Journal of Education for Pure Science- University of Thi-Qar
Vol.11, No1 (June, 2021)
Website: jceps.utq.edu.iq Email: jceps@eps.utq.edu.iq

DOI: http://doi.org/10.32792/utq.jceps.11.01.12

An Image Watermarking Technique Proposed Based on Discrete Cosine Transformation and Pseudo-Random Generator

Khadhim Mahdi Hashim Salwa Shakir Baawi Bushra Kamil Hilal

[1,3] Department of Computer Science College of Education for Pure Science University of Thi-Qar, Thi-Qar, Iraq.
[2]Department of Computer Information Systems College of Computer Science & IT University of Al-Qadisiyah, Diwaniyah, Iraq

Received 21/04/2021 Accepted 30/05/2021 Published 10/06/2021

Abstract:
The proposed technique to embed an image watermark in a host image depends on the discrete cosine transform and the Geffe algorithm in this search. The proposed algorithm uses the Geffe algorithm for encryption of the watermark. Discrete cosine transform (DCT) coefficients are then computed by dividing the host image into non-overlapped 8x8 blocks. The performance evaluation of the suggested technique was conducted using the peak signal-to-noise ratio for watermark 64x64 and 512x512 host images: Lena, Cameraman, and Pepper images give similar results. This technique does not require a host image in the watermark extraction process, then also watermark is extracted completely and better than its counterparts available.

Keywords: Steganography, Watermark, Image watermark, Geffe, Security.

1. Introduction:

Life is becoming more convenient as the internet and multimedia technologies develop, as well as the spread of digital multimedia content. Text, images, audio, and video, as well as other forms of multimedia, are becoming more widely accessible and used, and the scope of their spread also becomes wider [1]. One issue with a digital image is that it is simple to share and/or forge an infinite number of copies of the "original." If the image is copyrighted, this may cause issues. For the purpose of identifying possession, people devised a method of embedding the business name, a unique digital identifier, and other material inside multimedia archives. This method is named "Digital watermarking", which is a branch of information hiding technologies [2].

A digital watermark, in general, is a technique for adding secret copyright material or other authentication messages to digital media [3]. Any multimedia format, such as images, audios, or videos, can be used as the host media. Image watermarking has sparked a lot of interest in the scientific community in the area of watermarking for two reasons. First, because it is readily available, and second, because it contains sufficient redundant information to enable watermarks to be embedded [4]. A watermarking technique can be separated into two parts in general. The first is known as the embedding process, while the second is known as the extraction process. Watermarks are introduced into the host media during the embedding process, and the existence of the watermark must be checked during the extraction process [5].

Watermarking is the method of embedding a watermark, also known as a digital signature, tag, or logo, into a multimedia entity such that the watermark can be identified or removed later to render an assertion about the object [3].

Digital watermarking strategies can be divided into two categories based on the watermark embedding environment: spatial domain and transform domain. Spatial techniques immediately embed the watermark on the pixel intensity without transforming the pixel intensity of the original image [6], Based on the literature, many researchers worked on spatial domain techniques that have various characteristics. For instance, the technique of least substitution bit is one bit of secret message replaces a least significant bit of every pixel to hide information in a sequence of the binary number, another technique is the correlation-based Technique is the watermarking technique that exploits the correlation properties of additive pseudo-random noise patterns as applied to the image. In this technique, less information can be hidden and the Image quality may decrease [7]. Transform domain techniques are computationally difficult and robust for signal processing attacks. The transform domain techniques are good adoptions for the imperceptible and robust watermarking method [7]. The important frequency domain-based techniques, For instance, Discrete Wavelet Transform (DWT) is the filter-based system in which the image is decomposed into a set of four non-overlapping multi-resolution sub-bands. Where the quality of the image is degraded if the watermark is embedded in low-frequency information and holds high computational complexity. Discrete Cosine Transform (DCT) works on the watermark information that can be easily embedded into the lower or middle-frequency bands, lossy compression, and watermark destroys with cropping and scaling. Whilst, Discrete Fourier Transform (DFT) has a poor energy compaction property [8].

This paper focuses on the creation and application of DCT-based image digital watermarks. Improve a DCT transform and Geffe algorithm-based digital image watermarking algorithm. Geffe scrambles the initial watermark information to ensure the security of the watermarking. The experimental results show that it hides the gray-scale image's detail, and a simulation of some typical image attacks is performed.

The following is a description of the composition of the paper. The second section presents related work, the third section discusses discrete cosine transformation, the fourth section describes hidden key watermarking, the fifth section tests the proposed algorithm, and section six concludes with an overview of the experimental findings. Finally, section 7 provides a conclusion.

---

2. Related Work:

Transform domain watermarking is used to improve robustness, in which the image is converted into frequency coefficients using one of the transformation methods, such as the discrete cosine transform DCT. Centered on block-based DCT coefficient alteration, Parah et al. (2016) suggested. The sum of change to a block's DCT coefficient is determined by the scaling vector, DC coefficient, and median of some zigzag ordered AC coefficients in that block, according to the findings. The suggested scheme's robustness has been tested against a variety of singular and hybrid threats. Furthermore, comparative findings showed that the suggested strategy is more resistant to attacks [10].

Loan et al. proposed a secure and blind watermarking scheme in the DCT domain in their paper (2018). The messy encryption-based digital image watermarking technique can be used on both gray-scale and color photographs. Until embedding the watermark in the host image, the discrete cosine transform (DCT) is used. Prior to DCT implementation, the host image is divided into 8 x 8 non-overlapping blocks, and the watermark bit is embedded by changing the discrepancy between neighboring blocks' DCT coefficients [11].

A self-recovery fragile image watermarking with variable watermark capability was proposed by Wang et al. in (2018). This work improves the security of the watermarking algorithm by encrypting the watermark before embedding it with binary pseudo-random sequences. The results of the experiments showed that the proposed approach produced strong manipluate detection results, and the recovered image had improved image quality than other self-recovery fragile watermarking methods [12].

3. Discreet cosine transformation:

The discrete cosine transformation (DCT) is considered to have high energy concentration properties in the field of discrete cosine transformation (DCT) research. As a consequence, it's been applied to a variety of tasks, including image compression, watermarking, and pattern recognition [13].The signal is processed using discrete cosine transformation (DCT). The signal is converted from the spatial to the frequency domain. The DCT watermark is more durable than watermarking techniques used in space. The DCT divides an image into various frequency bands, making it much simpler to insert watermarking information in the image's frequency bands [2].

The most popular DCT description of N-dimensional 1-D sequence is [14]:

$$C(u) = a(u) \sum_{X=0}^{N-1} f(x) \cos\left[\frac{\pi(2x+1)u}{2N}\right], u = 0,1,2, \dots N-1 \quad \cdots (3.1)$$

The inverse transformation, on the other hand, is known as:

$$f(x) = \sum_{u=0}^{N-1} a(u)\, c(u) cos\left[\frac{\pi(2x+1)u}{2N}\right] \quad \dots (3.2)$$

The following are the definitions of equations (3.1) and (3.2):

$$a(u) = \begin{cases} \sqrt{\frac{1}{N}} & for\ u = 0 \\ \sqrt{\frac{2}{N}} & for\ u \neq 0 \end{cases} \quad \dots (3.3)$$

N is the pixel block's horizontal and vertical pixel size, and N=8 is the pixel block's horizontal and vertical pixel number.

This is generally the case. For u=0, Equation (1) explicitly indicates that:

$$c(u=0) = \sqrt{\frac{1}{N}} \sum_{x=0}^{N-1} f(x)$$

As a result, the first transform coefficient is equal to the average value of the sample sequence's sample. This value is referred to as the DC coefficient in the literature. The AC coefficients are the names given to all other transform coefficients.

The two-dimensional discrete cosine transformation is a direct extension of the one-dimensional case and is defined by:

$$C(u,v) = a(u)a(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x,y) cos\left[\frac{\pi(2x+1)u}{2N}\right] cos\left[\frac{\pi(2y+1)v}{2N}\right] \dots \quad (3.4)$$

where u,v=0,1,...N-1, and the following is the description of the inverse transform:

$$f(x,y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} a(u)a(v)\, c(u,v) cos\left[\frac{\pi(2x+1)u}{2N}\right] cos\left[\frac{\pi(2y+1)v}{2N}\right] \dots \quad (3.5)$$

Where: x,y= 0,1,…, N-1.

---

4. Secret Key Watermarking:

This section discusses how to acquire a secure key for watermarking. A stream cipher is a type of encryption algorithm that is widely used. Specific characters (usually binary digits) of a letter are encrypted one at a time, utilizing an encryption transformation that changes with time. It's possible to use a symmetric-key or public-key stream cipher. Asynchronous stream ciphers and synchronous stream ciphers are the two types of stream ciphers [14].

Linear Feedback Shift Registers are the most common building block for keystream generation in stream ciphers (LFSR). LFSRs have a number of advantages, including a stable hardware implementation, strong statistical properties, a long range, high linear complexity, and ease of study using algebraic techniques. The secret key in these ciphers is LFBSR's initial state [15]. The general structure of a LFBSR is depicted in Figure 1.

Many of the main stream generators that have been proposed in the literature use LFBSRs. This is due to a number of factors [16]:

a. Hardware implementation is well-suited to LFBSRs.
b. They have the ability to generate large time sequences.
c. They are capable of producing statistically sound sequences; and
d. They are easily analyzed using algebraic techniques due to their structure.

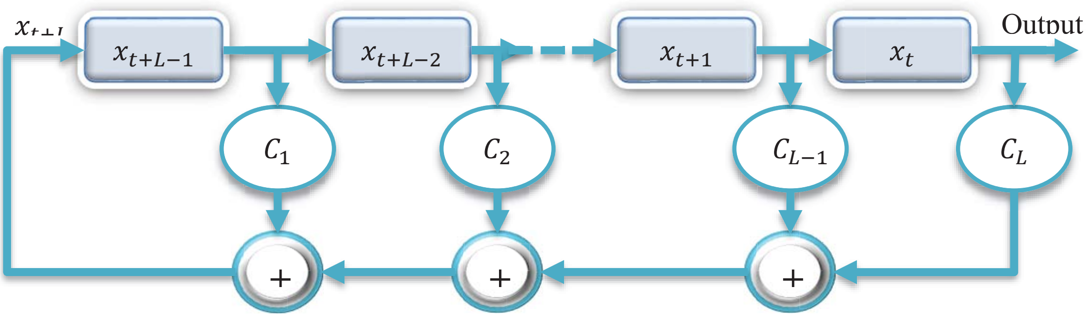

Figure 1. General structure of a LFBSR

A binary linear feedback shift register (LFBR) with an internal state of n bits is a finite state automaton of size n. The update function L shifts the state by one position per clock cycle, with the input bit being a linear function of the previous bits. Let $x = (x_0, \dots, x_{n-1})$ be the initial state more precisely. Recursion $x_t = (c_1 x_{t-1} \oplus \dots \oplus c_n x_{t-n})$ determines the output series, for $t \geq n$ and all c_i values are set to 0 and 1.

The use of a non-linear hybrid of linear feedback shift registers preserves the best aspects of LFSRs (statistical properties, quick generation, basic design) while excluding their cryptographic vulnerability. This method produces excellent results, and many stream ciphers are based on it [17]. Using many LFBSRs in parallel is one general strategy for destroying the linearity inherent in LFBSRs. The keystream is created as a nonlinear function f of the variable LFBSRs' outputs (e.g. Geffe generator) [14], as seen in Figure 2.

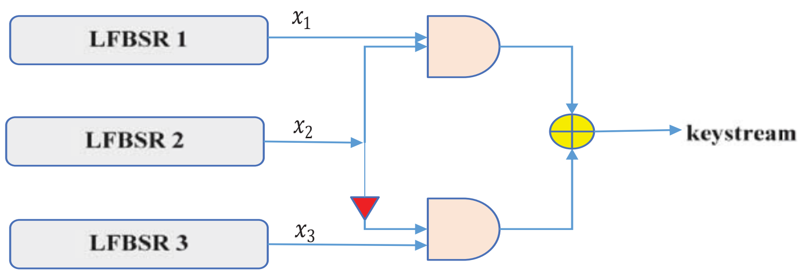

Figure 2. The generator of Geffe

Figure 2 shows Geffe's demonstration of a generator made up of three LFBSRs. Three LFBSRs are joined together to form Geffe's generator. LFBSR2 will be used as a control generator to connect either LFBSR1 or LFBSR3 to the output, but not both [17]. The Geffe generator consists of three maximum-length LFBSRs with nonlinear mixing functions, with lengths L_1, L_2, and L_3 that are pairwise relatively prime.

$$f(x_1, x_2, x_3) = x_1x_2 \oplus (1 + x_2)x_3 = x_1x_2 \oplus x_2x_3 \oplus x_3 \ldots \quad (6)$$

The output sequence has a maximal priod $(2^{L_1} - 1) \cdot (2^{L_2} - 1) \cdot (2^{L_3} - 1)$ and a linear complexity $L = L_1L_2 + L_2L_3 + L_3$.

5. The Proposed Method:

This proposed method performs in the frequency domain, embedding a pseudo-random sequence in a chosen series of DCT coefficients in this watermarking technique to apply an image to digital image file. To ensure watermark invisibility, watermark casting is done by utilizing the masking characteristics of the Human Visual System (HVS). Without using the original, the embedded sequence is removed.

5.1 Embedding process:

This section explains the fundamentals of an image watermarking algorithm, which consists of two steps: watermark embedding and extraction. The proposed watermark embedding algorithm requires two images: host (F) and watermark (W). F of size 512x512 pixels and the watermark image can be represented by different dimensions of (32x32, 64x64, 128x128). The required steps for the embedding procedure are mentioned below.

Algorithm: Embedding Watermark:

Input: Host image, watermark image, initial value of Geffe generator registers

Output: Watermarked image

Steps:

1. Read the host imag and watermark image
2. Make the size of host image 512x512
3. Let the watermark image size mxm pixels, where (m=32, 64, 128)
4. Convert the watermark image into binary image, then to vector.
5. Perform a nonlinear encrypting (using Geffe generator algorithm) in one process on the watermark image vector to increase safety and prevent non-responsible individuals from viewing the watermark image.
6. Divide the host image into 8x8 blocks, calculate the DCT transform for each block, and prepare a stream from the DCT coefficients. The number of blocks is the same as the number of pixels in the watermark image.

7. To insert every pixel of the encrypted watermark image into the corresponding block from the host image, use a mathematical relationship. As a result, the block watermarked image is obtained.

   Block watermarked = block host (1+a.g) … (5.1)

   Where g is a small value for the embedding factor (Seen more details in the Section of results and discussion), which is used to decide whether the watermark is visible or not (small value to get invisible watermark), and a is the value calculated using the color of the actual watermark pixel in equation 5.2.

$$a = \begin{cases} 1 & pixel = white \\ -1 & pixel = black \end{cases} \quad \dots(5.2)$$

8. After the watermark has been embedded, the IDCT transform is computed for each block in order to convert the current block values form frequency domain to spatial domain. After that, the watermarked image is reconstructed.

Figure 3 Embedding watermark algorithm

Figure 4 depicts the various components of the watermark embedding framework.

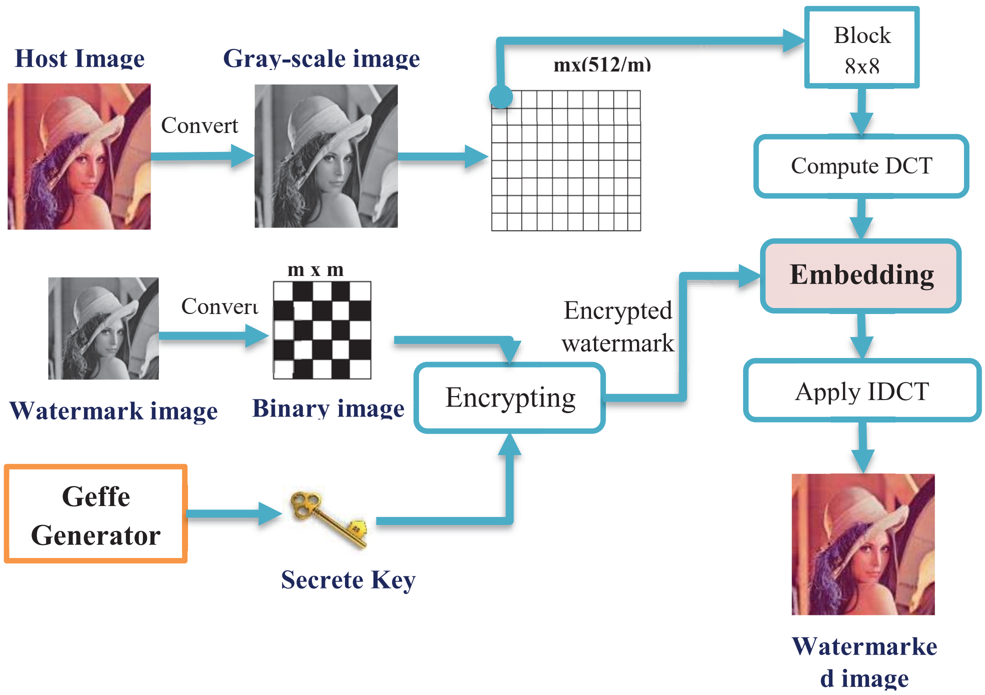

Figure 4. Watermarking embedding framework

In this study, tests were conducted on color images such as Lena, Cameraman, and Pepper, as well as a gray-scale watermark image.

The initial values of the Geffe generator are kept in the sender and receiver locations.

5.2 Extracting process:

The steps of algorithm for extracting a watermark from a watermarked image are inverse to those steps that mentioned in the previous section and have been briefly clarified here:

Algorithm: Extracting watermark image

Input: watermarked image, initial value of Geffe generator registers

Output: Reconstructed watermark image

Steps:

1. Read the watermarked image and divide it into eight 8x8 pixel non-overlapping image blocks.

2. The DCT approximation coefficients are extracted from the watermarked image using the IDCT algorithm.
3. Using an inverse mathematical relationship to the mathematical relationship used in the method of embedding the watermark on each block, calculate the corresponding pixels.
4. Recovered each pixel from the specific locations of watermarked image blocks.
5. The key watermark bits are generated when the Geffe generator is run.
6. The secret key bits are used to decode the encrypted watermark.
7. To obtain a watermark image that is visually identical to the original watermark image, decode the result of the previous phase.

Figure 5 Extracting watermark algorithm

Figure 6 offers the various components of watermarking extracting framework.

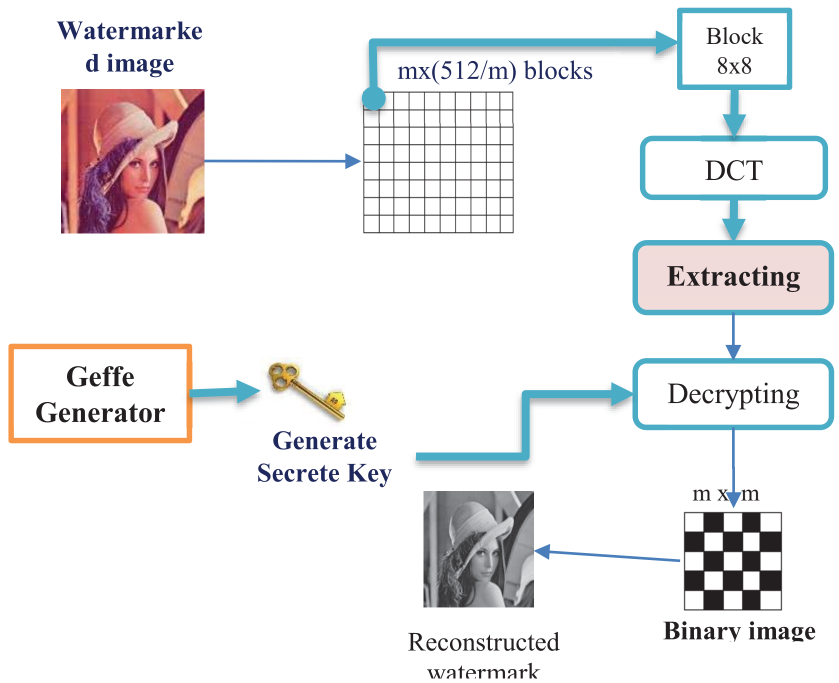

Figure 6. Watermarking extracting framework

6. Experimental Results Analysis and Discussion:

On an Intel Core i7 processor with 8 GB RAM, the suggested method for digital image watermarking is implemented in MATLAB (R2018b). The standard RGB or gray input images are of various sizes. The suggested method's PNSR, MSE, and NCC are used to assess the watermarked image.

According to (Khalifa and Saeed, 2020) in [13], MSE stands for the square of the variations in pixel values of the two images' respective pixels. Equation 6.1 gives the MSE of a NxM size image.

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[F(i,j) - K(i,j)]^2 \qquad \dots (6.1)$$

Whilst Equation 6.2, the ratio between the highest potential power of a signal and the power of vitiation noise that affects accuracy is known as PSNR.

$$PSNR = 10.LOG_{10}\left(\frac{Max_I^2}{MSE}\right) \qquad \dots (6.2)$$

Besides, the security of a watermark is a top priority for researchers. Unauthorized users would be unable to recognize, retrieve, or alter the embedded watermark [18]. According to the equation below, Normalized Cross Correlation (NC) has a value of [0,1] [7].

$$NCC = \frac{Mij.Rij}{{M_{(i,j)}}^2} \quad \ldots (6.3)$$

Where Mij is the orginal message embed, Rij is recovered image. The embedded and recovered watermarks are identical when NC=1. A fair watermark extraction value is widely agreed to be NC>0.7500. Where W' stands for Watermark Detection and W stands for original watermark. The watermark is measured in decibels, and the higher the PSNR frequency, the more embedded the watermark is [19]. The below are the findings of the experiments: -

Applying the proposed approach to three standard 512x512 color images: Lena, Cameraman, and Pepper. Lena, Navy, and Wolf are used as a watermark that is an mxm pixel binary image. Moreover, the embedding factor uses in different values to obtain on the best quality of watermarked image.

Table 1. Performance metrics of a watermarking algorithm without attack

Host image	Watermark image	Embeding factor	MSE	PSNR	NCC
Lena 512x512	Lena 64x64	0.1	7.174	8.5576	1
		0.01	0.0675	11.7006	1
		0.001	0.000675	31.7006	1
		0.0001	6.7599e-06	51.7006	1
Cameraman 512x512	Navy 64x64	0.1	7.5755	8.7941	1
		0.01	0.07575	11.2059	1
		0.001	0.0007575	31.2059	1
		0.0001	7.5755e-06	51.2059	1
Pepper 512x512	Wolf 64x64	0.1	19.0861	-12.8072	1
		0.01	0.1908	7.1928	1
		0.001	0.001908	27. 1928	1
		0.0001	1. 1908e-05	47. 1928	1

Table 1 shows the results of the MSE according to Equation (6.1) and PSNR according to Equation (6.2) analyses. high PSNR values are a secret for successful watermarking. Low MSE values, on the other hand, indicate an imperceptible watermark. The PSNR and MSE findings for the Navy image are the best, while those for the wolf image are the worst.

These results prove the proposed algorithm is robust and secure due to the NCC metric achieves a high value is equal to 1. And also, , we were able to show that the proposed algorithm is both robust and secure in our experiments. Furthermore, in this section, the embedding factor tests in a variety of values in order to determine the best embedding factor for the best values of PSNR and MSE. Figure 5 shows all of the host images and the watermark are used, as well as the recovered watermark that indicated in (Rd, Re, and Rf).

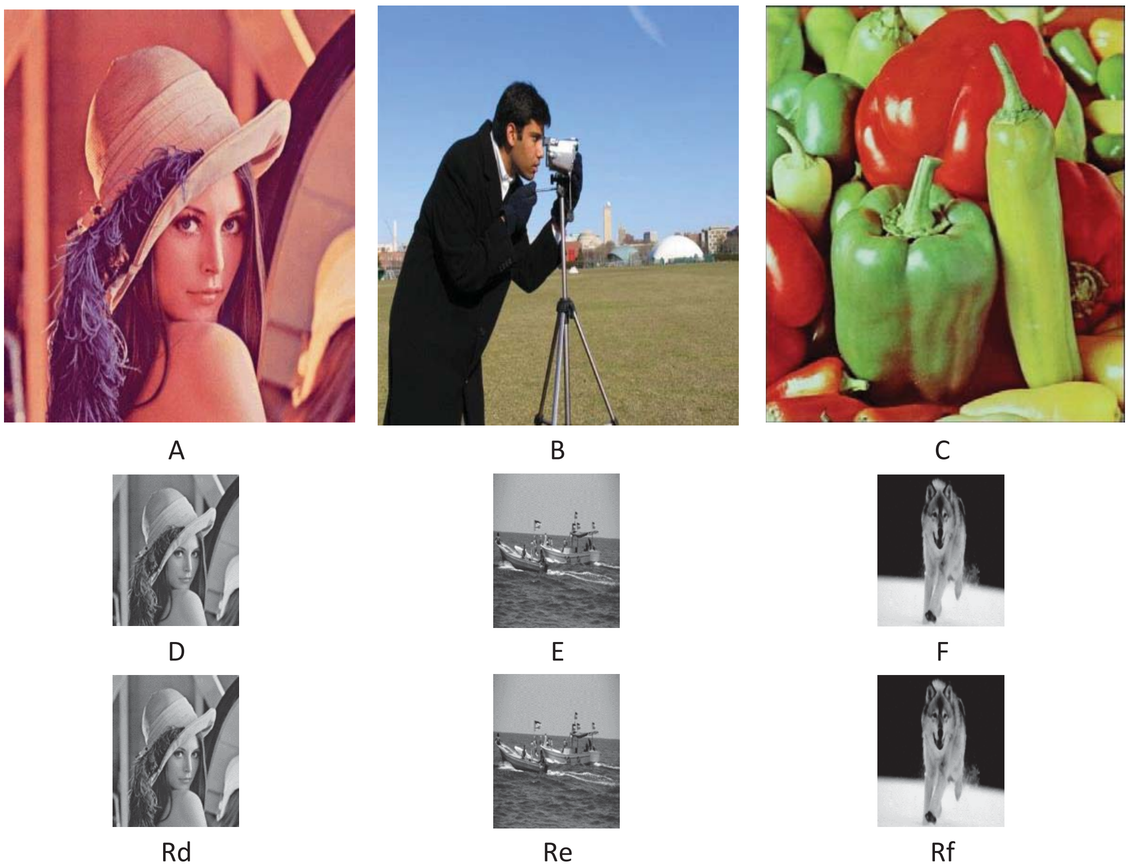

Figure 7 Original host images and watermark. (a) Lena, (b) Cameraman, (c) Pepper, and (d) Lena, (e) Navy, (f) Wolf watermark, as well as, recovered watermark (Rd) recovered Lena, (Re) recovered Navy, and (Rf) recovered Wolf .

By evaluating various values for the embedding factor (g in Equation(5.1)) such as (0.1, 0.01, 0.001, and 0.0001), Figure 8 presents the watermarked image that uses (0.1) for each images. In Figure 8, we noticed distortion in each watermarked image, particularly in the upper left corner, as indicated by the arrows.

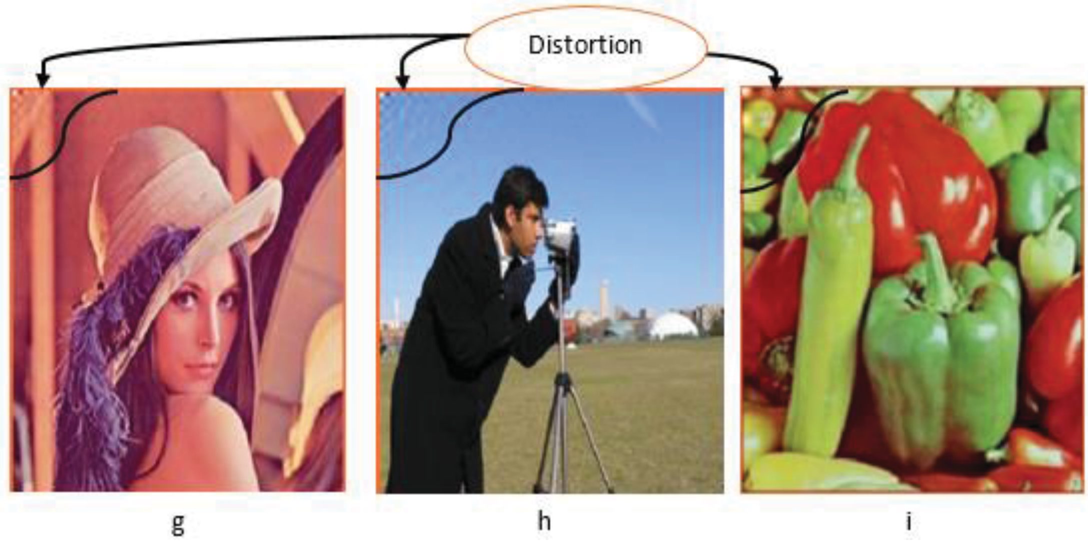

Figure 8 The watermarked image (g) Lena, (h) Cameraman, and (i) Pepper

We also noticed that the distortion decreases whenever smaller values of the embedding factor are taken, the output watermarked image using (0.0001), as shown in Figure 9.

Figure 9 The watermarked image (j) Lena, (k) Cameraman, and (l) Pepper

According to Figure 9, this study discovered that the embedding factor (0.0001) is the best value that can be added in the embedding equation to obtain on the best quality watermarked image and the reciever can recovered the watermark correctly. Therefore, we discard the output that used embedding factor values (0.01, 0.001) because the outputs have distortion and the values of PSNR is very small. In addition to the benefits of this strategy, there are several drawbacks to it, which can be summarized as follows: in each site of the sender and receiver, the embedding factor, secret key, host, and watermark image size should be known.

---

7. Conclusion:

This study suggested a DCT and Geffe algorithm-based digital image watermarking technique. The number of host image blocks was calculated using the number of watermark image pixels in this process. The watermark is inserted in the frequency domain after the DCT coefficients of each block of the host image have been prepared. The experimental findings demonstrated that the current scheme is safe and robust. In addition to the benefits of this strategy, there are several drawbacks to it, which can be summarized as follows: The embedding factor, secret key, and watermark image size must all be the same between the sender and recipient. Improving the suggested approach in the future to achieve a trade-off between security, robustness, and capacity.

Refrences:

[1] J. Han, X. Zhao, and C. Qiu, "A digital image watermarking method based on host image analysis and genetic algorithm," *J. Ambient Intell. Humaniz. Comput.*, vol. 7, no. 1, pp. 37–45, 2016, doi: 10.1007/s12652-015-0298-3.

[2] W. He-Jing, "A DCT Domain Image Watermarking Method Based on Matlab," *Int. J. Adv. Network, Monit. Control.*, vol. 2, no. 2, pp. 38–45, 2017, doi: 10.21307/ijanmc-2017-008.

[3] B. Ram, "Digital Image Watermarking Technique Using Discrete Wavelet Transform And Discrete Cosine Transform," *Int. J. Adv. Res. Technol.*, vol. 2, no. 4, pp. 19–27, 2013.

[4] V. M. Potdar, S. Han, and E. Chang, "A Survey of Digital Image Watermarking Techniques," *2005 3rd IEEE Int. Conf. Ind. Informatics*, pp. 709–716, 2005.

[5] S. P. Singh and G. Bhatnagar, "A new robust watermarking system in integer DCT domain," *J. Vis. Commun. Image Represent.*, vol. 53, no. February 2017, pp. 86–101, 2018, doi: 10.1016/j.jvcir.2018.03.006.

[6] K. PRABHA and I. S. SAM, "Robust color image watermarking by elliptical phase modification based on Walsh Hadamard Transform and Triangular Vertex Transform," *Sādhanā*, vol. 46, no. 1, 2021, doi: 10.1007/s12046-020-01553-5.

[7] D. R. I. M. Setiadi, E. H. Rachmawanto, and C. A. Sari, "Secure Image Steganography Algorithm Based on DCT with OTP Encryption," *J. Appl. Intell. Syst.*, vol. 2, no. 1, pp. 1–11, 2017, doi: 10.33633/jais.v2i1.1330.

[8] M. Begum and M. S. Uddin, "Digital image watermarking techniques: A Review," *Inf.*, vol. 11, no. 2, pp. 1–38, 2020, doi: 10.3390/info11020110.

[9] A. Fatahbeygi and F. Akhlaghian Tab, "A highly robust and secure image watermarking based on classification and visual cryptography," *J. Inf. Secur. Appl.*, vol. 45, no. January, pp. 71–78, 2019, doi: 10.1016/j.jisa.2019.01.005.

[10] S. A. Parah, J. A. Sheikh, N. A. Loan, and G. M. Bhat, "Robust and blind watermarking technique in DCT domain using inter-block coefficient differencing," *Digit. Signal Process. A Rev. J.*, vol. 53, pp. 11–24, 2016, doi: 10.1016/j.dsp.2016.02.005.

[11] N. A. Loan, N. N. Hurrah, S. A. Parah, J. W. Lee, J. A. Sheikh, and G. M. Bhat, "Secure and Robust Digital Image Watermarking Using Coefficient Differencing and Chaotic Encryption," *IEEE Access*, vol. 6, pp. 19876–19897, 2018, doi: 10.1109/ACCESS.2018.2808172.

[12] C. Wang, H. Zhang, and X. Zhou, "A self-recovery fragile image watermarking with variable watermark capacity," *Appl. Sci.*, vol. 8, no. 4, 2018, doi: 10.3390/app8040548.

[13] F. M. Khalifa and M. G. Saeed, "Image Watermarking Using All Phase Discrete Cosine Biorthogonal Transform in Selected Pixel Blocks," *Polytech. J.*, vol. 10, no. 1, pp. 68–73, 2020, doi: 10.25156/ptj.v10n1y2020.pp68-73.

[14] S. Sleibi Al-Rawi, A. Tariq Sadiq, and B. Ghbeash Farhan, "Digital Video Quality Metric Based on Watermarking Technique with Geffe Generator," *Comput. Sci. Eng.*, vol. 2, no. 7, pp. 138–146, 2013, doi: 10.5923/j.computer.20120207.03.

[15] K. Marton, A. Suciu, and I. Ignat, "Randomness in digital cryptography: A survey," *Rom. J. Inf. Sci. Technol.*, vol. 13, no. 3, pp. 219–240, 2010.

[16] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of Applied Cryptography*. 1996.

[17] A. Klein, *Stream ciphers*, vol. 9781447150. 2013.

[18] T. Reports, "Digital Text Watermarking Techniques Classification and Open Research Challenges : A Review," no. June, 2020.

[19] Y. Perwej, F. Parwej, and A. Perwej, "An Adaptive Watermarking Technique for the copyright of digital images and Digital Image Protection," *Int. J. Multimed. Its Appl.*, vol. 4, no. 2, pp. 21–38, 2012, doi: 10.5121/ijma.2012.4202.
K. PRABHA and I. S. SAM, "Robust color image watermarking by elliptical phase modification based on Walsh Hadamard Transform and Triangular Vertex Transform," *Sādhanā*, vol. 46, no. 1, 2021, doi: 10.1007/s12046-020-01553-5.

[20] C. Kumar, A. K. Singh, and P. Kumar, "A recent survey on image watermarking techniques and its application in e-governance," Multimed. Tools Appl., vol. 77, no. 3, pp. 3597–3622, 2018, doi: 10.1007/s11042-017-5222-8.

[21] M. Begum and M. S. Uddin, "Digital image watermarking techniques: A Review," Inf., vol. 11, no. 2, pp. 1–38, 2020, doi: 10.3390/info11020110.