reduces the embedding capacity of the actual watermark. Block wise DCT along with Dynamic Fuzzy Inference System (DFIS) was introduced (Hossein *et al*., 2010) in which adaptive method of pixel insertion selection was done. The insertion place was decided by applying block DCT and DFIS on cover image as well watermark image and DC coefficients was modified. Multiple attacks on the watermarked image were able to reduce the non correlation below the required level. Adaptive pixel selection according to the local complexity was proposed (Xiaolong *et al*., 2011) which incorporate adaptive embedding but optimal adaptive-embedding-threshold used for image partition was determined iteratively which was its major drawback. A sample projection approach (Akhaee *et al*., 2011) was employed for the embedding of watermark. Four samples from the approximation coefficients were taken to make a line segment in 2-D space. The slope of the line was invariant to the gain factor. Then the watermark code was embedded by projecting the line on the line build by message bits. The algorithm showed good results for AWGN, compression and filtering attacks but it was sensitive to the collusion attacks. The embedding of pixel value can also be adaptive which may depend upon local parameters. It can also be dependent upon the difference between actual pixel value and JND value calculated (Seung *et al*., 2011).

After reviewing the literature, we come to conclude that for developing an algorithm for watermarking certain points are to taken care. Digital image watermarking to be done should be blind and in frequency domain for the robustness of algorithm. For robustness of watermark, dual encryption techniques are to be used which are Arnold transformation and Chaos. For robustness against attacks, error correcting technique is to be employed which will be Hamming error correction technique for which extra data bits are less than the other ECC techniques. The Hamming codes are to be added row wise as well as column wise which makes it 2-D. Thus the iterations can improve the results of the extracted watermark. Watermark embedding is to be done in not single frequency but in multiple middle frequencies. Further, multiple frequencies are not equally immune to the attacks. So the frequencies which are less immune should contain low value of the watermark and which are highly immune should have more value of the watermark. Thus the embedding the pixel value will be adaptive.

THE PROPOSED SCHEME

The proposed scheme is made up after concluding the literature survey. It utilizes the advantages of discrete cosine transformation, wavelet transform, Arnold Transform, Chaos and Hamming error codes. Two encryption techniques are used to enhance the security of the watermark.

The image for watermarking is first applied by the discrete wavelet transform up to two levels as shown in Fig. 1. This is because depending upon the discrete wavelets theory and human visual characteristics; we know that the embeddable watermarking capacity will decrease with the increase of layer numbers.

Then 2 D-DCT will be applied on the middle frequency band which is HL 2. Discrete Cosine Transform (DCT) have the advantage over the other domains like, spatial and DWT. It is more robust against the attacks specifically jpeg lossy compression because of its energy compaction property.

As far as watermark is concerned, Hamming error codes are to be inserted row wise as well as column wise. Arnold and chaos encryption will be applied on the coded watermark. Arnold transformation defined by (1) is a one-to-one transformation:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \left[\begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} mod 1 \right] \quad (1)$$

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \left[\begin{pmatrix} 1 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} mod N \right] (x, y) \in 0, 1\ldots., N\text{-}1 \quad (2)$$

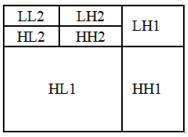

Fig. 1: Three level wavelet decomposition

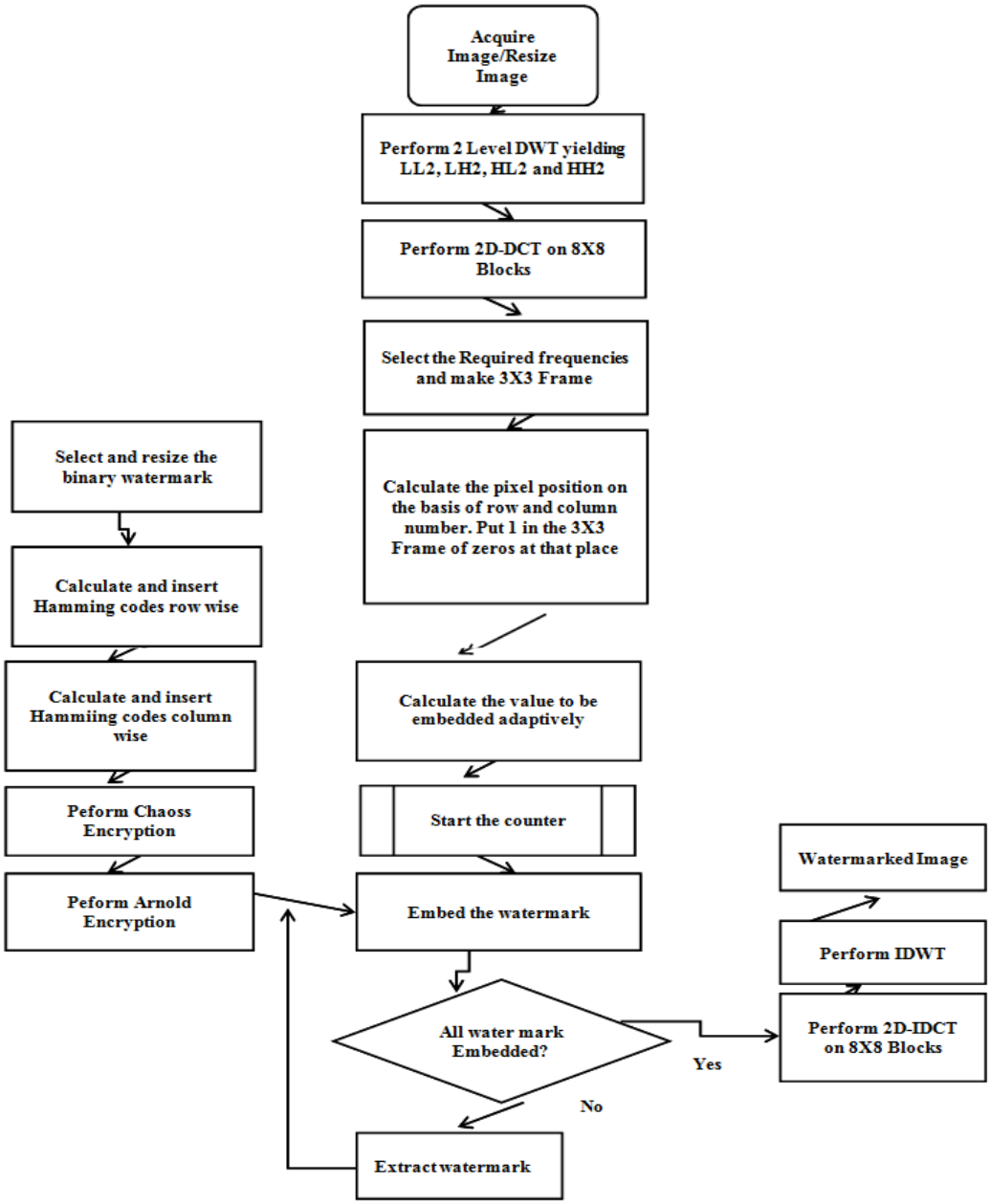

Fig. 2: Flow chart for embedding algorithm

Equation (2) is used to transform each and every pixel coordinates of the images. Where (x, y) is the location coordinates of the original image pixels and (x', y') is the location coordinates of image pixels that after transform When all the coordinates are transformed, the image we obtain is scrambled images. Chaos signals are a kind of pseudorandom, irreversible and dynamical signals, which process good characteristics of pseudorandom sequences. Chaotic systems are highly sensitive to initial parameters. The output sequence has good randomness, correlation, complexity and is similar to white noise and shown in (3):

$$x(n+1) = \mu * x(n) * [1 - x(n)] \quad (3)$$

where, $\mu \in (0, 4)$; $x(n) \in (0, 1)$. By initializing μ and $x(0)$, we can get the required chaotic signal. In order to get chaotic sequences, the chaotic signal $x(n)$ must be transformed into binary sequence $s(n)$. So quantized function $T[x(n)]$ is used and can be given by (4):

$$T[x(n)] = \begin{cases} 0 & x(n) \epsilon \bigcup_{k=0}^{2^{m-1}} I_{2k}^{m} \\ 1 & x(n) \epsilon \bigcup_{k=0}^{2^{m-1}} I_{2k=1}^{m} \end{cases} \quad (4)$$

where, m is random integer and should be greater than 0. ($I_0^m, I_0^m, \ldots\ldots$) is continuous equal interval in [0, 1] and the interval is divided by 2^m.

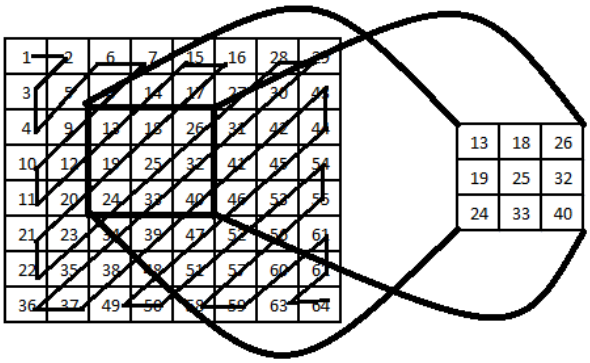

Fig. 3: Selection of random middle frequencies

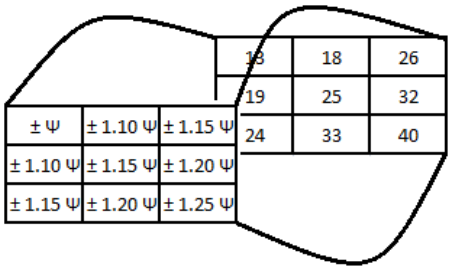

Fig. 4: Embedding strength for different frequency of pixels

	Col. 3	Col. 4	Col. 5
Row3	0	0	1
Row4	0	0	0
Row5	0	0	0

Fig. 5: Calculated pixel (3, 5) in the 3×3 frame

Pixel selection and embedding strength: The proposed scheme utilizes the advantages of DCT, DWT, Arnold Transform, Chaos and Hamming as ECC. Two encryption techniques are used to enhance the security of the watermark. The image for watermarking is first applied by the DWT up to two levels as shown in Fig. 1.

The original image was taken and resized to 1024×1024 pixels. Discrete wavelet transformation was applied on the resized image up to second level which will yield LL2, LH2, HL2 and HH2 bands. Middle frequency band was taken which was now 256×256 sized and 8×8 discrete cosine transformation was applied as shown in the flow chart for the embedding algorithm shown in Fig. 2. We targeted particular middle frequencies, so that the algorithm should be able to sustain low pass and high pass filtering attacks. For a particular 8×8 DCT block, nine middle frequencies were selected for embedding the watermark. These nine frequencies are randomly selected in our case but it could also depend upon the application. The selection of frequencies to form the 3×3 mask is shown in Fig. 3.

As it is clear from Fig. 3 that top left corner have the lowest frequency pixel selected and right bottom have the highest frequency from the selected pixels. We know that low frequency pixels are less immune to the common signal processing attacks and high frequency pixels are more. Thus, we also need the embedded value to be adaptive for these pixels. Thus, we embedded low embedding strength in the low frequency pixels and more embedding strength in the high frequency pixels. The embedding strength frame is shown in Fig. 4.

Here, Ψ represents the embedding strength calculated according to the formula:

$$\Psi = +\sigma\text{^}2/\alpha, \text{ if watermark is 1} \quad (5)$$

$$\text{and } \Psi = -\sigma\text{^}2/\alpha, \text{ if watermark is 0} \quad (6)$$

where,
σ^2 = The variance of the original image
α = The embedding strength division factor

Now the task is to find the exact pixel depending on the row and column number, where the pixel value is to be embedded. First of all, all the numeric in the first row and first column number of that particular block are to be added till the end result becomes a single digit. This digit will remain the same for embedding as well as extraction process as the size of the image will remain same and so the number of 8×8 blocks is. Then new coordinates of the embedding pixel are to be decided as follows:

```
if (xm == 1)||(xm == 6)||(xm == 8)
x' = 3;
else if (xm == 0)||(xm == 2)||(xm == 3)||(xm == 5)
x' = 4;
else
x' = 5;
end
if (ym == 1)||(ym == 6)||(ym == 8)
y' = 3;
else if (ym == 0)||(ym == 2)||(ym == 3)||(ym == 5)
y' = 4;
else
y' = 5;
end
```

where, x_m and y_m are the first row and column numbers of that particular block and (x', y') is the new calculated pixel position. On this pixel position 1 has to be inserted and rest all pixels of 3×3 are to be made 0 as shown in Fig. 5. Here we have shown that (x', y') are coming out to be (3, 5).

Now combining this frame with the other two frames shown in Fig. 4, the net value and the pixel

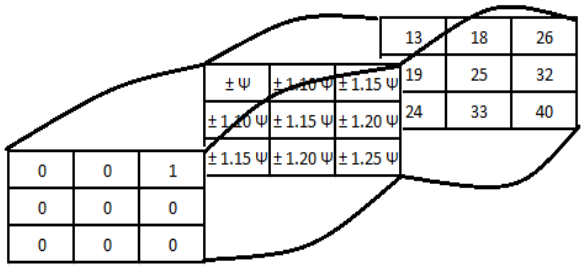

Fig. 6: Net frame for embedding pixel place and embedding strength

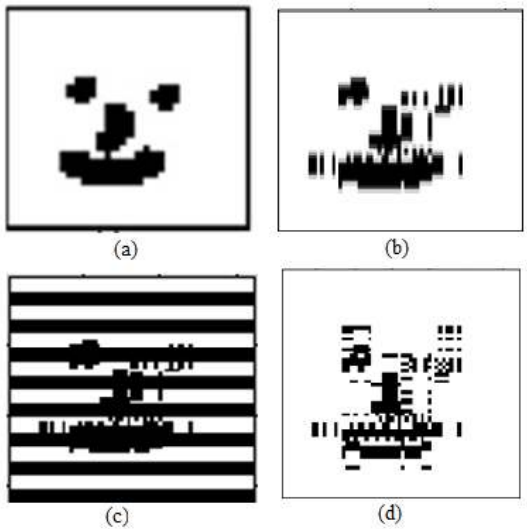

Fig. 7: (a) Original watermark, (b) row wise inserted hamming codes, (c) column wise inserted zeros, (d) column wise inserted hamming codes

position can be decided according to these three frames as shown Fig. 6.

Watermark pre-processing: Original binary watermark is to be resized to 32×32 pixels shown in Fig. 7b. Original watermark is converted to 64×64 watermark, by inserting 4 rows and 4 columns as watermark and rest 4, 4 as zeros. Then hamming codes was calculated and inserted on these places row wise and column wise respectively as shown in Fig. 7d. Then Chaos and Arnold encryptions were performed on the processed watermark and are shown in Fig. 8a and b respectively.

Watermark extraction: Watermark extraction procedure is shown in Fig. 9. Few steps are same as presented in watermark embedding algorithm. Threshold in our case was a crisp set which was considered to be zero. Further when extraction was performed, inverse Arnold and inverse chaos was performed. Now the extracted watermark contained the 2-dimensional Hamming codes. We used Hamming (7, 4) for which we need only seven bits and we used

Fig. 8: (a) Chaos encrypted watermark, (b) chaos and Arnold encrypted watermark

eighth bit as a parity which we used to find that if there are more than one, even number of errors. We performed inverse Hamming codes column wise and then performed inverse Hamming row wise. If error was found with the parity, inverse hamming was skipped for that particular row or column. After extraction some iterations were performed due to the fact that, after column wise extraction, it is possible that some error remained there which can be corrected after row wise inverse hamming. Now, after removal of some errors, again same iterations were performed to reduce the bit error rate.

PERFORMANCE EVALUATION

The performance of the watermarked image can be evaluated on the basis of Peak Signal to Noise Ratio (PSNR) in decibels (dB). Higher the value of PSNR better is the quality of the watermarked image. PSNR more than 30 dBs is considered to be the acceptable quality image in which watermark is making no alteration to the quality of the image:

$$\text{MSE} = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i,j) - K(i,j\})]^2 \quad (7)$$

$$\text{PSNR} = 10\log_{10}\left(\frac{R}{MSE}\right) \quad (8)$$

where, MSE is the mean square error of the watermarked image and the original image and m, n are the number of rows and number of columns. I and K are the watermarked images.

The quality of the extracted watermark is evaluated using term Normalized Cross-correlation (NC). The ideal value of the NC is 1 which means the original and the extracted watermarks are exactly the same which is given by the (9):

$$\text{NC} = \frac{\sum_{i=0}^{m-1}\sum_{j=0}^{n-1} W(i,j) * W'(i,j)}{\sqrt{\sum_{i=0}^{m-1}\sum_{j=0}^{n-1} W(i,j)^2}\sqrt{\sum_{i=0}^{m-1}\sum_{j=0}^{n-1} W'(i,j)^2}} \quad (9)$$

The Bit Error Rate (BER) can be calculated as given in (10):

$$\text{BER} = \frac{\sum_{1}^{m*n} w(i) EXOR\, w'(i)}{m*n} \quad (10)$$

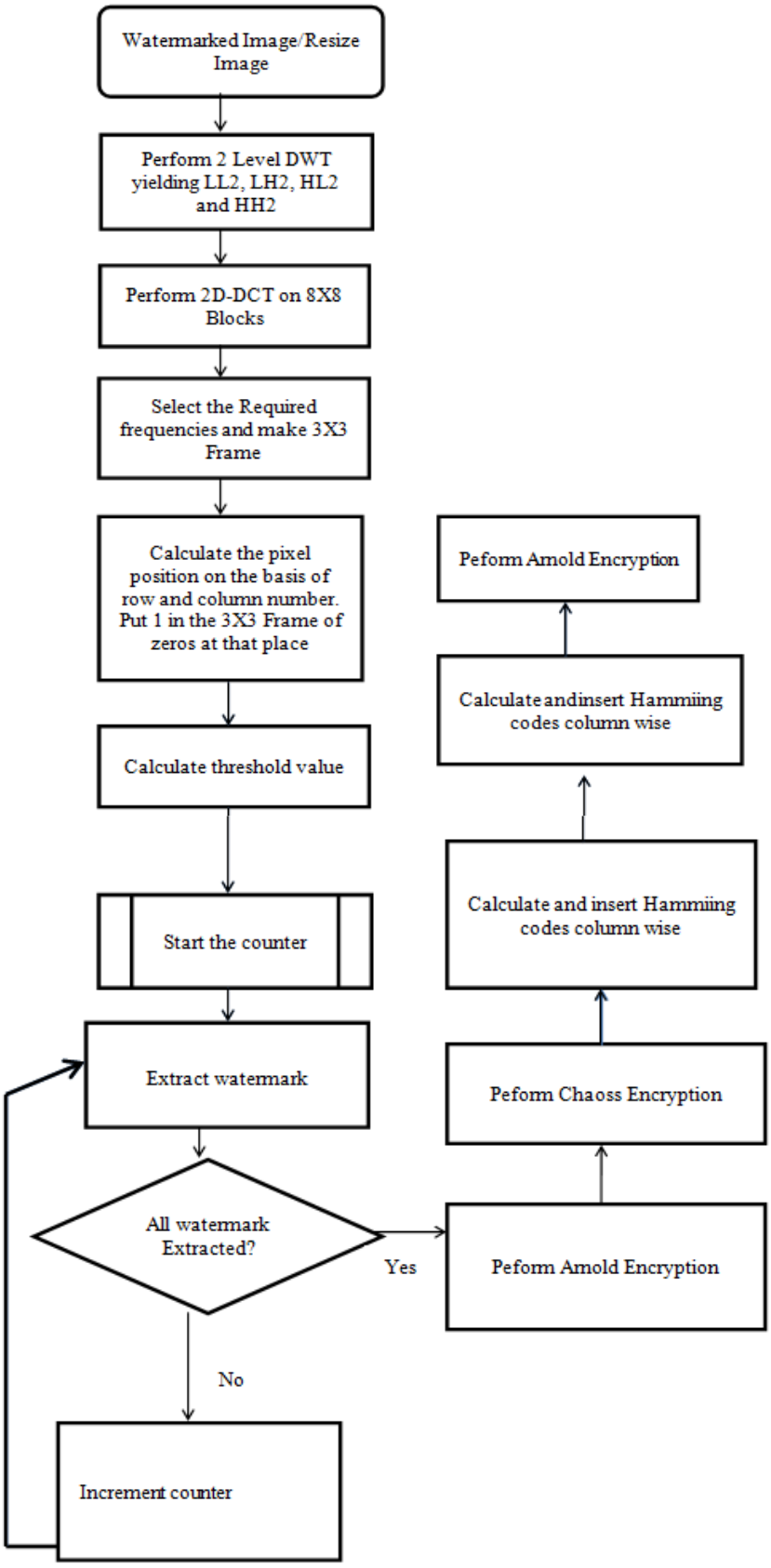

Fig. 9: Extraction algorithm

Fig. 10: Watermarked images, (a) cameraman having PSNR 35.92 dB, (b) Lena having PSNR 37.77 dB, (c) peppers having PSNR 38.37 dB, (d) baboon having PSNR 31.98 dB

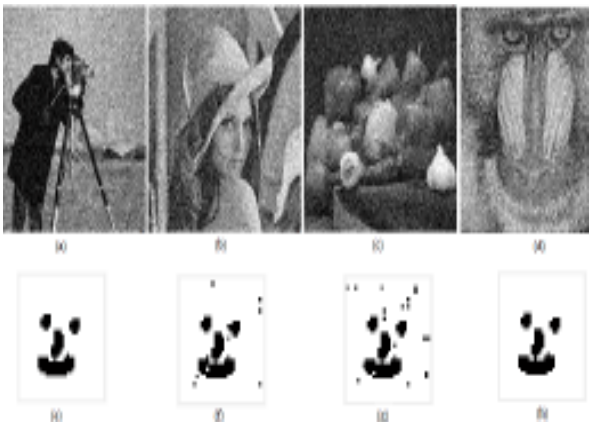

Fig. 11: Gaussian noised images with mean = 0 and variance = 0.02, (a) cameraman image, PSNR = 17.43 dB and extracted watermark shown in (e) with NC = 1, (b) Lena image, PSNR = 17.15 dB and extracted watermark shown in (f) with NC = 0.998, (c) peppers image, PSNR = 17.41 dB and extracted watermark shown in (g) with NC = 0.993, (d) baboon image, PSNR = 16.98 dB and extracted watermark shown in (h) with NC = 1

where, W (i, j) is the original watermark and W' (i, j) is the extracted watermark.

RESULTS AND DISCUSSION

The watermarked Cameraman, Lena, Peppers and Baboon images with size 1024×1024 are shown in Fig. 10. These watermarked images are tested for various attacks and then extracted the watermark from the attacked images. The NC values are calculated for original watermark and extracted watermark at three steps which are without error correcting codes, with error correcting codes and with iterations of error correcting codes.

Gaussian noise: The watermarked images were tested against Gaussian noise with mean = 0 and different variance values. The Gaussian noised images (a-d) and extracted watermarks (e-h) are shown in Fig. 11. The overall performance of the algorithm against Gaussian noise with different values of variance is shown in Fig. 12. With different values of variance from 0 to 0.15, the algorithm shows robust against till variance value of 0.10 but it also depend upon the frequencies present in the image. If the image is having high frequencies like in baboon, the watermark was able to be protected till variance value 0.15.

Salt and pepper noise: The watermarked images were tested against salt and pepper noise with different values of depth. The salt and pepper noised images (a-d) with depth of 10% and extracted watermarks (e-h) are shown in Fig. 13. The overall performance of the algorithm against salt and pepper noise with different values depth is shown in Fig. 14. With different values depth from 1 to 25%, the algorithm shows robust against salt and pepper noise till 15% depth. If the image is having high frequencies like in baboon, the watermark was able to be protected till depth of 25%.

Speckle noise: The watermarked images were tested against speckle noise with different values of variance. It is a granular noise that inherently exists in and degrades the quality of the active radar and synthetic aperture radar images. The speckle noised images (a-d) with variance and extracted watermarks (e-h) are shown in Fig. 15. The overall performance of the algorithm against salt and pepper noise with different values depth is shown in Fig. 16. With different values of variance from 0.1 to 0.20, the algorithm shows robustness against speckle noise till variance of 0.35. If the image is having high frequencies like in baboon, the watermark was able to be protected till depth of 25%.

Jpeg compression: For evaluating our watermarking algorithm, watermarked images were compressed by jpeg compression with quality factors 50, 40, 30 and 20%, respectively and watermark is extracted from the compressed images and corresponding PSNR values and NC values are given in Table 1. The watermarked images with 30% jpeg compression (a-d) and extracted watermarks (e-h) are shown in Fig. 17.

Median filtering: Generally median filtering is able to attack the watermark present in the watermarked images. Thus we applied 3×3 and 5×5 mask with the median filter and corresponding PSNR and NC values are listed in Table 1 and watermarked images with 3×3 median filter (a-d) and the extracted watermarks (e-h) are shown in Fig. 18.

Additive White Gaussian Noise (AWGN): It is a type of noise in which we can set the target values of snr and then model that whether our watermark will be retained or not. We added AWGN with 20, 15 and 10 dB SNR on the watermarked images and corresponding PSNR and NC value are mentioned in the Table 1. The watermarked images (a-d) of cameraman, Lena,

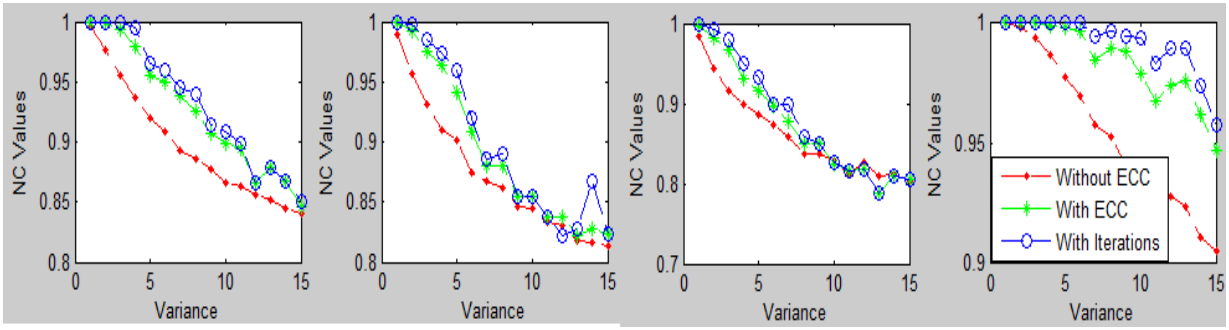

Fig. 12: Extracted NC values for extracted watermark from, (a) cameraman, (b) Lena, (c) peppers, (d) baboon images. The noise was Gaussian noise with 0 mean

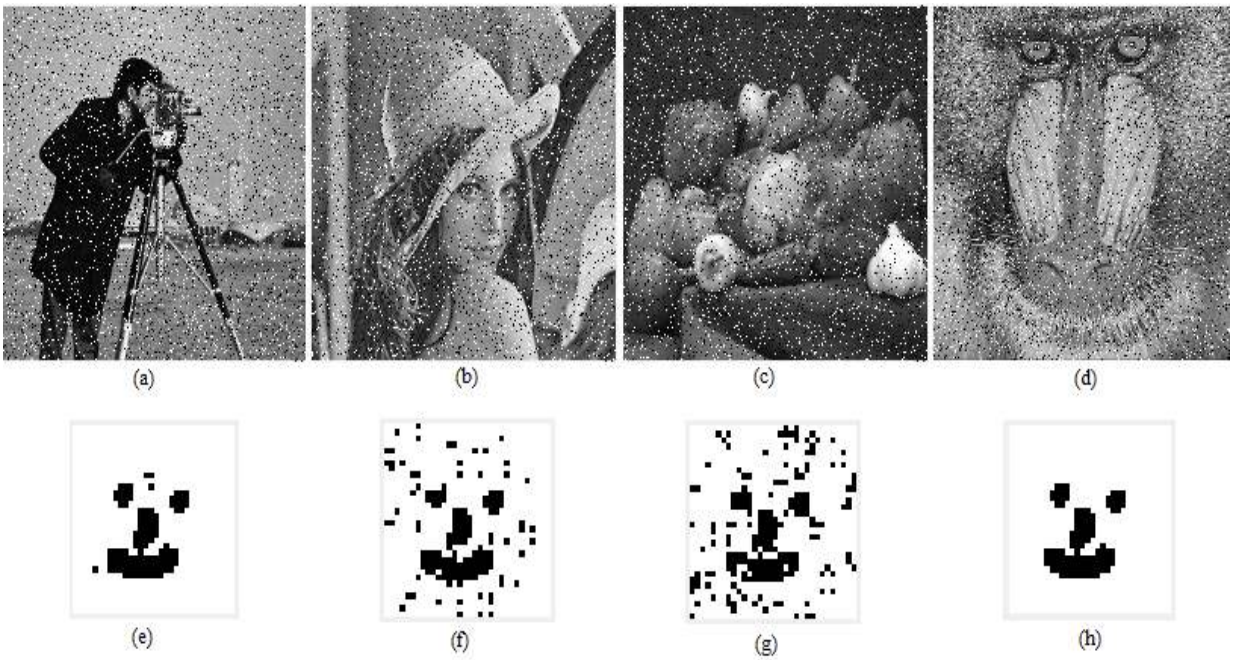

Fig. 13: Salt and pepper noised images with depth of 10%, (a) cameraman image, PSNR = 15.02 dB and extracted watermark shown in (e) with NC = 1, (b) Lena image, PSNR = 18.42 dB and extracted watermark shown in (f) with NC = 0.979, (c) peppers image, PSNR = 14.98 dB and extracted watermark shown in (g) with NC = 0.953, (d) baboon image, PSNR = 15.48 dB and extracted watermark shown in (h) with NC = 1

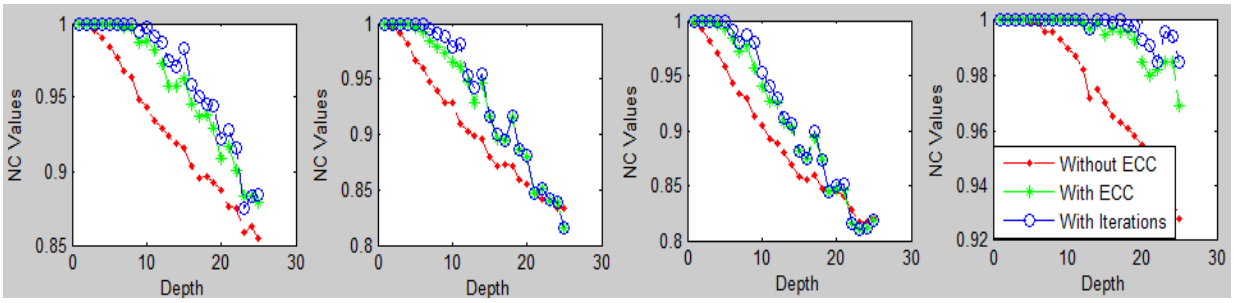

Fig. 14: Extracted NC values for extracted watermark from, (a) cameraman, (b) Lena, (c) peppers, (d) baboon images. The noise was salt and pepper noise with 10% depth

Peppers and Baboon and extracted watermarks (e-h) are shown in Fig. 19.

Scaling: The watermarked images are tested for scaling attack in which watermarked images were resized to 60,

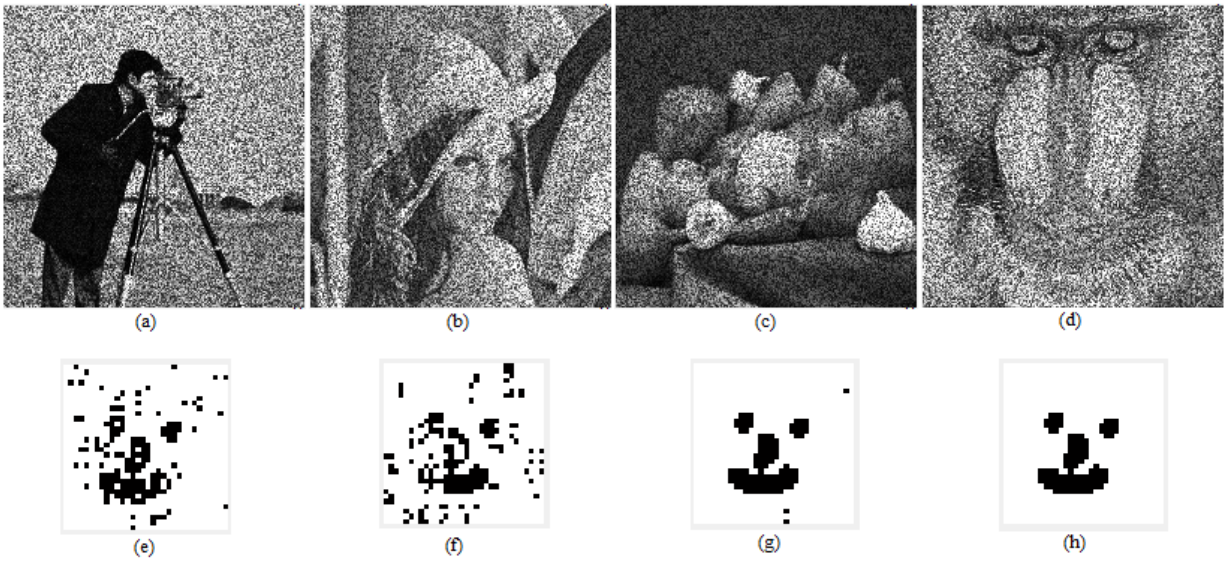

Fig. 15: Speckle noised images with variance of 0.20, (a) cameraman image, PSNR = 13.12 dB and extracted watermark shown in (e) with NC = 0.969, (b) Lena image, PSNR = 13.24 dB and extracted watermark shown in (f) with NC = 0.951, (c) peppers image, PSNR = 16.19 dB and extracted watermark shown in (g) with NC = 0.994, (d) baboon image, PSNR = 12.92 dB and extracted watermark shown in (h) with NC = 1

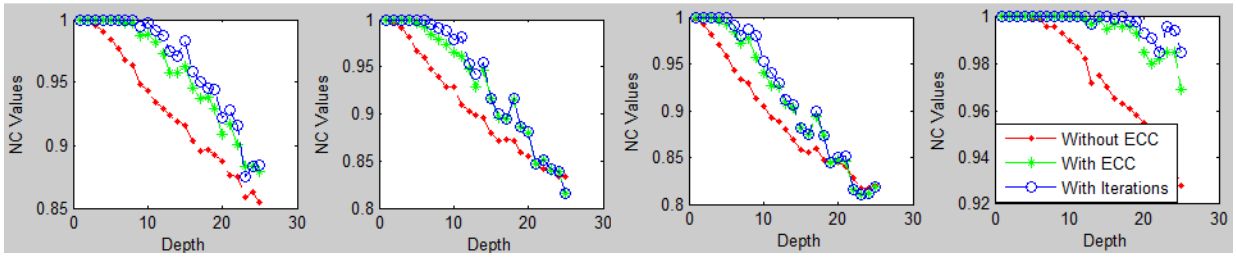

Fig. 16: Extracted NC values for extracted watermark from, (a) cameraman, (b) Lena, (c) peppers, (d) baboon images. The noise was speckle noise with variance 0.20

Fig. 17: Watermarked, (a) cameraman, (b) Lena, (c) peppers, (d) baboon images with 30% jpeg compression and corresponding extracted watermarks (e-h)

Fig. 18: Watermarked, (a) cameraman, (b) Lena, (c) peppers, (d) baboon images with 3×3 median filtering and corresponding extracted watermarks (e-h)

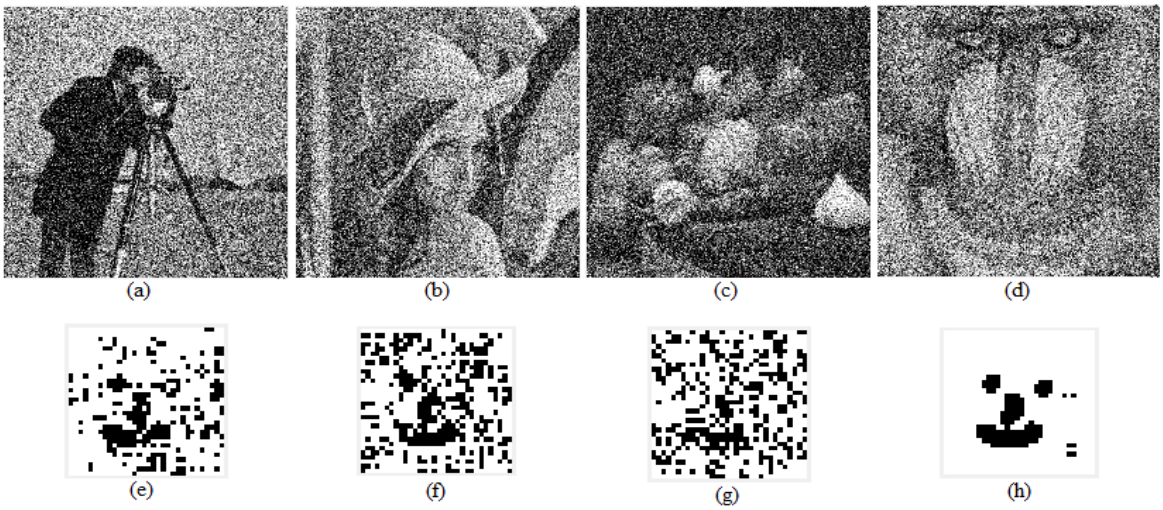

Fig. 19: Watermarked, (a) cameraman, (b) Lena, (c) peppers, (d) baboon images with Additive White Gaussian Noise (AWGN) having depth of 10 SNR and corresponding extracted watermarks (e-h)

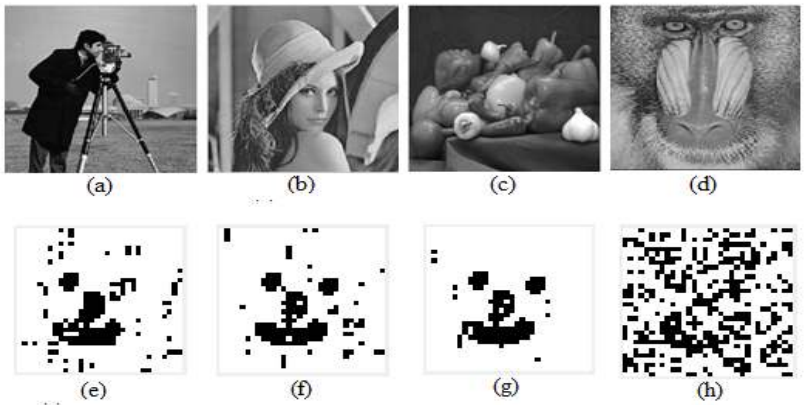

Fig. 20: Watermarked, (a) cameraman, (b) Lena, (c) peppers, (d) baboon images with scaling the image by 30% of the original size in both row wise and column wise directions and corresponding extracted watermarks (e-h)

Table 1: PSNR and NC of cameraman, Lena, pepper and baboon images against different attacks

Attack		Cameraman		Lena		Peppers		Baboon	
Name	Depth	PSNR	NC	PSNR	NC	PSNR	NC	PSNR	NC
JPEG attack (QF =)	20%	38.15	0.711	39.41	0.675	40.53	0.670	31.31	0.989
	30%	38.17	0.906	40.82	0.786	41.95	0.739	31.80	1.000
	40%	39.15	0.991	40.57	0.900	41.28	0.877	31.08	1.000
	50%	37.61	1.000	40.68	0.987	41.85	0.982	31.44	1.000
Median filter	3×3	39.26	0.868	41.89	0.929	42.32	0.926	32.51	0.740
	5×5	38.32	0.986	42.56	0.992	44.26	0.982	27.33	0.949
Impulse noise	5%	17.98	1.000	18.42	1.000	17.95	0.999	18.42	1.000
	10%	15.02	0.997	15.44	0.979	14.98	0.953	15.48	1.000
	15%	13.31	0.983	13.68	0.916	13.26	0.882	13.77	1.000
	20%	12.07	0.922	12.42	0.881	11.99	0.850	12.53	0.993
	25%	11.10	0.884	11.45	0.816	11.04	0.818	11.58	0.985
Speckle noise	5%	18.60	1.000	18.79	1.000	21.88	1.000	18.32	1.000
	10%	15.73	0.998	15.98	0.985	19.03	1.000	15.57	1.000
	15%	14.18	0.987	14.37	0.968	17.36	0.999	14.00	1.000
	20%	13.12	0.969	13.24	0.951	16.19	0.994	12.92	1.000
	25%	12.32	0.963	12.39	0.917	15.28	0.991	12.09	0.997
Gaussian noise	5%	13.85	0.965	13.62	0.959	14.02	0.933	13.48	1.000
mean = 0 and Var =	10%	11.51	0.909	11.33	0.839	11.68	0.855	11.22	0.994
	15%	10.33	0.851	10.21	0.818	10.46	0.824	10.13	0.958
AWGN (SNR =)	20 dB	19.89	1.000	19.92	1.000	19.94	1.000	19.74	1.000
	15 dB	14.96	0.993	14.99	0.993	14.98	0.988	14.91	1.000
	10 dB	9.99	0.934	10.00	0.858	10.00	0.821	9.980	0.993
Scaling	60%	46.23	0.997	48.43	1.000	49.10	1.000	35.36	0.904
	50%	47.99	0.984	50.67	0.989	51.49	0.998	35.20	0.826
	40%	44.08	0.987	48.38	0.989	50.04	0.998	29.79	0.850
	30%	39.41	0.961	45.27	0.976	48.39	0.991	26.62	0.806
	20%	32.98	0.835	38.44	0.863	42.81	0.858	23.75	0.723

Table 2: Comparison of PSNR and NC for Lena, pepper and baboon images against different attacks

		Hossein method (Hossein *et al*., 2010)						Proposed method					
Attack		Lena		Peppers		Baboon		Lena		Peppers		Baboon	
Name	Depth	PSNR	NC	PSNR	NC	PSNR	NC	PSNR	NC	PSNR	NC	PSNR	NC
Gaussian low pass	3×3	33.25	0.894	32.28	0.899	28.67	0.794	42.03	1.000	46.12	1.000	34.87	1.000
Average filter	5×5	29.60	0.857	29.08	0.827	24.34	0.717	37.77	1.000	38.37	1.000	31.98	1.000
Gaussian noise	M = 0 Var = 0.002	26.95	0.663	27.08	0.649	26.96	0.749	26.65	1.000	26.71	1.000	25.79	1.000
Blurring	3×3	29.12	0.940	28.81	0.940	23.89	0.883	37.77	1.000	38.37	1.000	31.98	1.000
Sharpen	Alpha = 1	25.34	0.730	25.15	0.714	21.10	0.503	27.11	1.000	29.96	1.000	20.59	1.000
Scaling	50%	33.94	0.952	31.85	0.940	29.29	0.886	50.67	0.989	51.49	0.998	35.20	0.825
Scaling	25%	28.80	0.714	28.00	0.693	23.58	0.407	44.27	0.933	47.83	0.949	25.11	0.755
Aspect ratio	X = 1, Y = 1.2	18.01	0.450	21.01	0.420	14.63	0.723	37.99	1.000	38.60	1.000	32.18	1.000
Aspect ratio	X = 0.8, Y = 1.0	27.50	0.673	26.49	0.580	26.36	0.566	45.16	1.000	45.74	1.000	37.98	1.000
Median filter	5×5	31.17	0.876	32.08	0.881	24.60	0.599	42.56	0.991	44.26	0.982	27.33	0.949

Table 3: Comparison of proposed method and Hossein method [] after multiple attacks in terms of PSNR and NC

	PSNR		Non correlation			
Attack type	Hossein method (Hossein *et al*., 2010)	Proposed method	Hossein method (Hossein *et al*, 2010)	Mohanty method (Mohanty *et al*, 2008)	Asatryan method (Asatryan *et al*, 2009)	Proposed method
No attack	51.72	38.37	0.9979	0.9974	0.9990	1.0000
AF (5×5) +JP (50)	29.80	41.84	0.5943	0.4233	0.6118	0.9832
S (1/2) + B (3) + JP (60)	29.35	46.08	0.5492	0.3711	0.4999	0.8155
SH (1) + MF (5×5)	29.52	38.44	0.5750	0.4277	0.6256	0.8399
MF (5×5) + S (1/2) + JP (60)	30.66	44.01	0.5490	0.4405	0.3304	0.8281
GN (0, 0.002) + MF (5×5) + JP (50)	30.34	38.38	0.4021	0.2832	0.4481	0.8059
GL (5×5) + CAR (1, 1.2) + JP (50)	31.79	42.12	0.7637	0.6008	0.4568	0.9856
WF (3×3) + B (2) + JP (40)	31.13	43.95	0.6697	0.5343	0.5928	0.6995
B (2) + GN (0, 0.002) + JP (50)	27.60	29.20	0.5152	0.3900	0.5222	1.0000
S (1/4) + JP (40)	29.06	43.28	0.6415	0.3827	0.3535	0.6784

50, 40, 30 and 20%, respectively of their actual size. Then they again resized back to the actual size of watermarked images. The PSNR and NC values are shown in Table 1 and watermarked images of Cameraman, Lena, Peppers and Baboon with 30% of scaling, 1024×1024 images were resized to 308×308 images and again to 1024×1024 image (a-d) and extracted watermarks (e-h) are shown in Fig. 20.

The results with various attacks are listed in Table 2 and are compared with the three state of the art methods mentioned in the table. Further, multiple attacks are performed on the images and results are

presented in Table 3. These results are compared with the Hossein method of watermarking.

CONCLUSION

We have presented an algorithm for digital image watermarking for gray scale images. The algorithm shows robustness against some common signal attacks like jpeg compression, median filtering, addition of AWGN, salt and pepper noise, speckle noise and scaling. The watermark was able to survive till the image gets badly affected for its quality, which is then for no use. This is due to the fact that, we inserted our watermark after selection of particular band of frequencies by using DWT and then applying DCT. Also, error correcting odes helps to reduce the error and in turn increases the non correlation. The algorithm proved to be efficient than the techniques compared in both individual attacks as well as multiple attacks.

REFERENCES

Abdul, W., P. Carre and P. Gaborit, 2009. List decoding of reed solomon codes for wavelet based colour image watermarking scheme. Proceeding of the 2009 16th IEEE International Conference on Image Processing (ICIP), pp: 3637-3640.

Akhaee, M.A., S.M.E. Sahraeian and C. Jin, 2011. Blind image watermarking using a sample projection approach. IEEE T. Inform. Forensics Sec., 6(3): 883-893.

Cika, P., 2009. Watermarking scheme based on discrete wavelet transform and error-correction codes. Proceeding of the 16th International Conference on Systems, Signals and Image Processing (IWSSIP 2009), pp: 1-4.

Dan, B. and S. James, 1998. Collusion-secure fingerprinting for digital data. IEEE T. Inform. Th., 44(5): 1897-1905.

Domingo-Ferrer, J. and J. Herrera-Joancomarti, 2000. Short collusion-secure fingerprints based on dual binary Hammin gcodes. Electr. Lett., 36(20): 1697-1699.

Guyeux, C. and J.M. Bahi, 2010. An improved watermarking scheme for internet applications. Proceeding of the 2010 2nd International Conference on Evolving Internet (INTERNET), pp: 119-124.

Hongtao, G., S. Fulin and Z. Yong, 2002. Color image text watermarking using wavelet transform and error-correcting code. Proceeding of the 2002 6th International Conference on Signal Processing, 2: 1584-1587.

Hossein, R., M. Reza and E.M. Mohsen, 2010. A new robust watermarking scheme to increase image security. EURASIP J. Adv. Signal Process., 2010: 1-30.

Jayalakshmi, M., S.N. Merchant and U.B. Desai, 2008. Optimum retrieval of watermark from wavelet significant coefficients. Inform. Secur., 2(4): 119-128.

Kung, C.M. and T.K. Troung, 2006. Visual robust oblivious watermarking technique using error correcting code. Proceeding of the International Conference on Communication Technology (ICCT '06), pp: 1-4, 27-30 2006.

Limin, G., H. Jiwu and Y.Q. Hi, 2003. Analysis of the role played by error correcting coding in robust watermarking. Proceedings of the 2003 International Symposium on Circuits and Systems (ISCAS '03), 3: III-798-III-801.

Man, X. and J. Jian-Guo, 2009. A novel watermarking algorithm in entropy coding based on image complexity analysis. Proceedings of the International Conference on Multimedia Information Networking and Security (MINES '09), 2L18-20: 126-129.

Nayak, J., P.S. Bhat, M.S. Kumar and U.R. Acharya, 2004. Reliable transmission and storage of medical images with patient information using error control codes. Proceedings of the 1st IEEE Indicon 2004 India Annual Conference, pp: 147- 150.

Seung, W.J., T.H. Le and K. Sung-Jea, 2011. A new histogram modification based reversible data hiding algorithm considering the human visual system. Signal Process. Lett., 18(2): 95-98.

Shijun, X., J.K. Hyoung and H. Jiwu, 2008. Invariant image watermarking based on statistical features in the low-frequency domain. IEEE T. Circuits Syst. Video Technol., 18(6): 777-790.

Solanki, K., N. Jacobsen, U. Madhow, B.S. Manjunath and S. Chandrasekaran, 2004. Robust image-adaptive datahiding using erasure and error correction. IEEE T. Image Process., 13(12): 1627-1639.

Terzija, N., M. Repges, K. Luck and W. Geisselhardt, 2002. Digital image watermarking using discrete wavelet transform: Performance comparison of error correction codes. Proceeding of Visualization, Imaging and Image Processing.

Xiaolong, L., Y. Bin and Z. Tieyong, 2011. Efficient reversible watermarking based on adaptive prediction-error expansion and pixel selection. IEEE T. Image Process., 20(12): 3524-3533.

Yongqing, X., Miros\&\#322 and P. Aw, 2008. M-ary phase modulation for digital watermarking. Int. J. Appl. Math. Comput. Sci., 18(1): 93-104.

Zhang, Y., L.C. Lin, Q.S. Li and ZT. Jiang, 2009. A blind watermarking algorithm based on block DCT for dual color images. Proceedings of the 2nd International Symposium on Electronic Commerce and Security (ISECS '09), 1: 213-217.