Performance Analysis of DWT and 4-LSB for Information Hiding in Images (Steganography)

Thazin Aung, Ei Ei Soe Tun
University of Computer Studies, Yangon
julyrain444@gmail.com ; eestun@gmail.com ;

Abstract

Steganographic techniques allow one party to communicate information to another without a third party knowing that the communication is occuring. Steganography works by replacing bits of useless or unused data in regular computer files with bits of different, invisible information. This paper presents the performance analysis for two steganographic algorithms, DWT and 4-LSB algorithm. 4-LSB algorithm works in spatial domain, where dimension can be any size. DWT algorithm works in frequency domain, where there are two dimensions, time and frequency. This system is implemented to hide image and text in the container image. According to experimental results, the similarity of extracted image and secret image by 4-LSB is higher than those of DWT but steganography by DWT method can resist to attacks more than 4-LSB algorithm.

1. Introduction

The development in technology and networking has posed serious threats to obtain secured data communication. One method of providing more security to data is information hiding. Information hiding relates to both Watermarking and Steganography. The Steganography is used for secret data transmission. In steganography, the secret image is embedded in the cover image and transmitted in such a way that the existence of information is undetectable. The digital images, videos, sound files and other computer files can be used as carrier to embed the information. The object in which the secret information is hidden is called cover object. Stego image is referred as an image that is obtained by embedding secret image into cover image. The hidden message may be plain text, cipher text or images etc. Information hiding process in steganographic system starts by identifying a cover medium's redundant bits. The embedding process creates a stego medium by replacing these redundant bits with data from the hidden message.

This paper presents the implementation of Steganography in color image (24-bit image) or gray scale image (8-bit image). In this system, DWT algorithm and 4-LSB algorithm are implemented. Performance analysis for various images is performed to measure the quality of both algorithms. It is defined by three measurements, which are Peak Signal to Noise Ratio (PSNR), Mean Square Error (MSE) and Similarity Measurement (Sim) for similarity calculation.

This paper is organized as follows. Section 1 is the introduction, section 2 is related work. About Image Files is presented in section 3. In section 4, algorithms of steganography are described. Section 5 is the proposed system design and section 6 is the system implementation, detailed process of Image hiding by 4-LSB and DWT and experimental results. Section 7 is the conclusion and future work of the system.

2. Related Work

Steganography can be implemented in various ways. All are based on finding unused space on paper, in sound, or in files in which to hide a message [5, 1]. Many algorithms have been developed to provide robust and secure steganography – each of which uses different embedding techniques. Image hiding methods can be divided into either spatial-domain or frequency-domain. Spatial-domain approaches are discussed in [9, 7, 8]. In these approaches, the hidden information is stored in the least significant bits (LSBs) of the pixels of the cover image. Spatial-domain techniques are intuitive but not robust. The composite image usually cannot be processed using operations such as intensity enhancement, resampling, requantization, image enhancement, cropping, and lossy image compression like JPEG.

Frequency domain techniques [6, 3, 4, 2] take advantage of the human visual system's low sensitivity to high and middle frequency information. A common transform framework is the block-based discrete cosine transform (DCT) [6, 3, 4, 2]. Typically, DCT-based methods divide the cover image into 8 × 8 pixel blocks and apply the DCT transform to each block. Hidden information inserted

into the high frequencies is vulnerable to attack. Conversely, information insertion into the low frequencies may be seen. DCT approaches the middle-frequency information of the cover image is modified according to the embedded image to create the composite image. The embedded image can be extracted by subtracting the discrete cosine transforms of the composite image from the original image. Recently DWT is also used to hide secret message in cover image. It includes adding secret message to the approximation coefficients of low-pass filters of DWT signals. DWT based steganography approach is much more robust to attacks.

3. Steganography

Steganography is known as "covered writing“, hiding information in ways that prevent the detection of hidden messages. A steganographic method is to take the individual pixels in an image. Embedding data, which is to be hidden, into an image requires two files. The first is the innocent-looking image that will hold the hidden information, called the cover image. The second file is the message—the information to be hidden. A message may be plain text, ciphertext, other images, or anything that can be embedded in a bit stream. When combined, the cover image and the embedded message make a stegoimage. Image hiding process can be performed in two domains:

3.1. Spatial Domain

In this domain, image pixels are especially on least significant bits that have less perceptual effect on the images. It is simple and easy to implement, although it is weak at various attacks and noise.

3.2. Frequency Domain

This domain made on the frequency coefficients of images like Discrete Fourier transform (DFT), Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT). It is robust against various attacks and noise and it can get the worst visualization quality of host signals if messages are hidden into the detail coefficient of transform domain

4. Information Hiding Technique

There are two types of main algorithms for information hiding: Least Significant Bit (LSB) insertion: which is processed in spatial domain, and Masking and Filtering: DWT algorithm may be used. It is processed in frequency domain

4.1 Least Significant Bit (LSB) Insertion

Least significant bit (LSB) insertion is a simple approach to embedding information in a cover file. It is vulnerable to even a slight image manipulation. Image conversion from a format like GIF or BMP which reconstructs the original message exactly (lossless compression) to a JPEG which does not (lossy compression) and then back could destroy the information hidden in the LSBs. LSB insertion can be performed in 24-bit, 8-bit or gray-scale images.

4.1.1 4-LSB. Each of these pixels in an image is made up of a string of bits. 4-least significant bits of 8-bit true color image hold 4-bit of the secret message by simply overwriting the data that was already there. The impact of changing the 4-least significant bits is almost always entirely imperceptible. Figure 1 shows the general description of 4-least significant bits, it shows the number 115 and the corresponding binary value, the 4-LSB is highlighted in red. Only these values are allowed to change.

Figure 1: 4-LSB (right 4 bits to store secret message)

In this paper, 4-least significant bits are used to embed secret message in cover image. The message may be a few thousand bits (often at 7 or 8 bits per text character) embedded in millions of other bits. Digital images are commonly stored in either 24-bit or 8-bit files. If an 8-bit image is viewed as a grid and the grid is made up of cells, these cells are called pixels. Each pixel consists of an 8-bit binary number (or a single byte), and each 8-bit binary number refers to the color palette (a set of colors defined within the image). All color variations for the pixels are derived from three primary colors: red, green, and blue. Each primary color is represented by 1 byte (8 bits).

4.2 Discrete Wavelet Transform

A two-dimensional discrete wavelet transform [21, 22] and its inverse are the extension of the one-dimensional discrete wavelet transform. They are implemented using a one-dimensional DWT and IDWT along each of the x and y coordinates. In other words, we apply a low-pass filter and a high-pass filter along each of the two coordinates. The original image is decomposed into four sub-images as follows:

LL: obtained by applying low-pass filters on both coordinates.

HL and LH: obtained by applying the high-pass filter on one coordinate and the low-pass filter on the other coordinate.

HH: obtained by applying the high-pass filter on both coordinates.

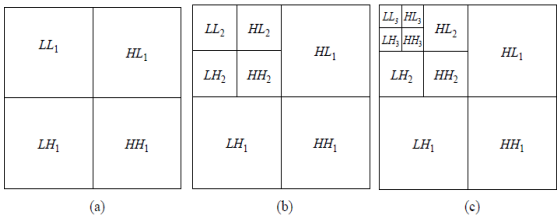

Figure 2: Stages of DWT

Figure 2 presents three stages of DWT used in this paper. In the first stage of the transform, the original image is split into four quarter-size images, the upper left (LL_1), the upper right (HL_1), the lower left (LH_1), and the lower right (HH_1). In the subsequent stages, LL_i, $i \geq 1$, is recursively decomposed into four quarter-size components denoted LL_{i+1}, LH_{i+1}, HL_{i+1}, and HH_{i+1}. Given the wavelet coefficients of an image f(x, y), f(x, y) can be reconstructed using the wavelet coefficients as well as LL_3. LL_3 is the region that contains the lowest frequency information. Because the human vision is sensitive to low spatial frequencies, LL_3 is the most important component in the reconstruction process.

5. Proposed System

This system presents the information hiding process in 4-LSB algorithm and DWT algorithm. Information hiding process consists of the following processes as shown in Figure 3. In 4-LSB, it generates pixel values of the cover image and those pixel values are converted into binary values (1, 0). Secret message is also converted into binary values.

In DWT, DWT values of secret image are embedded into least significant DWT values of cover image. To measure the quality of container image and secret image after image hiding process, performance analysis is done by 3 measurements, which are Peak Signal to Noise Ratio (PSNR), Mean Square Error (MSE) and Similarity Measurement (Sim) for similarity calculation.

The objective of this system is to measure the quality of image hiding algorithm and their reliability in ways of computational measurements. Figure 3 presents the proposed system design. It accepts two inputs (cover image and secret image / message) and applied into 4-LSB algorithm and DWT algorithm. Both algorithms produce stegno images and performance analysis is performed on both algorithms.

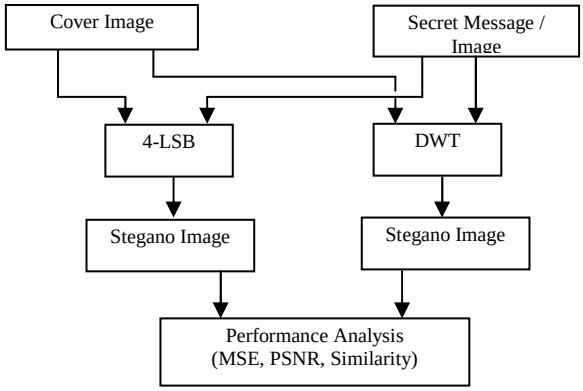

Figure 3: System overview

6. System Implementation

This system is implemented using Matlab programming language. The implementation consists of Image hiding by DWT and 4-LSB algorithm. Both image and message can be hidden in this system.

6.1. Implementation of 4-LSB

In this system, 4-least significant bits are used to embed secret message in cover image. Process of implementing 4-LSB algorithm is shown in Figure 4, where cover image and secret image (message) are converted into binary values. Then secret message is constructed by replacing 4-least significant bits of cover image with 4 bits of secret message.

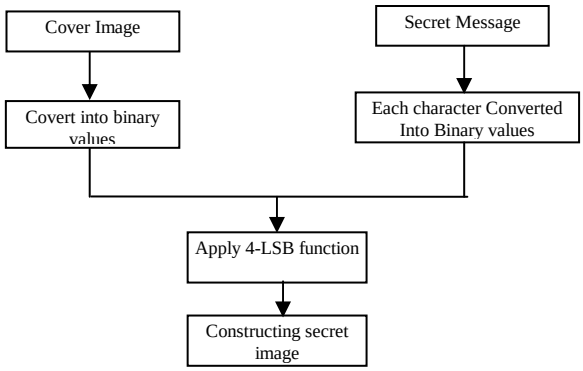

Figure 4: Implementation of 4-LSB

Following algorithm presents the implementation of 4-LSB encoding.

Input: ContainerImage CI, SecretImage SI
Output: SteganoImage STI

Processing

Step1: Decomposing into byte values
decompose CI into byte values.

decompose SI into byte values

Step2: Replacing bits

Extract higher 4 bits of SI byte values
Remove lower 4 bits of CI byte values
Replace lower 4 bits of CI byte values with higher 4 bits of SI values

Step3: Reconstructing the embedded image into stegano-image

The implementation for 4-LSB decoding is presented as shown below:

Input: SteganoImage STI
Output: SecretImage SI

Processing

Step1: Decompose STI into byte values.
Step2: Extract lower 4 bits of STI byte values
Step3: Reconstruct the extracted lower bits into Secret Image SI

6.2. Implementation of DWT

Two-dimensional discrete wavelet transform with DWT type "haar" is applied into information hiding area. Low-pass filter and high-pass filter along each of the two coordinates are used. Image hiding is based on the following observations.

1. An image embedded in the high frequencies is vulnerable to attack. However, images embedded in the low frequencies may be visible.
2. LL1, LL2, and LL3 in a three-stage wavelet transform are the most important components for reconstructing an embedded image. As long as these three components are available, we can reconstruct an image that is perceptibly similar to the original embedded image.

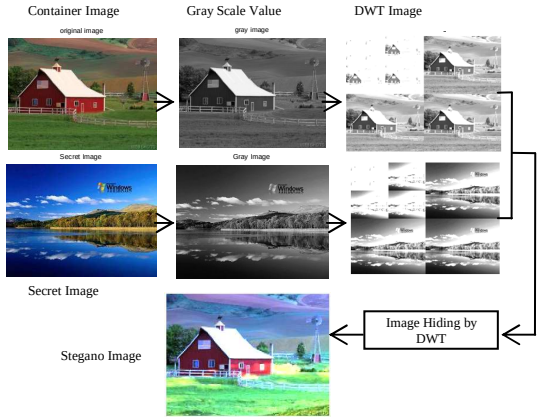

Figure 5: Image Hiding by DWT

Figure 5 presents image hiding process by DWT. 2D- DWT is applied to both cover image and secret image. DWT of secret image is embedded in low pass filter of DWT of cover image. The implementation of image hiding by DWT is shown in following algorithm.

Input: CoverImage CI,
SecretImage SI

Output: SteganoImage

Processing

Step1: Apply DWT to decompose the cover image CI into CI' using four non-overlapping sub-bands:

approximation band cA1 and
details bands cH1, cV1, and cD1 by (also know as LL1, HL1, LH1 and HH1)
two dimensional wavelet decomposition with haar

Step2: Apply DWT To decompose cA1 into cA1', cA2, cH2, cV1 and cD2 (LL2, HL2, LH2, HH2)
Step3: Apply DWT To decompose cA2 into cA2', cA3, cH3, cV3 and cD3 (LL3, HL3, LH3, HH3)
Step4: The same process as in Step 1 is performed into Secret Image
Step5: Embed High level wavelet of secret image into low level wavelet of container image
Step6: Compute Inverse DWT into Embedded DWT wavelets using reconstructing filter (Stegano Image)

The implementation of information un-hiding by DWT is shown as follows:

Input: SteganoImage SI
Output: Secret Image

Processing

Step1: Apply DWT to decompose the cover image SI into SI' using four non-overlapping sub-bands:

approximation band cA1 and
details bands cH1, cV1, and cD1 by (also know as LL1, HL1, LH1 and HH1)
two dimensional wavelet decomposition with haar

Step2: Apply DWT To decompose cA1 into cA1', cA2, cH2, cV1 and cD2 (LL2, HL2, LH2, HH2)

6.3. Experimental Results

This system is tested with different images with different size. According to experimental results, stegano image of DWT algorithm can resist to attacks more than 4-LSB algorithm. Performance analysis is performed based on following measurements.

Mean Square Error (MSE): MSE represents the cumulative squared error between the reconstructed and the original image. The lower the value of MSE,

the lower the error. Mean-squared error can be computed using the following equation:

$$MSE = \frac{\sum_{M,N} [I_1(m,n) - I_2(m,n)]^2}{M * N}$$

(Eqn. 1)

where, M and N are the number of rows and columns in the input images, respectively.
I1 = Original Image
I2 = Transformed Image

Peak Signal to Noise Ratio (PSNR): The PSNR computes peak signal-to-noise ratio, between two images. This ratio is often used as a quality measurement between the original and an output image. The higher the PSNR, the better the quality of the reconstructed image. To compute the PSNR, the block first calculates the mean-squared error. PSNR can be computed using the following equation:

$$PSNR = 20 \log_{10} \left[\frac{1}{MSE} \right]$$

(Eqn. 2)

Similarity Measurement (Sim): The similarity between two digital images could be quantified by correlation function. Each image is normalized by its root power. So the Similarity measurement is defined as:

$$p(w, w`) \frac{\sum_{i=1}^{N} w_i \times w`_i}{\sqrt{\sum_{i=1}^{N} w_i^2} \sqrt{\sum_{i=1}^{N} w`_i^2}}$$

(Eqn. 3)

Where,
w = Original Image
w` = Transformed Image
n = number of pixels

The higher values of similarity measurement imply more similarity between the original image and transformed image.

The experimental results for DWT and 4-LSB are shown in following tables. Table 1 and 2 show maximum secret image and message size, where N represents the dimension of cover image. According to the tables, 4-LSB algorithm can store more secret image / size than DWT algorithm.

Table 1: Maximum Secret Image Size

No.	Algorithm	Container Size	Secret Size
1	4-LSB	N	N – 3
2	DWT Algorithm	N	N / 16

Table 2: Maximum Secret Message Size

No.	Algorithm	Container Size	Secret Size
1	4-LSB	N	(N / 2) – 4
2	DWT Algorithm	N	N / 16

Table 3 and 4 present the experimental results of 4-LSB and DWT algorithms in their performance measures. According to experimental results, 4-LSB produces more similarity values than DWT algorithm. But in the processing time, DWT algorithm is faster than 4-LSB algorithm.

Table 3: Experimental Process for Hiding Image (4-LSB)

No.	Container	Secret	PSNR	MSE	Sim	Time (s)
1	1680 x 1050	113 x 134	12.4	0.239	0.999	60.39
2	1280 x 960	113 x 134	26.11	0.049	0.999	48.79

Table 4: Experimental Process for Hiding Image (DWT)

No.	Container	Secret	PSNR	MSE	Sim	Time (s)
1	1680 x 1050	113 x 134	6.8	0.43	0.981	5.69
2	1280 x 960	113 x 134	3.11	0.549	0.999	3.84

4-LSB algorithm can only resist ‘Salt & Pepper’ noise type. It is vulnerable ‘Gaussian’ and ‘Poisson’ attacks. Image Hiding with DWT, can resist to all noise types, ‘Gaussian’, ‘Poisson’ and ‘Salt & Pepper’. Experimental results are presented in Table 5 and Table 6. Table 5 shows performance measures for 4-LSB with ‘Salt & Pepper’ noise type. Header values in Table 5 are noise values added to the stegano image. Table 6 describes performance measures for DWT with different noise types. Therefore DWT is more robust to different attacks.

Table 5: PSNR and MSE for different noise values (4-LSB)

	0.05	0.01	0.001
PSNR	10.5674	24.3984	42.4734
MSE	0.29623	0.060267	0.007522
Sim	98.28 %	99.3%	99.5%

Table 6: PSNR and MSE for different noise types (DWT)

	Gaussian	Poisson	Salt & Pepper
PSNR	4.4879	1.6697	1.677
MSE	0.3317	0.277	0.277
Sim	99.57%	99.98%	99.99%

7. Conclusion

This system presents the performance analysis of DWT and 4-LSB for Information Hiding in Images (Steganography). Images used in this system can be any type of image and can be any size, but secret image must be less than or equal to the container image. Text messages can also be hidden in the container image. Three measurements (MSE, PSNR and Similarity) are used to measure the quality of DWT and 4-LSB. Time measurement is also used in this system. According to the experimental results, DWT is faster than 4-LSB algorithm and DWT is more robust to different attacks.

8. References

[1] Cox, I. J., Kilian, J., Leighton, T. and Shammoon, T. "Secure spread spectrum watermarking for multimedia," Technical Report 95-10, NEC Res. Inst., Princeton, NJ, 1995.

[2] Hsieh, M. S., Tseng, D. C. and Huang, Y. H., "Hiding digital watermarks using multiresolution wavelet transform," IEEE Transactions on Industrial Electronics, Vol. 48, 2001, pp. 875-882.

[3] Hsu, C. T. and Wu, J. L., "Hidden digital watermarks in images," IEEE Transactions on Image Processing, Vol. 8, 1999, pp. 58-68.

[4] Kii, H., Onishi, J. and Ozawa, S., "The digital watermarking method by using both patchwork and DCT," in Proceedings of the IEEE Multimedia Computing and System, Vol. 1, 1999, pp. 895-899.

[5] Koch, E. and Zhao, J. "Toward robust and hidden image copyright labeling," in Proceedings of the IEEE Nonlinear Signal and Image Processing, 1995, pp. 452-455.

[6] Piva, A., Barni, M., Bartolini, F. and Cappellini, V., "DCT-based watermark recovering without resorting to the uncorrupted original image", in Proceedings of the IEEE International Conference on Image Processing, Vol. 1, 1997, pp. 520-523.

[7] Turner, L. F., "Digital data security system," Patent IPN WO 89/08915, 1989.

[8] Walton, S. "Image authentication for a slippery new age," Dr. Dobb's Journal, Vol. 20, 1995, pp. 18-26.

[9] Wu, D. C. and Tsai, W. H., "Image hiding in spatial domain using an image differencing approach," in Proceedings of Conference on Computer Vision, Graphics, and Image Processing, 1998, pp. 280-287.