http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 13 (16): 2534-2543, 2014
ISSN 1812-5638 / DOI: 10.3923/itj.2014.2534.2543

Comparative Analysis of (5/3) and Haar IWT Based Steganography

[1]V. Thanikaiselvan, [1]P. Arulmozhivarman, [1]Siddhanta Chakrabarty, [1]Ashutosh Agarwa1,
[2]S. Subashanthini and [3]Rengarajan Amirtharajan
[1]School of Electronics Engineering,
[2]School of Information Technology, VIT University, Vellore, Tamil Nadu, 632014, India
[3]School of Electrical and Electronics Engineering, SASTRA University, Thanjavur, 613401, India

Abstract: Steganography is the technique of hiding information inside other information. It provides data security. Image steganography can be implemented in both, the spatial and transform domain. In this study, transform domain steganography has been adopted. A cover image is transformed to the frequency domain using Integer Wavelet Transform (IWT) and a secret image is embedded in it using Least Significant Bit (LSB) substitution. The secret data is embedded only in high frequency sub-bands of the frequency domain transform of the cover image. Both adaptive and non-adaptive embedding techniques are employed and the results are compared. Also, random traversing for embedding the secret data is implemented for higher security. Haar and (5/3) IWT based algorithms are used. The Peak Signal to Noise Ratio (PSNR) values and payload capacity are obtained and compared for above algorithms.

Key words: Security, steganography, IWT, LSB, random traversing, adaptive embedding

INTRODUCTION

With the susceptibility of electronic information to attack and malicious distortion, a need for security of electronic data has become imperative. Steganography conceals secret information within some 'cover' media. This aids in hiding the very existence of sensitive information from malicious users. Steganography can be applied to various fields such as text, image, audio and video. Text steganography is the oldest form of information hiding. Essentially, a secret message was hidden in an otherwise harmless message. For example, every nth letter of a sentence or paragraph spelled out a message. However, text steganography greatly limits the type and amount of information that can be concealed. Image steganography hides information within an image called the 'cover image'. The hidden information can be another image, text or any other type of data (Amirtharajan and Rayappan, 2013; Amirtharajan *et al*., 2013a-j; Cheddad *et al*., 2010).

Image steganography can be carried out either in the spatial domain (image domain) (Amirtharajan and Rayappan, 2013; Amirtharajan *et al*., 2013a-j; Chan and Cheng, 2004; Cheddad *et al*., 2010; Dey *et al*., 2011; Janakiraman *et al*., 2013, 2014a, b; Nithyanandam *et al*., 2011) or in the transform (frequency) domain (Le Gall and Tabatabai, 1988; El-Safy *et al*., 2009; Peng *et al*., 2012; Nag *et al*., 2010, 2011; Thanikaiselvan *et al*., 2011, 2012a, b, c, 2013a, b; Wong *et al*., 2007). Numerous systems exist for both methods for communication information security (Praveenkumar *et al*., 2014a-l, 2012a, b, 2013a, b; Rajagopalan *et al*., 2014a-d; Ramalingam *et al*., 2014a, b). Transform domain steganography provides higher security and better hiding capabilities with less degradation to the cover image. Spatial domain steganography can usually be detected by statistical means if a large number of pixels of the image have been modified to contain secret information. Dey *et al*. (2011) and Nag *et al*. (2010, 2011) have used a transform based scheme. The former has observed PSNR values below 30 dB which makes the encoding vulnerable to steganalysis attacks. Nag *et al*. (2010, 2011) and Nithyanandam *et al*. (2011) have employed Huffman coding procedure for added security.

A common and effective method for information hiding proposed by Chan and Cheng (2004) is simple and easy to use yet highly efficient. The LSB substitution has been adopted in this study too, following Chan and Cheng (2004). Thanikaiselvan *et al*. (2012b) proposed a random traversing method which echoed the movements of a knight on a chess board without effecting image quality and enhancing security. Adaptive steganography has been explored by (El-Safy *et al*., 2009; Peng *et al*., 2012). El Safy *et al*'s method is effective while being low

Corresponding Author: V. Thanikaiselvan, School of Electronics Engineering, VIT University, Vellore, Tamil Nadu, 632014, India Tel: +91 4362 264101 Fax: +91 4362 264120

in complexity. This study uses modified basic adaptive scheme to implement adaptive steganography. Transform domain steganography (Ramalingam *et al*., 2014a, b; Thanikaiselvan *et al*., 2011, 2012b, c, 2013a, b) provides high security. The DCT (Discrete Cosine Transform) based steganography is quite common (Fazli *et al*., 2010; Kumar *et al*., 2010; Sarita and Choudhary, 2012).

In this study, transform domain steganography techniques are proposed and their efficiency and suitability is compared for use in steganography. The transformations that are compared are Haar integer wavelet transform and (5/3) integer wavelet transform. Both adaptive and non-adaptive techniques to embed the secret information are also compared. Additionally, random traversing and block-coding have also been adopted. The embedding is done by LSB substitution. Block-coding helps to increase security.

MATERIALS AND METHODS

(5/3) Integer wavelet transform: Wavelets are basis functions used to represent signals. Integer Wavelet Transforms (IWT) are the wavelet transforms that map integers to integers. Every sub-band or wavelet transformation is connected with filters of finite length which can be obtained as integer wavelet succeeded by lifting steps. The integer wavelet divides the signal into even and odd samples. A wavelet transform that maps integers to integers can be obtained by combining the lifting steps and rounding off. Didier Le Gall and Tabatabai (1988) developed a new and efficient sub-band coding of digital images. Out of the two solutions provided for the design of symmetric short tap filters, the second one produces visually pleasant and smooth outputs. This filter has unequal lengths for the high pass and low pass coefficients 5 and 3, respectively. Thus, the name (5/3) filter is obtained.

Let d(n) contain the odd samples and s(n) contain the even samples. The odd samples are replaced by Eq. 1 (prediction step) and then the even samples are replaced by Eq. 2 (updating step):

$$d(n) = d(n) - \left\lfloor \frac{1}{2}\left(s(n) + s(n+1)\right) + \frac{1}{2} \right\rfloor \quad (1)$$

$$s(n) = s(n) - \left\lfloor \frac{1}{4}\left(d(n-1) + d(n)\right) + \frac{1}{2} \right\rfloor \quad (2)$$

The even samples s(n) give the high frequency wavelet coefficients and odd samples d(n) give the low frequency wavelet coefficients.

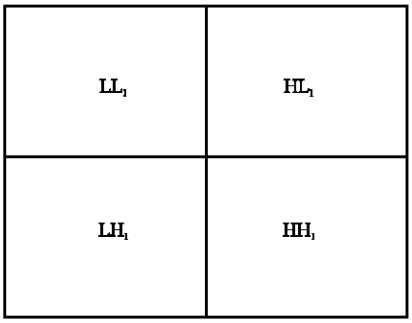

Fig. 1: Transform domain sub-bands

The wavelet transform creates 4 sub-bands as shown in the Fig. 1. Secret information is embedded in the HH (High High), HL (High Low) and LH (Low High) bands. The low frequency LL (Low Low) bands contain the approximations.

The inverse of this transformation is reversible. It is easily implemented by inverting the Eq. 1 and 2 by making the d(n) and s(n) on the right hand side of the equations and following the above steps in reverse, for example, instead of splitting there will be merging.

This process was adapted for 2 dimensional samples by following the usual procedure, that is, the above equations were first applied to the image matrix row wise, treating them as 1-D samples and then the result was subjected to the equations column wise. This gave us the final transformed matrix with 4 sub-bands.

Haar transform: The Haar wavelet is a sequence of rescaled "square-shaped" functions which together form a wavelet family or basis.

The image matrix is separated into even and odd columns basis and F-high (High frequency components) and F-low (low frequency components) are found using the following equation:

$$\text{F-high} = \text{odd}(i,j) - \text{even}(i,j) \quad (3)$$

$$\text{F-low} = \text{even}(i,j) + \text{floor}(\text{F-high}/2) \quad (4)$$

L-even, L-odd, H-even, H-odd matrix is developed:

$$\text{L-even} = \text{Even rows from F-low} \quad (5)$$

$$\text{L-odd} = \text{Odd rows from F-low} \quad (6)$$

$$\text{H-even} = \text{Even rows from F-high} \quad (7)$$

$$\text{H-odd} = \text{Odd rows from F-high} \quad (8)$$

First level decomposition of the image can be obtained through the following equations:

$$HH1 = h(odd)-h(even) \quad (9)$$

$$HL1 = h(even)+floor\ (hh/2) \quad (10)$$

$$LH1 = l(odd)-l(even) \quad (11)$$

$$LL1 = l(even)+floor\ (lh/2) \quad (12)$$

where, HH1 is the diagonal co-efficients, HL1 is the horizontal co-efficients, LH1 is the vertical co-efficients and LL1 is the approximation coefficients.

Adaptive embedding: In this study, the information is embedded using both methods adaptive and non-adaptive and their results are compared. Non adaptive embedding refers to the process where a fixed number of message bits (either 1, 2, 3 or 4) are embedded in each coefficient. In adaptive embedding, on the other hand, each coefficient is analyzed and depending on its value the number of bits that can be substituted is determined. The equation which determines the number of bits as suggested by El-Safy *et al.* (2009), is given below:

$$L = \begin{cases} K+3, & \text{if } C_o \geq 2^{k+3} \\ k+2, & \text{if } 2^{k+2} \leq C_o < 2^{k+3} \\ 2k+1, & \text{if } 2^{k+1} \leq C_o < 2^{k+2}; \quad 0 \leq k \leq 4 \\ k, & \text{if } C_o < 2^{k+1} \end{cases} \quad (13)$$

where, L is the number of bits to be embedded, k is the minimum number of bits to be embedded and C_o is the transform domain coefficient.

In this study, the above equation is slightly modified for comparison. In Eq. 13, assuming k = 1 and L = 4 if C_o is greater than equal 2^4 and so on. The modification proposed is that instead of 2^4 following equation uses 2^6:

$$L = \begin{cases} K+3, & \text{if } C_o \geq 2^{5} \\ k+2, & \text{if } 2^{k+4} \leq C_o < 2^{k+5} \\ 2k+1, & \text{if } 2^{k+3} \leq C_o < 2^{k+4}; \quad 0 \leq k \leq 4 \\ k, & \text{if } C_o < 2^{k+3} \end{cases} \quad (14)$$

This modification increases the PSNR value and improves the image quality. However, the payload capacity is significantly affected. Proposed work entitled the adaptive scheme implementing Eq. 13 as 'adap1' and the one implementing Eq. 14 as 'adap3'. This is because, taking the example of the last case, in Eq. 13 one LSB bit is replaced if the coefficient is of two bits or less. This leaves, at most, one MSB (Most Significant Bit) untouched, thus the name 'adap1'. In Eq. 14, one LSB is replaced if the coefficient value is of four bits or less, thus leaving a maximum of three MSB untouched and hence 'adap3'.

Random traversing: A random traversing algorithm has been adopted to enhance security of the hidden information. Randomness has been employed in 2 stages. In the first stage, the blocks in which the image was divided are chosen randomly. For example, consider an image with dimensions 512×512. Assume that, it is divided into blocks of size 128×128. This implies that the image is divided into 16 blocks each of size 128×128. The first stage of random traversing randomizes the sequence of the blocks before embedding, so that the hidden information is spread all over the final image in a random fashion.

The second stage involves randomization of the coefficients within a particular block. A block consists of the 4 transform domain sub-bands; HH, HL, LH and LL. Embedding is done in 3 of those sub-bands excluding LL. Hence, the second stage random traversing randomizes the sequence of coefficients that are selected for embedding within a particular block. This random pattern, however, remains constant for every block. Both random patterns are stored separately and these serve as the "keys" with which the information can be extracted. Without these "keys" extraction of information will result in meaningless or incorrect data. Consequently the receiver needs to possess the keys beforehand in order to 'unlock' or decrypt the information.

PROPOSED ALGORITHM

Embedding process: The embedding process refers to the hiding of the secret information in the cover image and the steps that precede it for the preparation of the cover image.

Embedding has been done by LSB substitution using the equation given below. This formula was suggested by Chan and Cheng (2004). It is simple yet efficient:

$$C(i,j) = C(i,j)\text{-mod}\ (C(i,j), 2^k)+m(l) \quad (15)$$

where, C is the transform domain coefficient, m is the segment of the secret information (in base 10) to be embedded in this coefficient and k denotes the number of bits to be replaced.

The steps are enumerated below:

- **Step 1:** Read the cover image
- **Step 2:** Split it into blocks. The block sizes can be any of 8×8, 16×16, 32×32, 64×64, 128×128, 256×256 or 512×512

- **Step 3:** Transform each block using IWT. (Haar or (5/3))
- **Step 4:** Prepare the keys for random traversing. The keys are the random patterns according to which the blocks are traversed and also the random order in which the blocks are selected
- **Step 5:** For adaptive embedding, run the analysis to determine number of bits for each coefficient
- **Step 6:** Read secret message or information in binary
- **Step 7:** Traverse the blocks according to the keys and insert the information using Eq. 15; adaptively or non-adaptively as the case may be using Eq. 13-14, respectively
- **Step 8:** Once all the information has been inserted by LSB substitution, perform inverse IWT on each block to transform the image back to spatial domain
- **Step 9:** The image now carries the secret information hidden safely inside and can now be used for transmission or other purposes. This image is called the stego image

Extraction process: The process to extract the data is almost same as that for embedding. Extraction is implemented by the following Eq. 16:

$$m(l)=\text{mod}\,(C(i,j), 2^k) \tag{16}$$

where, the variables denote the same values as in Eq. 15.

The steps followed in the extraction process are:

- **Step 1:** Read the stego image
- **Step 2:** Split it into blocks. The block sizes can be any of 8×8, 16×16, 32×32, 64×64, 128×128, 256×256 or 512×512
- **Step 3:** Transform each block using IWT (Haar or (5/3))
- **Step 4:** It is assumed that the keys used to determine the traversing pattern are available with the receiver. Traverse the blocks using the keys
- **Step 5:** Apply Eq. 16 on each coefficient. 'k' is fixed if it is non-adaptive and for adaptive it is obtained through Eq. 13-14. This gives us the hidden message

RESULTS AND DISCUSSION

The main feature of steganography is that the stego image should hold up to visual scrutiny. However, mathematical methods also exist to compare the performance or quality of the embedding algorithm. The PSNR and MSE (Mean Square Error) are widely used in steganographic algorithms for this purpose. The PSNR stands for peak signal to noise ratio. It is the ratio of the square of the maximum value that the signal is allowed to have to the noise MSE. The MSE is a cumulative squared error between the original image and stego image:

$$\text{MSE} = \frac{1}{nm}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I(i.j) - K(i,j)]^2 \tag{17}$$

$$\text{PSNR} = 10\log_{10}\frac{(255)^2}{\text{MSE}} \tag{18}$$

where, I denotes the original image, K denotes the stego image and n and m are the image dimensions.

The performance and capacity of (5/3) IWT and Haar transform are compared in adaptive and non-adaptive schemes with varying block sizes. The PSNR values and embedding capacities have been recorded. They have been organized into 2 main categories, those obtained by (5/3) IWT and those obtained by Haar transform. Under these two headings, they have been further divided into non-adaptive and adaptive embedding. Within each of these types, classifications exist based on the number of bits embedded; 1 bit, 2 bit etc., for non-adaptive and adap1 and adap3 for adaptive. The results obtained are given below with Lena image used as a sample and shown in Fig. 2. The image is compared before and after embedding the secret information.

The PSNR values of all different types of steganography explored in this project are recorded below. All images have dimensions 512×512. 'Globe' is the only RGB color image, all others are grayscale. It is clear from the tables that block sizes have negligible effect on PSNR values. Also, the variation of PSNR value between different images under any one particular type of steganography is very little. Table 1 and 2 record the PSNR values of non-adaptive embedding and Table 3-6 record the PSNR values as well embedding capacity of adaptive embedding.

For easier comparison of the PSNR values and number of bits embedded, figure were plotted with the values from the tables above for the 'Lena' image. From the approximately straight lines of all configurations, it is clear that block sizes have very little influence on PSNR values. Analyzing Fig. 3, for 1 bit LSB embedding for non-adaptive (5/3) IWT system gives the best PSNR and 'adap1' (5/3) IWT gives the lowest. The 1 bit non-adaptive embedding and 'adap3' with Haar transform take the second and third places respectively. They are followed by 2 bit with (5/3) IWT and 2 bit with Haar IWT. Below the 40dB threshold, in this order, are 3 bit with (5/3) IWT, ' adap1' with Haar, 3 bit with Haar, 'adap3 (5/3) with IWT and 'adap1' with (5/3) IWT. The 4 bit LSB

Fig. 2(a-g): (a) Original Lena image, (b) Haar transform with non-adaptive embedding (1 bit), (c) (5/3) IWT with non-adaptive embedding (1 bit), (d) Haar transform adaptive with 'adap1', (e) (5/3) adaptive with 'adap1', (f) Haar transform adaptive with 'adap3' and (g) (5/3) adaptive with'adap3'

Table 1: PSNR (dB) values of non-adaptive embedding with (5/3) IWT

	Block size						
Images	8×8	16×16	32×32	64×64	128×128	256×256	512×512
1 bit LSB: Total bits embedded = 196608							
Lena	50.9608	50.9495	50.9521	50.9461	50.9342	50.9692	50.9376
Lifting body	50.9862	50.9506	50.9541	50.9424	50.9377	50.9307	50.9361
Einstein	50.9264	50.8745	50.8907	50.8764	50.8589	50.8631	50.8439
Man	50.9020	50.8993	50.8971	50.9092	50.8949	50.8864	50.9023
Globe	50.7561	50.7175	50.7318	50.7307	50.7347	50.7378	50.7287
2 bit LSB: Total bits embedded = 393216							
Lena	45.1044	45.0934	45.0845	45.0805	45.0979	45.0991	45.0785
Lifting body	45.1050	45.0574	45.0838	45.0731	45.0917	45.0766	45.0881
Einstein	45.1092	45.0712	45.0952	45.0749	45.0890	45.0819	45.0827
Man	44.9254	44.9109	44.9234	44.9354	44.9317	44.9426	44.9282
Globe	44.6756	44.6641	44.6660	44.6591	44.6719	44.6756	44.6550
3 bit LSB: Total bits embedded = 786432							
Lena	39.1558	39.1399	39.1248	39.1358	39.1252	39.1428	39.1397
Lifting body	39.1048	39.1014	39.0836	39.0792	39.0804	39.0824	39.0739
Einstein	39.0317	39.0621	39.0569	39.0537	39.0555	39.0425	39.0551
Man	38.7655	38.7589	38.7800	38.7851	38.7846	38.7676	38.7760
Globe	38.4517	38.4373	38.4310	38.4471	38.4449	38.4307	38.4351
4 bit LSB: Total bits embedded = 786432							
Lena	32.9658	32.9826	32.9596	32.9421	32.9643	32.9644	32.9348
Lifting body	32.6879	32.6947	32.6651	32.6639	32.6806	32.6683	32.6707
Einstein	32.7709	32.7576	32.7583	32.7597	32.7630	32.7403	32.7479
Man	32.5228	32.5021	32.5050	32.5011	32.5047	32.5117	32.5208
Globe	32.3236	32.3074	32.2956	32.3023	32.3198	32.2950	32.3130

Table 2: PSNR (dB) values of non-adaptive embedding with Haar transform

	Block size						
Images	8×8	16×16	32×32	64×64	128×128	256×256	512×512
1 bit LSB: Total bits embedded = 196608							
Lena	48.9949	49.0218	49.0266	49.0083	49.0294	49.0372	49.0540
Lifting body	49.0424	49.0333	49.0517	49.0370	49.0157	49.0371	49.0198
Einstein	48.9836	49.0145	48.9916	48.9818	48.9719	48.9972	49.0039
Man	48.5399	48.5817	48.5553	48.5732	48.5493	48.5463	48.5445
Globe	48.5676	48.5959	48.5921	48.5866	48.5836	48.5853	48.5934
2 bit LSB: Total bits embedded = 393216							
Lena	43.2729	43.2128	43.2239	43.2265	43.2273	43.2347	43.2389
Lifting body	43.2730	43.2534	43.2912	43.2730	43.2440	43.2439	43.2644
Einstein	42.8142	42.7869	42.7747	42.7958	42.8042	42.7860	42.8042
Man	41.9305	41.9273	41.9344	41.9348	41.9287	41.9177	41.9198
Globe	42.1675	42.1819	42.1684	42.1836	42.1590	42.1645	42.1696
3 bit LSB: Total bits embedded = 786432							
Lena	37.2748	37.2890	37.2574	37.2834	37.2588	37.2718	37.2656
Lifting body	37.2350	37.2279	37.2590	37.2698	37.2577	37.2503	37.2638
Einstein	36.1649	36.1534	36.1889	36.1720	36.1350	36.1650	36.1639
Man	35.3122	35.3212	35.3059	35.2911	35.3269	35.3077	35.3058
Globe	35.7702	35.7564	35.7830	35.7713	35.7535	35.7583	35.7695
4 bit LSB: Total bits embedded = 786432							
Lena	31.1320	31.1217	31.1056	31.1465	31.1146	31.1105	31.1318
Lifting body	31.0151	31.0397	31.0445	31.0278	31.0377	31.0458	31.0315
Einstein	29.5965	29.6022	29.6106	29.6142	29.5962	29.6137	29.5839
Man	28.8751	28.8835	28.8857	28.9055	28.8949	28.8700	28.8776
Globe	29.5432	29.5334	29.5348	29.5380	29.5492	29.5555	29.5348

Table 3: PSNR (dB) values of adaptive embedding (adap1) with (5/3) IWT

	Lena		Einstein		Man		Lifting body		Globe	
Block size	PSNR	Bits	PSNR	Bits	PSNR	Bits	PSNR	Bits	PSNR	Bits
8×8	35.3938	521599	35.7103	461341	35.5315	461229	35.6976	455950	36.2118	486553
16×16	35.4299	516040	35.7907	451191	35.6044	453829	35.7705	447749	36.2398	483375
32×32	35.4304	513003	35.8065	447024	35.6127	450691	35.7676	445505	36.2493	482048
64×64	35.4666	510993	35.8356	444818	35.6345	448963	35.7764	444638	36.2500	481169
128×128	35.4969	509635	35.7946	443670	35.6290	447887	35.7560	444161	36.2513	480669
256×256	35.5139	509022	35.8250	442909	35.6544	446944	35.7885	443935	36.2739	480462
512×512	35.4989	508560	35.7964	442641	35.6694	446592	35.8148	443330	36.2847	478801

Table 4: PSNR (dB) values of adaptive embedding (adap1) with Haar transform

	Lena		Einstein		Man		Lifting body		Globe	
Block size	PSNR	Bits	PSNR	Bits	PSNR	Bits	PSNR	Bits	PSNR	Bits
8×8	38.2820	408038	44.9376	251068	41.5436	287280	42.8952	295660	36.3191	434125
16×16	38.3230	408038	44.9396	251068	41.5345	287280	42.9227	295660	36.3891	434125
32×32	38.3558	408038	44.9178	251068	41.5484	287280	42.9089	295660	36.3755	434125
64×64	38.3228	408038	44.9149	251068	41.5195	287280	43.0059	295660	36.3386	434125
128×128	38.3303	408038	44.9176	251068	41.5046	287280	42.9350	295660	36.3432	434125
256×256	38.3343	408038	44.9077	251068	41.5166	287280	42.9606	295660	36.3310	434125
512×512	38.3486	408038	44.9743	251068	41.5155	287280	42.9182	295660	36.3918	434125

Table 5: PSNR (dB) values of adaptive embedding (adap3) with (5/3) IWT

	Lena		Einstein		Man		Lifting body		Globe	
Block size	PSNR	Bits	PSNR	Bits	PSNR	Bits	PSNR	Bits	PSNR	Bits
8×8	37.0151	391448	36.3016	400505	36.2470	395375	35.9398	409111	37.4034	387323
16×16	37.0666	386769	36.3553	393569	36.3031	388168	36.0146	400869	37.4414	383163
32×32	37.0816	384521	36.3068	391532	36.3531	386157	36.0072	398715	37.4233	381799
64×64	37.0738	383313	36.3211	390705	36.3249	385052	36.0558	398101	37.4310	381056
128×128	37.0842	382159	36.3222	390469	36.3397	384457	36.0147	397767	37.4557	380727
256×256	37.0999	381685	36.3645	390214	36.3682	383662	36.0678	397633	37.4336	380575
512×512	37.0914	381434	36.3551	390127	36.3312	383227	36.0548	397281	37.4517	379279

substitution has been excluded from the figure as the PSNR values are below 35 and hence, the method loses significance as a strong security measure.

On analysis of Fig. 4, it is quite obvious that in non-adaptive embedding, the embedding capacity does not vary based on block size or transform technique.

Table 6: PSNR (dB) values of adaptive embedding (adap3) with haar transform

	Lena		Einstein		Man		Lifting body		Globe	
Block size	PSNR	Bits	PSNR	Bits	PSNR	Bits	PSNR	Bits	PSNR	Bits
8×8	46.3776	230820	48.6896	200483	47.3568	211440	48.2286	203647	42.2405	287355
16×16	46.3366	230820	48.6591	200483	47.3897	211440	48.2686	203647	42.3140	287355
32×32	46.3298	230820	48.6731	200483	47.3458	211440	48.2217	203647	42.3181	287355
64×64	46.3964	230820	48.6877	200483	47.3878	211440	48.2782	203647	42.2602	287355
128×128	46.4109	230820	48.7126	200483	47.3719	211440	48.2077	203647	42.3321	287355
256×256	46.3282	230820	48.6948	200483	47.3008	211440	48.2435	203647	42.2821	287355
512×512	46.3346	230820	48.6648	200483	47.3540	211440	48.2383	203647	42.2775	287355

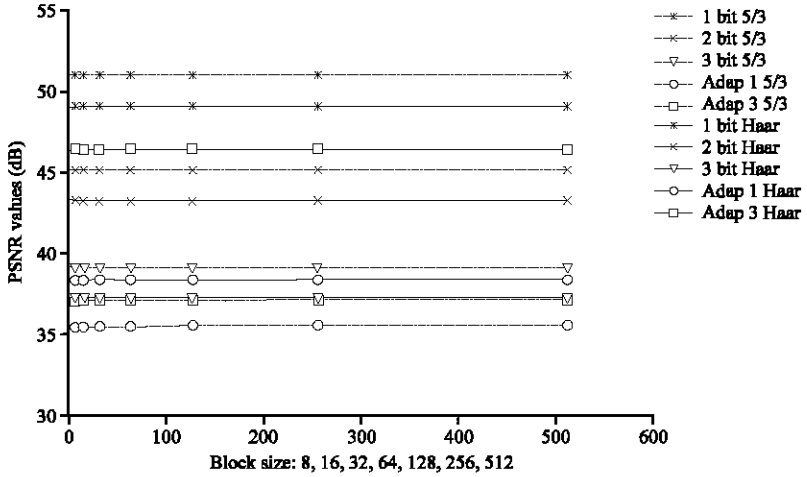

Fig. 3: Comparison of PSNR values (with sample image lena)

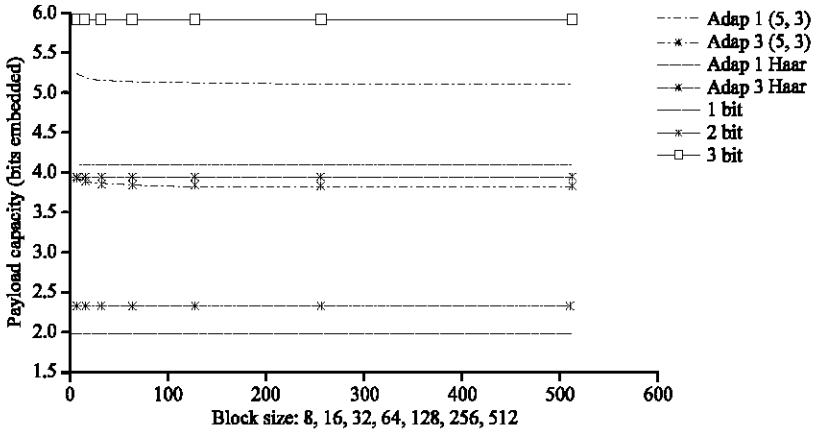

Fig. 4: Comparison of payload capacity

Payload capacity also remains constant in the case of adaptive embedding with Haar transform. However, for adaptive embedding with (5/3) IWT it varies with block size. This difference between the 2 transform techniques is due to the reason that Haar transform calculations involve the current value whereas (5/3) IWT calculations require previous and next values too. Therefore, with varying block sizes these previous and next values change, thus giving different coefficients associated with each block size. However, for block sizes greater than 64, the capacity becomes almost constant. Of course, 'adap1' has higher capacity than 'adap3' by virtue of their natures. Also, it is quite clear that (5/3) IWT provides higher embedding capacity than Haar transform for adaptive embedding.

Table 7: Comparative analysis

Method	Total number of iterations to extract data (TI)
Proposed method	(1024!*3!*64!*4*64*3)*1024
Fazli *et al.* (2010)	TI = 256*256*3
Song *et al.* (2012)	TI = 64!*64
Sakkara *et al.* (2012)	TI = 16!*64*64*3

Steganalysis: Steganalysis is the blind extraction of secret data in stego images. This proposed method is highly secured and robust against blind steganalysis or attacks. This method has been done in transform domain method. Therefore, secret data cannot be extracted from the spatial domain. There are three keys used for embedding (Key 1, 2 and 3) these keys impart high randomness for embedding. Following number of iterations are required to extract the hidden information from the stego image generated by the proposed method with block size of n×n. Complexity of the extraction based on the block size only.

Generalized Total number of Iterations (GTI):

$$GTI = [(\frac{512\times512}{n\times n})!\times\ 3!\times\ (\frac{n\times n}{4})!\times\ 4\times(n\times n)\times3]\times(\frac{512\times512}{n\times n}) \quad (19)$$

For example 16×16 (n = 16) block is considered and substituted for the above equation, then the total number of Iterations (TI):

$$TI = (1024!*3!*64!*4*64*3)*1024$$

Where:

1024! = Possible order of traversal among 1024 number of 16×16 sized blocks
3! = Gives the possible order of the subbands into which the data is embedded
64! = Represents different random traversing on a subband
4 = Represents the maximum bit length.
64 = Represents the total number of co-efficients in each subband
3 = Represents the total number of subbands
1024 = Represents the total number of 16×16 blocks

The proposed methodology is compared with the existing techniques and the TI values are tabulated in Table 7. Thus the large difference observed in the total number of iterations required for the proposed technique and the existing technique clearly elucidate the enhanced data security against blind attacks.

CONCLUSION

In this study, secret information embedded inside an image using 2 different transform techniques, namely Haar transform and (5/3) IWT with adaptive and non-adaptive technique. The results of the combinations of all 4 configurations were observed, recorded and compared. If PSNR value is important to the user then (5/3) IWT is the better choice for non-adaptive embedding and for adaptive, it is haar transform. If payload capacity is more important than PSNR values, then the reverse is true for adaptive embedding. The advantage of (5/3) transform over Haar transform is its relatively simple equations which reduce computation time and complexity. The only disadvantage that (5/3) IWT faces is its dependence on next and previous values which varies the transform domain output when the block sizes are varied. Haar transform on the other hand, gives uniform output for all block sizes. For future study, other transform methods, like (9/7) IWT can be compared with Haar transform and (5/3) IWT.

REFERENCES

Amirtharajan, R. and J.B.B. Rayappan, 2013. Steganography-time to time: A review. Res. J. Inform. Technol., 5: 53-66.

Amirtharajan, R., G. Devipriya, V. Thamikaiselvan and J.B.B. Rayappan, 2013a. High capacity triple plane embedding: A colour stego. Res. J. Inform. Technol., 5: 373-382.

Amirtharajan, R., K. Karthikeyan, M. Malleswaran and J.B.B. Rayappan, 2013b. Kubera kolam: A way for random image steganography. Res. J. Inform. Technol., 5: 304-316.

Amirtharajan, R., K.M. Ashfaaq, A.K. Infant and J.B.B. Rayappan, 2013c. High performance pixel indicator for colour image steganography. Res. J. Inform. Technol., 5: 277-290.

Amirtharajan, R., M.V. Abhiram, G. Revathi, J.B. Reddy, V. Thamikaiselvan and J.B.B. Rayappan, 2013d. Rubik's cube: A way for random image steganography. Res. J. Inform. Technol., 5: 329-340.

Amirtharajan, R., P. Archana and J.B.B. Rayappan, 2013e. Why image encryption for better steganography. Res. J. Inform. Technol., 5: 341-351.

Amirtharajan, R., P.S. Priya and J.B.B. Rayappan, 2013f. Pixel indicated user indicator: A muxed stego. Res. J. Inform. Technol., 5: 73-86.

Amirtharajan, R., R. Subrahmanyam, J.N. Teja, K.M. Reddy and J.B.B. Rayappan, 2013g. Pixel indicated triple layer: A way for random image steganography. Res. J. Inform. Technol., 5: 87-99.

Amirtharajan, R., S. Sulthana and J.B.B. Rayappan, 2013h. Seeing and believing is a threat: A visual cryptography schemes. Res. J. Inform. Technol., 5: 435-441.

Amirtharajan, R., S.D. Roy, N. Nesakumar, M. Chandrasekar, R. Sridevi and J.B.B. Rayappan, 2013i. Mind game for cover steganography: A refuge. Res. J. Inform. Technol., 5: 137-148.

Amirtharajan, R., V. Rajesh, P. Archana and J.B.B. Rayappan, 2013j. Pixel indicates, standard deviates: A way for random image steganography. Res. J. Inform. Technol., 5: 383-392.

Chan, C.K. and L.M. Cheng, 2004. Hiding data in images by simple LSB substitution. Pattern Recognit., 37: 469-474.

Cheddad, A., J. Condell, K. Curran and P. Mc Kevitt, 2010. Digital image steganography: Survey and analysis of current methods. Signal Process., 90: 727-752.

Dey, N., A. B. Roy and S. Dey, 2011. A novel approach of color image hiding using RGB color planes and DWT. Int. J. Comput. Applic., 36: 19-24.

El-Safy, R.O., H.H. Zayed and A. EI-Dessouki, 2009. An adaptive steganographic technique based on integer wavelet transform. Proceedings of the International Conference on Networking and Media Convergence, March 24-25, 2009, Cairo, Egypt, pp: 111-117.

Fazli, S., S. Gholamrezaei and A. Bazrafshan, 2010. Advanced wavelet based *Steganography* for colored images. Proceedings of the International Congress on Ultra Modern Telecommunications and Control Systems and Workshops, October 18-20, 2010, Moscow, pp: 377-380.

Janakiraman, S., S. Rajagopalan, K. Thenmozhi, H.N. Upadhyay and J. Ramanathan *et al*., 2013. Captivating CODEC Stego (CCS): A cover on camouflage. Res. J. Inform. Technol., 5: 160-170.

Janakiraman, S., J. Chakravarthy, B. Radhakrishnan, K. Thenmozhi, J.B.B. Rayappan and R. Amirtharajan, 2014a. Cover as key and key as data: An inborn stego. Inform. Technol. J., 13: 1969-1976.

Janakiraman, S., K.V.S.K. Kumar, R.R.K. Reddy, A. Srinivasulu, R. Amirtharajan, K. Thenmozhi and J.B.B. Rayappan, 2014b. Humming bird with coloured wings: A feedback security approach. Inform. Technol. J., 13: 2022-2026.

Kumar, K.B.S., K.B. Raja, R.K. Chhotaray and S. Pattanaik, 2010. Bit length replacement steganography based on DCT coefficients. Int. J. Eng. Res. Technol., 2: 3561-3570.

Le Gall, D. and A. Tabatabai, 1988. Sub-band coding of digital images using symmetric short kernel filters and arithmetic coding techniques. Proceedings of the International Conference on Acoustics, Speech and Signal Processing, Volume 2, April 11-14, 1988, New York, pp: 761-764.

Nag, A., S. Biswas, D. Sarkar and P.P. Sarkar, 2010. A novel technique for image steganography based on block-DCT and huffman encoding. Int. J. Comput. Sci. Inform. Technol., 2: 103-112.

Nag, A., S. Biswas, D. Sarkar and P.P. Sarkar, 2011. A novel technique for image steganography based on DWT and Huffman encoding. Int. J. Comput. Sci. Inform. Technol., 4: 561-570.

Nithyanandam, P., T. Ravichandran, N.M. Santron and E. Priyadharshini, 2011. A spatial domain image steganography technique based on matrix embedding and Huffman encoding. Int. J. Comput. Sci. Security, 5: 456-468.

Peng, F., X. Li and B. Yang, 2012. Adaptive reversible data hiding scheme based on integer transform. Signal Process., 92: 54-62.

Praveenkumar, P., R. Amirtharajan, K. Thenmozhi and J.B.B. Rayappan, 2012a. Phase for face saving-a multicarrier stego. Proc. Eng., 30: 790-797.

Praveenkumar, P., R. Amirtharajan, K. Thenmozhi and J.B.B. Rayappan, 2012b. Regulated OFDM-role of ECC and ANN: A review. J. Applied Sci., 12: 301-314.

Praveenkumar, P., M. Nagadinesh, P. Lakshmi, K. Thenmozhi, J.B.B. Rayappan and R. Amirtharajan, 2013a. Convolution and viterbi EN (DE) coders on OFDM hides, rides and conveys message-A neural STEGO. Proceedings of the International Conference on Computer Communication and Informatics, January 4-6, 2013, Coimbatore, pp: 1-5.

Praveenkumar, P., R. Amirtharajan, K. Thenmozhi and J.B.B. Rayappan, 2013b. Can we reduce PAPR? OFDM+PTS+SLM+STEGO: A novel approach. Asian J. Sci. Res., 6: 38-52.

Praveenkumar, P., G.S. Hemalatha, B. Reddy, K. Thenmozhi, J.B.B. Rayappan and R. Amirtharajan, 2014a. Secret link through simulink: A stego on OFDM channel. Inform. Technol. J., 13: 1999-2004.

Praveenkumar, P., K. Thenmozhi, J.B.B. Rayappan and R. Amirtharajan, 2014b. Purposeful error on OFDM: A secret channel. Inform. Technol. J., 13: 1985-1991.

Praveenkumar, P., K. Thenmozhi, J.B.B. Rayappan and R. Amirtharajan, 2014c. Stego in multicarrier: A phase hidden communication. Inform. Technol. J., 13: 2011-2016.

Praveenkumar, P., K. Thenmozhi, J.B.B. Rayappan and R. Amirtharajan, 2014d. Inserted embedding in OFDM channel: A multicarrier stego. Inform. Technol. J., 13: 2017-2021.

Praveenkumar, P., K. Thenmozhi, J.B.B. Rayappan and R. Amirtharajan, 2014e. Data puncturing in OFDM channel: A multicarrier stego. Inform. Technol. J., 13: 2037-2041.

Praveenkumar, P., K. Thenmozhi, J.B.B. Rayappan and R. Amirtharajan, 2014f. Reversible steganography on OFDM channel-a role of RS coding. Inform. Technol. J., 13: 2052-2056.

Praveenkumar, P., K. Thenmozhi, J.B.B. Rayappan and R. Amirtharajan, 2014g. Spread and hide-a stego transceiver. Inform. Technol. J., 13: 2061-2064.

Praveenkumar, P., R. Amirtharajan, K. Thenmozhi and J.B.B. Rayappan, 2014h. Sub carriers carry secret: An absolute stego approach. J. Applied Sci., 14: 1728-1735.

Praveenkumar, P., R. Amirtharajan, K. Thenmozhi and J.B.B. Rayappan, 2014i. Double layer encoded encrypted data on multicarrier channel. J. Applied Sci., 14: 1689-1700.

Praveenkumar, P., R. Amirtharajan, K. Thenmozhi and J.B.B. Rayappan, 2014j. Coded crypted converted hiding (C^3H)-a stego channel. J. Applied Sci., 14: 1786-1797.

Praveenkumar, P., R. Amirtharajan, R.S. Janani, K. Thenmozhi and J.B.B. Rayappan, 2014k. Multi (Carrier+Modulator) adaptive system: An anti fading stego approach. J. Applied Sci., 14: 1836-1843.

Praveenkumar, P., R. Deepak, K. Thenmozhi, J.B.B. Rayappan and R. Amirtharajan, 2014l. Reversible steganography on OFDM channel: A role of cyclic codes. Inform. Technol. J., 13: 2047-2051.

Rajagopalan, S., H.N. Upadhyay, S. Varadarajan, J.B.B. Rayappan and R. Amirtharajan, 2014a. Gyratory assisted info hide-a nibble differencing for message embedding. Inform. Technol. J., 13: 2005-2010.

Rajagopalan, S., K. Pravallika, R. Radha, H.N. Upadhyay, J.B.B. Rayappan and R. Amirtharajan, 2014b. Stego on song-an amalgam of VI and FPGA for hardware info hide. Inform. Technol. J., 13: 1992-1998.

Rajagopalan, S., P.J.S. Prabhakar, M.S. Kumar, N.V.M. Nikhil, H.N. Upadhyay, J.B.B. Rayappan and R. Amirtharajan, 2014c. MSB based embedding with integrity: An adaptive RGB Stego on FPGA platform. Inform. Technol. J., 13: 1945-1952.

Rajagopalan, S., Y. Ravishankar, H.N. Upadhyay, J.B.B. Rayappan and R. Amirtharajan, 2014d. Modeling combo PR generator for Stego Storage Self Test (SSST). Inform. Technol. J., 13: 1936-1944.

Ramalingam, B., R. Amirtharajan and J.B.B. Rayappan, 2014a. LCC-LSB-FPGA stego-A reconfigurable security. J. Applied Sci., 14: 2139-2148.

Ramalingam, B., R. Amirtharajan and J.B.B. Rayappan, 2014b. Stego on FPGA: An IWT approach. Sci. World J. 10.1155/2014/192512

Sakkara, S., D.H. Akkamahadevi, K. Somashekar and K. Raghu, 2012. Integer wavelet based secret data hiding by selecting variable bit length. Int. J. Comput. Applic., 48: 7-11.

Sarita, K.L. and S. Choudhary, 2012. An improved BPCS image steganography in integer wavelet transform domain using 4÷4 block size. Int. J. Eng. Res. Technol., 1: 1-8.

Song, X., S. Wang and X. Niu, 2012. An integer DCT and affine transformation based image steganography method. Proceedings of the 8th International Conference on Intelligent Information Hiding and Multimedia Signal Processing, July 18-20, 2012, Piraeus, pp: 102-105.

Thanikaiselvan, V., P. Arulmozhivarman, R. Amirtharajan and J.B.B. Rayappan, 2011. Wave (let) decide choosy pixel embedding for stego. Proceedings of the International Conference on Computer, Communication and Electrical Technology, March 18-19, 2011, India, pp: 157-162.

Thanikaiselvan, V., P. Arulmozhivarman, J.B.B. Rayappan and R. Amirtharajan, 2012a. Graceful graph for graceful security-towards a STE (G) Raph. Res. J. Inform. Technol., 4: 220-227.

Thanikaiselvan, V., P. Arulmozhivarman, R. Amirtharajan and J.B.B. Rayappan, 2012b. Horse riding and hiding in image for data guarding. Proc. Eng., 30: 36-44.

Thanikaiselvan, V., P. Arulmozhivarman, R. Amirtharajan and J.B.B. Rayappan, 2012c. Wavelet Pave the Trio travel for a secret mission: A stego vision. Global Trends Inform. Syst. Software Applic., 270: 212-221.

Thanikaiselvan, V., P. Arulmozhivarman, S. Subashanthini and R. Amirtharajan, 2013a. A graph theory practice on transformed image: A random image steganography. Sci. World J., Vol. 2013. 10.1155/2013/464107

Thanikaiselvan, V., K. Santosh, D. Manikanta and R. Amirtharajan, 2013b. A new steganography algorithm against chi square attack. Res. J. Inform. Technol., 5: 363-372.

Wong, K., X. Qin and K. Tanaka, 2007. A DCT-based Mod4 steganographic method. Signal Process., 87: 1251-1263.