ANZIAM J. 48(2006), 23–35

WAVELET AND DISCRETE COSINE TRANSFORMS FOR INSERTING INFORMATION INTO BMP IMAGES

I. OREA-FLORES[1], M. A. ACEVEDO[2] and J. LÓPEZ-BONILLA[✉2]

(Received 1 July, 2004; revised 26 April, 2006)

Abstract

In this work we use the Discrete Wavelet Transform in watermarking applications for digital BMP images with the objective is to guarantee some level of security for the copyright. We also compare the results with the Discrete Cosine Transform for the same application. Results are obtained from a number of tests, primarily in order to validate the security level and the robustness of the watermark, but also to prove that the original image suffers only very small variations after the watermark is embedded. We also show how to embed the watermark, where to insert it and the capacity supported for inserting an image.

2000 *Mathematics subject classification*: primary 42C40, 44A15; secondary 62P25, 94A60.
Keywords and phrases: watermarking, steganography, wavelet transform, discrete cosine transform.

1. Introduction

Watermarking arises from the need to protect copyrights in digital documents such as text, audio, images and video [9]. Due to espionage and the high volume of illegal copies of different types of media, it is of great importance to hide information (steganography) or to authenticate it. Since most communication channels are inherently insecure, we must find a way to interchange information in a secure manner even if using these channels [3]. One alternative way of achieving this goal is to transmit information that appears to be "normal" at a first glance while containing hidden information. We concentrate on BMP files for hiding information since they are a widely used image format.

[1]Unidad Profesional Interdisciplinaria en Ingeniería y Tecnologas Avanzadas, Instituto Politécnico Nacional, Av. IPN 2580, La Laguna Ticomán, 07340 México D.F.; e-mail: iorea@ipn.mx.
[2]Sexyón de Estudios de Posgrado e Investigación Escuela Superior de Ingeniería Mecánica y Eléctrica Instituto Politécnico Nacional Edif. Z-4, 3er piso. Col Lindavista, 07738 México D.F.; e-mail: macevedo@ipn.mx, jlopezb@ipn.mx.

In this work we present some results using the Discrete Cosine Transform (DCT) and wavelets for steganography and watermarking.

In the first part of this article we present a brief description of the DCT and wavelet techniques in one and two dimensions (vectors and matrices, respectively) used to insert information in the frequency domain. After this transformation, information is inserted in the middle and high frequency range of the BMP image.

The inverse procedure must be applied in order to recover the original BMP file. We then obtain the correlation index of the original and modified images using these two techniques in order to gain insight into how much the resulting image has been modified. Finally, we make a comparison between the DCT and wavelets to implement the steganography and watermarking on BMP files. Performance is measured through the correlation index as well as the information inserting capacity.

2. Discrete cosine transform

At the moment, the techniques that are employed the most in watermarking are those that introduce a domain transformation rather than space domain techniques. This is because by applying a frequency transformation it is possible to observe certain characteristics of the image that simplifies the manipulation of the information. This allows us to have a higher degree of security and robustness in the watermarks [4].

The DCT maps the values of the pixels of the image, one by one from the time domain to the frequency domain. Due to the arithmetic form of the DCT, this process is reversible [5, 10].

We assume a one-dimensional image consisting of a linear series of N pixels. Each pixel corresponds to a gray scale $p(x)$ $(0 \leq x < N)$ where $p(x)$ is a function that varies in space. This image can then be represented by the sum of the components of this space f with a frequency ranging from 0 to $N-1$. The grayscale function is given by

$$p(x) = \sqrt{\frac{2}{N}} \sum_{f=0}^{N-1} C(f)S(f) \cos\left[\frac{(2x+1)\pi f}{2N}\right]$$
$$= \frac{S(0)}{\sqrt{N}} + \sqrt{\frac{2}{N}} \sum_{f=1}^{N-1} S(f) \cos\left[\frac{(2x+1)\pi f}{2N}\right], \tag{2.1}$$

where

$$C(f) = \begin{cases} 1/\sqrt{2} & f = 0, \\ 1 & f > 0. \end{cases}$$

To calculate (2.1) we first need to find the coefficients $S(f)$:

$$\{S(f),\ 0 \leq f < N\}.$$

https://doi.org/10.1017/S1446181100003394 Published online by Cambridge University Press

The first term in (2.1) corresponds to the constant component or the zero frequency component. This can be calculated as the average value of $p(x)$, given by

$$S(0) = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} p(x).$$

The general expression for $S(f)$ is

$$S(f) = \sqrt{\frac{2}{N}} C(f) \sum_{x=0}^{N-1} p(x) \cos\left[\frac{(2x+1)\pi f}{2N}\right]. \tag{2.2}$$

Equation (2.2) is the one-dimensional DCT of $p(x)$, and (2.1) is the inverse DCT of $S(f)$ [3, 4, 9].

3. The two-dimensional DCT

An $N \times N$ pixel matrix can be represented by the sum of $N \times N$ cosine functions in the form of

$$p(x, y) = \frac{2}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} C(u)C(v)S(u, v) \cos\left[\frac{(2x+1)\pi u}{2N}\right] \cos\left[\frac{(2x+1)\pi v}{2N}\right], \tag{3.1}$$

where $S(0, 0) = (1/N) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} p(x, y)$. The general equation for $S(f)$ is

$$S(u, v) = \frac{2}{N} C(u)C(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} p(x, y) \cos\left[\frac{(2x+1)\pi u}{2N}\right] \cos\left[\frac{(2x+1)\pi v}{2N}\right], \tag{3.2}$$

Equation (3.2) is the two-dimensional DCT of $p(x, y)$.

Frequency values are ordered diagonally as shown in Figure 1. Hence, the lowest frequency is placed in position (1, 1) while position (8, 8) holds the highest frequency value [7].

4. Steganography using the DCT

Modern steganography systems are very robust since they use some form of transformation from one domain to another.

Transformation methods from one domain to another hide the message in special areas of the image to be transmitted, making the system more robust to specialised attacks, like compression, allowing us to insert a watermark [10].

https://doi.org/10.1017/S1446181100003394 Published online by Cambridge University Press

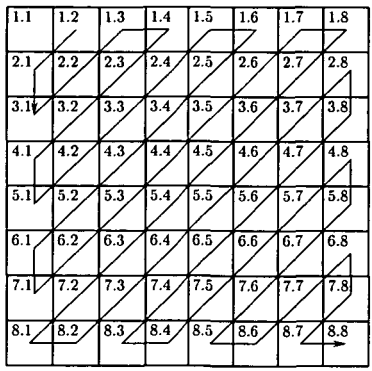

FIGURE 1. Frequencies are placed in a diagonal form from the lowest to the highest value.

The process of insertion is carried out in the frequency domain, therefore we have to transform the image from the space domain to the frequency domain. However, we cannot transform the image in an RGB format directly, we must first convert it to the luminance and crominance equivalent using the next set of equations [4]:

$$Y = 0.99R + 0.587G + 0.114B, \quad Cb = 0.5 + \frac{B - Y}{2}, \quad Cr = 0.5 + \frac{R - Y}{2}.$$

Our tests show that watermarks are more efficient, in terms of information recovery, when applied in the YCbCr space rather than the RGB space. Hence, it is necessary to move from the RGB space to the YCbCr space before the frequency domain transformation takes place.

Once the image is in the YCbCr format we can transform it to the frequency domain.

During the coding procedure, the image is divided into 8×8 blocks of pixels; exactly one bit of the hidden message is coded in each block. The process of insertion starts by selecting the block b_i in a pseudo-random manner. This block is used to code the i-th bit of the hidden message. We then transform the image to the frequency domain using the DCT, resulting in the image blocks $B_i = D\{b_i\}$.

Then we localise two coefficients in the block that will be used to insert the message. Each coefficient is denoted by two indices (u_1, v_1) and (u_2, v_2). Both coefficients represent a component in the medium and high frequency range. This guarantees that the hidden information will be saved in a significant part of the image. This also assures that the insertion process will not degrade the image significantly since middle range frequency coefficients have very similar values. The coefficients used could be (4, 1) and (3, 2), for example. Now, to insert a hidden message in the image we just have to compare the values of the coefficients in the high or middle frequency range.

https://doi.org/10.1017/S1446181100003394 Published online by Cambridge University Press

One frequency block codes a "1" if $B_i(u_1, v_1) > B_i(u_2, v_2)$, otherwise, it codes a "0". When compression is used, the coefficient values may be altered, therefore it is recommended that $|B_i(u_1, v_1) > B_1(u_2, v_2)| > x$ for every x bigger than zero; this could be achieved by adding a random number to both coefficients. The bigger the value of x that is chosen, the more robust the system is when compression is used, but the image could be more affected. We then perform the inverse DCT to have the image coefficients in the space domain [4, 5, 10].

5. Wavelet transforms

Wavelet analysis is a recent mathematical development that can be used for a variety of applications in the areas of mathematics, physics and engineering [2]. Wavelets are functions with the general form:

$$\psi_{j,k}(x) = a^{j/2}\psi(a^j x - kb),$$

where $\psi \in L^2(R)$, $a > 0$, $b \in R$, and j, k are integers.

That is, wavelets are functions that are generated by a single function ψ (the generating wavelet). The standard values of a and b are $a = 2$ and $b = 1$.

The wavelet coefficients of $f \in L^2(R)$ are defined as

$$\tilde{f}(j, k) = \int_{-\infty}^{\infty} f(x)\overline{\psi}_{j,k}(x)\, dx$$

and the series $\sum_{j,k=-\infty}^{\infty} \tilde{f}(j, k)\psi_{j,k}(x)$ is called the associated wavelet series with f.

6. The discrete wavelet transform (DWT)

The signal $x[n]$ is processed through a mirror bank filter in quadrature [6]. The resulting signal of each filter is decimated by a factor of 2.

The resolution of the signal, which is a measure of the amount of detailed information in the signal, is modified by the filter and the scale is modified by the decimation. Decimation corresponds to the decrease of sampling frequency or to the elimination of certain samples of the signal. The process of the filtering and successive decimation is known as sub-band codification. This procedure can be expressed as:

$$y_h[k] = \sum_n x[n] \cdot g[2k - n] \tag{6.1}$$

and

$$y_l[k] = \sum_n x[n] \cdot g[2k - n],$$

https://doi.org/10.1017/S1446181100003394 Published online by Cambridge University Press

where $y_h[k]$ and $y_l[k]$ are the outputs of the high pass and low pass filters, respectively, after the decimation of 2.

This is the operation of the DWT, which analyses the signal in different frequency bands with different resolutions through the decomposition of the signal in components of high and low energy. The decomposition of the signal in different frequency bands is obtained through the successive filtering of the signal. The original sequence $x[n]$ is processed by a high pass filter $g[n]$ and a low pass filter $h[n]$.

The sub-band codification can be applied as many times as needed in order to achieve a higher degree of decomposition. Each filtering and decimation operation results in half the number of samples and therefore half the resolution in time. Depending on the chosen wavelet, the coefficients of $g[n]$ and $h[n]$ will change.

There are several wavelet transforms, but in this paper we focus on the Haar transform since our tests showed that under attack, this is the transform that conserved the watermark almost unchanged (this is due to the distribution of the energy). It is thus the most appropriate for our application and we only apply one level of transformation [1, 12, 14].

The Haar wavelet transform breaks the discrete signal $x = (x_1, x_2, \ldots, x_N)$ in two sub-signals of half the length of the original. The first sub-signal $a_1 = (a_1, \ldots, a_{N/2})$ is called the average of signal x and is calculated as follows. The first value a_1 is the average of the first couple of values of x: $(x_1 + x_2)/2$. This is then multiplied by $\sqrt{2}$, thus $a_1 = (x_1 + x_2)/\sqrt{2}$. Similarly, the next value is calculated using the next couple of values of x as $a_2 = (x_3 + x_4)\sqrt{2}$. All values of a_m are obtained in this manner, by averaging pairs of values of x and then multiplying by $\sqrt{2}$. The general formula to obtain a_m is:

$$a_m = \frac{x_{2m-1} + x_{2m}}{\sqrt{2}} \quad \text{for } m = 1, 2, \ldots, \frac{N}{2}.$$

The other sub-signal is called the difference of signal x and is denoted by $d_1 = (d_1, d_2, \ldots, d_{N/2})$. It is obtained as follows. The first value of d_1 corresponds to half the difference between the first couple of values of x: $(x_1 - x_2)/2$ and it is then multiplied by $\sqrt{2}$, resulting in $d_1 = (x_1 - x_2)/\sqrt{2}$. The rest of the values of d_m are obtained in a similar way using

$$d_m = \frac{x_{2m-1} - x_{2m}}{\sqrt{2}} \quad \text{for } m = 1, 2, \ldots, \frac{N}{2}.$$

This procedure accommodates the low frequencies in a_1 while the high frequencies are placed in d_1.

The wavelet transform can be done in various levels. In this paper we will only focus on the first level [12, 14].

https://doi.org/10.1017/S1446181100003394 Published online by Cambridge University Press

7. The discrete wavelet transform in two dimensions (TWD2)

A discrete image x is an $M \times N$ matrix of real numbers as shown in (6.1).

The wavelet transformation in two dimensions is obtained in the same manner as in the previous section for one dimension as follows:

$$x = \begin{pmatrix} x_{1,M} & x_{2,M} & \cdots & x_{N,M} \\ \vdots & \vdots & \ddots & \vdots \\ x_{1,2} & x_{2,2} & \cdots & x_{N,2} \\ x_{1,1} & x_{2,1} & \cdots & x_{N,1} \end{pmatrix}.$$

(A) Apply the wavelet transform to each row of x to generate a new matrix.

(B) Apply the wavelet transform to the new matrix generated in the previous step but now to each column.

This will create four sub-images of $M/2$ rows and $N/2$ columns each:

$$f \longrightarrow \left(\begin{array}{c|c} h^1 & d^1 \\ \hline a^1 & v^1 \end{array}\right). \tag{7.1}$$

The sub-image a^1 is calculated by averaging over the rows and then averaging over the columns. The sub-image created is then a compression of the original with the low frequency components of the image.

The sub-image h^1 is calculated as the average of the rows and the difference of the columns. This saves the horizontal details of the image and contains the medium-low frequency components.

The sub-image v^1 is similar to h^1 except that it holds the vertical details of the image and it contains the medium- high frequency components.

Finally, d^1 contains the diagonal details since it is obtained as the difference of both the rows and the columns and it contains the high frequency components [4, 12, 14].

Figure 2 shows the visual decomposition in frequencies of an image when the Haar wavelet transform of one level is applied.

8. Steganography using the discrete wavelet transform

We now have the matrices a^1, h^1, v^1 and d^1; the matrix a^1 is maintained without change since medium frequency components are contained here while a hidden message can be embedded in the rest of the matrices. The insertion of the message is accomplished in this manner: we compare the first couple of values of each matrix, if the first value is higher than the second, we consider it to code a "1", otherwise it

https://doi.org/10.1017/S1446181100003394 Published online by Cambridge University Press

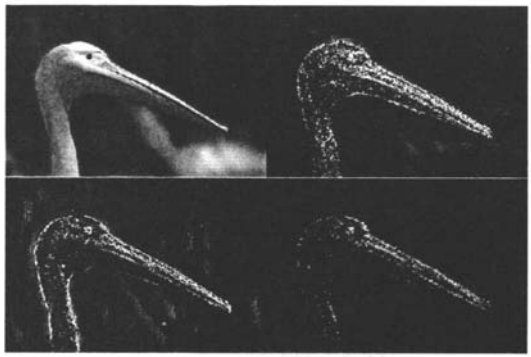

FIGURE 2. One level wavelet transform.

1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8
2.1	2.2	2.3	2.4	2.5	2.6	2.7	2.8
3.1	3.2	3.3	3.4	3.5	3.6	3.7	3.8
4.1	4.2	4.3	4.4	4.5	4.6	4.7	4.8
5.1	5.2	5.3	5.4	5.5	5.6	5.7	5.8
6.1	6.2	6.3	6.4	6.5	6.6	6.7	6.8
7.1	7.2	7.3	7.4	7.5	7.6	7.7	7.8
8.1	8.2	8.3	8.4	8.5	8.6	8.7	8.8

FIGURE 3. Frequencies conserved intact.

is coded a "0", we then compare the next pair of values and continue this procedure until the whole matrix has been compared.

9. Attacks

Once the watermark has been inserted in the digital document, it is vulnerable to a wide range of intentional attacks [11]. There are many attacks with the objective of eliminating a watermark. It is important to notice that in order for an attack to be effective, it has to eliminate the watermark without visibly altering the image [13]. We only consider three of the most direct attacks: compression, distortion of brightness and noise.

(A) Compression: When a compression algorithm is applied to the image and then the image is obtained in its original format.

https://doi.org/10.1017/S1446181100003394 Published online by Cambridge University Press

TABLE 1. Insertion capacity with DCT and DWT.

Original Image	Insertion Technique	Insertion Capacity	Lost Energy	PSNR (dB)
Image 1	DCT	5. 39%	0. 8389%	60.9217
	DWT	51. 22%	0. 1230%	66.1421
Image 2	DCT	5. 53%	1. 1219%	60.8776
	DWT	50. 98%	0. 0986%	66.1956
Image 3	DCT	5. 20%	0. 9336%	61.2812
	DWT	51. 01%	0. 1432%	66.4136
Image 4	DCT	5. 66%	0. 8396%	60.8547
	DWT	51. 04%	0. 1132%	67.3012
Image 5	DCT	5. 59%	0. 8187%	61.2634
	DWT	51. 77%	0. 1741%	67.1324
Image 6	DCT	5. 66%	1. 1043%	61.5995
	DWT	50. 98%	0. 1082%	67.5538
Image 7	DCT	5. 50%	1. 1461%	61.6772
	DWT	51. 09%	0. 0965%	66.8119
Image 8	DCT	5. 51%	1. 1401%	61.3451
	DWT	50. 93%	0. 0856%	67.2012
Image 9	DCT	5. 75%	1. 0568%	60.7267
	DWT	51. 03%	0. 0930%	66.3230
Image 10	DCT	5. 68%	1. 0394%	60.9094
	DWT	51. 02%	0. 1580%	66.7715

(B) Distortion of Brightness: This type of attack involves adding a small value to each pixel of the image in order to modify it completely with no visible impact. The sum of the value is performed uniformly over all of the image [8].

(C) Noise: There are many different types of noise that can be added to the image. We focus on three in particular, namely multiplicative noise, impulsive noise and Gaussian noise. It is important to mention that all the tests were done in MATLAB where all numerical parameters are normalised; in particular, all parameters involving image operations are normalised in the range [0, 1].

10. Implementation and tests

We now show some results obtained from different tests with both techniques, such as insertion capacity and recovery percentage of the watermark when it is subject to different attacks.

(A) Insertion Capacity. For both techniques, low frequency values are unchanged. Hence, using the DCT technique, the frequency values that are not used are marked in the shaded region in Figure 3, and the rest are used to insert the information. This

https://doi.org/10.1017/S1446181100003394 Published online by Cambridge University Press

procedure is performed in each plane of the image, that is, in the chrominance and luminance planes for both the DCT and DWT techniques.

Using the same criteria, we apply the DWT directly in each plane of the image of $N \times N$, resulting in four sub-matrices of $M/2 \times N/2$ grouped in frequencies. Information is inserted in the middle-low, middle-high and high frequencies.

Table 1 shows a comparison of both techniques in the insertion process for 10 different images. It also shows the energy loss and the PSNR (dB).

(B) Compression. This is the most direct attack because it is almost impossible to visibly notice any changes in the image after performing a compression. Hence this is a very aggressive kind of attack.

By introducing the watermark only in the luminance matrix and in the middle-low frequencies we achieve a robust watermark when compression is applied to the image. Under this condition, both techniques can recover the original watermark completely after the compression procedure. However, the insertion capacity is drastically reduced.

Table 2 shows the capacity insertion of robust watermarks that can support a compression procedure.

TABLE 2. Insertion capacity after a compression.

Original Image	Insertion Technique	Insertion Capacity	Original Image	Insertion Technique	Insertion Capacity
Image 1	DCT	0.57%	Image 6	DCT	0.55%
	DWT	4.29%		DWT	4.27 %
Image 2	DCT	0.54%	Image 7	DCT	0.53%
	DWT	4.26%		DWT	4.3%
Image 3	DCT	0.56%	Image 8	DCT	0.53%
	DWT	4.26%		DWT	4.22%
Image 4	DCT	0.53%	Image 9	DCT	0.55%
	DWT	4.27%		DWT	4.27%
Image 5	DCT	0.54%	Image 10	DCT	0.55%
	DWT	4.28%		DWT	4.26%

(C) Brightness Distortion in the Image. Brightness distortion makes the image brighter by adding the same value to all the pixels of the image or darkens the image by subtracting the same value. If these values are not too big, then the changes in the image will not be visually detected. However simple, this is a very effective attack against watermarks.

Brightness distortion does not eliminate the watermark because of the way that it is inserted. As we know, both techniques compare two elements, a "1" is inserted if one value is greater than the other and a "0" is inserted if one values is lower that the other value. By adding a value to all the elements of the image, this relationship is not

https://doi.org/10.1017/S1446181100003394 Published online by Cambridge University Press

modified.

(D) Noise. Noise is also a form of an effective attack to either modify or eliminate the watermark, so long as the level of noise is not too high. The level of noise cannot be too high because then the image will suffer a big distortion and will become visibly different from the original image, making it obvious that an attack has occurred.

TABLE 3. Multiplicative noise inserted in image.

Multiplicative Noise	Insertion Technique	PSNR (dB)	Watermark recuperate
Variance = 0.002	DCT	67.5236	100%
	DWT	67.5412	100%
Variance = 0.006	DCT	65.1269	91%
	DWT	65.8561	97%
Variance = 0.008	DCT	60.7826	86%
	DWT	61.0124	87%
Variance = 0.012	DCT	58.9445	77%
	DWT	60.2098	82%
Variance = 0.015	DCT	57.8923	71%
	DWT	59.2895	75%

TABLE 4. Impulsive noise inserted in image.

Impulsive Noise	Insertion Technique	PSNR (dB)	Watermark recuperate
Density = 0.0015	DCT	62.2917	83%
	DWT	63.7054	85%
Density = 0.003	DCT	61.1455	81%
	DWT	61.3548	84%
Density = 0.005	DCT	59.0944	79%
	DWT	59.1510	81%
Density = 0.008	DCT	57.6612	70%
	DWT	57.9785	73%
Density = 0.01	DCT	55.4344	64%
	DWT	56.8501	69%

Finally, a watermark is inserted in 100 different images and the PSNR of the original image is measured and compared to the PSNR of the image with the watermark. This is done for both the DCT and DWT techniques. The results are shown in Figure 4.

11. Conclusions

The DWT technique for the insertion of digital watermarks is efficient because embedded information in the image can be recovered. It is secure since the embedded

https://doi.org/10.1017/S1446181100003394 Published online by Cambridge University Press

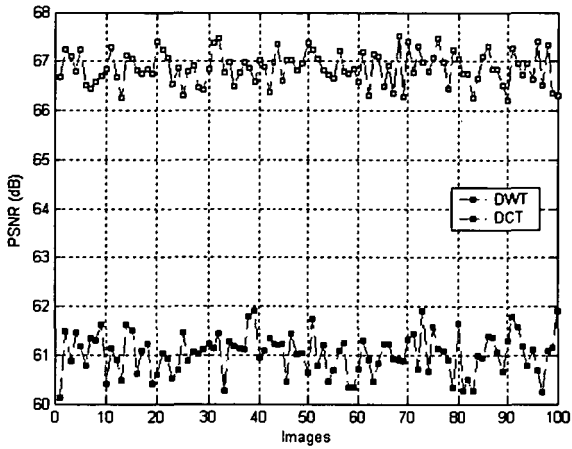

FIGURE 4. PSNR of the DWT and the DCT.

information is not visible to any non authorised person and it is practically impossible to detect. Most importantly, watermarks are robust because we have shown that they can resist compression by commercial algorithms.

Watermarks also resist a brightness distortion, that is, attacks that alter the image uniformly, and this type of attack never modifies the watermark. In the case of multiplicative noise, noise with a variance greater than 0.002 does not affect the embedded information. However, impulsive noise—and Gaussian noise also—affect the watermark but they also visibly affect the image and therefore, the attack is evident.

TABLE 5. Gaussian noise inserted in image.

Gaussian Noise	Insertion Technique	PSNR (dB)	Watermark recuperate
Variance = 0.002	DCT	62.1489	81%
	DWT	62.3488	82%
Variance = 0.005	DCT	59.8430	78%
	DWT	60.1055	79%
Variance = 0.006	DCT	58.4931	75%
	DWT	59.6156	77%
Variance = 0.008	DCT	57.8096	70%
	DWT	58.4861	72%
Variance = 0.01	DCT	56.0483	64%
	DWT	56.7056	67%

In all the previous tests, the DCT and DWT techniques showed very similar results. However, it is important to notice that the DWT technique always achieves a better performance. For example, when the watermark is embedded, the image is less

https://doi.org/10.1017/S1446181100003394 Published online by Cambridge University Press

affected. Also, in all the tests with noise, the DWT technique always achieves a higher percentage of recovery. Finally, we compare the capacity for inserting information and we can see that the wavelet technique is much better than the DCT technique.

After several tests with different images we obtained the following results. For the DCT we could hide approximately 0.54% of the information compared to the size of the original image file for watermarking. For steganography the percentage of information for hidden information is 4.82%.

For the wavelet method, we could insert approximately 4.27% of information compared to the original size of the image file for watermarking. For steganography, the percentage is approximately 51.3% of embedded information.

References

[1] M. A. Acevedo, I. Orea-Flores and J. López-Bonilla, "Wavelets and discrete cosine transform for hidden information into images", *Sampling Theory in Signal and Image Processing (STSIP Journal)* **4** (2005) 141–150.

[2] J. J. Benedetto and A. I. Zayed, *Sampling, Wavelets, and Tomography* (Birkhäuser, Boston, 2003) 11–17.

[3] A. Kaarna and P. Toivanen, "Digital watermarking of spectral images in PCA/wavelet-transform domain", *IEEE Trans. Image Process.* **6** (2003) 220–224.

[4] S. Katzenbeisser and F. A. P. Petitcolas, *Information Hiding*, Principles of Steganography (Artech House, London, 2000).

[5] D. Knuth, *The art of computer programming, Vol. 2, Seminumerical Algorithms*, 2nd ed. (Addison–Wesley, Reading, Mass., 1981).

[6] S. K. Mitra (ed.), *Digital signal processing: a computer based approach*, 2nd ed. (McGraw-Hill, New York, 2003) 88–94.

[7] I. Orea, "Intercambio de información utilizando protocolos de canal subliminal", Tesis UPIITA-IPN, (Mexico DF, 2002).

[8] F. A. P. Petitcolas, R. J. Anderson and M. Kuhn, *Attacks on copyright marking systems* (Artech House, London, 1998).

[9] J. J. K. Ruanaidh, W. J. Dowling and F. M. Boland, "Watermarking digital images for copyright protection", *IEE Proc.-Vis. Image Signal Process.* **153** (1996) 250–256.

[10] B. Schneirer, *Applied cryptography* (John Wiley and Sons, New York, 1994).

[11] J. K. Su, J. J. Eggers and B. Girod, "Capacity of digital watermarks subjected to an optimal collusion attack", in *X European Signal Processing Conference (EUSIPCO 2000), Tampere, Finlandia, Sept. 2000*, http://www.lnt.de/~eggers/publications.html

[12] M. Vetterli and J. Kovacevic, *Wavelets and subband coding* (Prentice Hall, Englewood Cliffs, NJ, 1995).

[13] S. Voloshynovskiy, S. Pereira, T. Pun, J. J. Eggers and J. K. Su, "Attacks on digital watermarks: Classification, estimation-based attacks and benchmarks", *IEEE Comm. Mag.* **39** (2001) 118–127.

[14] J. S. Walter, *A primer on wavelets and their scientific applications* (Chapman & Hall/CRC, London 1999).