An Effectively Utilized Histogram Modification Based Watermarking Scheme

Nader H. H. Aldeeb and Ibrahim S. I. Abuhaiba

Computer Engineering Department, Islamic University of Gaza, Gaza, Palestine
nader_deeb@yahoo.com, isiabuhaiba@gmail.com

Abstract

In this paper, we propose a Copyright Protection (CP) watermarking scheme, which is a robust, blind, and reversible watermarking scheme; aimed at protecting the copyright of color images against geometrical attacks. It is a development of an already existing watermarking scheme. Watermark embedding in this scheme, as well as in its counterpart, mainly depends on the permutation of the histogram bins. However, we present a new embedding rule, which increased the average capacity by about 55 bits, and it increased the quality, PSNR, of the watermarked image from 35.01 dB to 38.05 dB. The embedded watermark demonstrates 100 % robustness against a variety of geometrical attacks, like Flipping (H, V, and Both), Rotation (90°, 180°, and 270°), Scattering, Warping, Skewing, and their combinations. Finally, our proposed CP watermarking scheme showed a faster watermark embedding process than that of its counterpart by an average reduction in time equals, 4.84 seconds.

Keywords: *Color Image Watermarking, Copyright Protection, Geometrical Attacks, Robust, Reversible, Blind*

1. Introduction

The rapid growth of web-based applications and ease of media manipulation gave rise to problems such as illegal copying and redistribution of digital media. This has made publishers, authors, artists, and photographers afraid that others will claim their innovations and products. The only way to protect these innovations is by attaching them with the owner's information robustly. For example, patient's name, age, and initial diagnosis are assumed as sensitive data and it usually attached with patient's medical images. However, the illegal manipulation of such attached information could lead to dangerous problems. For example, in medical images, changing the patient name could lead to a false diagnosis of a specific patient because images are improperly interchanged between patients. This is absolutely not accepted in medical applications. Thereby, healthcare institutions are ethically obliged to protect images as well as its related information, copyright, from being modified. Therefore, a technique for protecting the copyright of the digital media is needed.

Digital watermarking is a method of hiding information (watermark) into a host (cover) signal (image, audio, or video) so that the watermark can be detected or extracted later to make an assertion about the cover signal [1]. When digital watermarking is used for copyright protection of images, the embedded watermark must survive against attacks. The robustness of the watermarking scheme refers to the survivability of the embedded watermark against attacks [2]. Thus, in copyright protection schemes, the more robustness of the watermarking against attacks the more successful the watermarking algorithm. Digital watermarking techniques were initially used for limited intents, now it becomes a well-defined science with its own resourceful schemes. Presently, it is the core of many modern applications. To name a few amongst its innumerable applications, digital watermarking is employed in Copyright Protection (CP) [3], Content Authentication (CA) [4], Fingerprinting [5], Telemedicine or e-health [1], Copy Prevention or Control [6], Content Description [7], Secret Communication [8], and ID Card Security [9]. Generally, the effective watermarking scheme should satisfy certain requirements to be reliable, such as invisibility, imperceptibility, un-ambiguity, low complexity, and either robustness or fragility, based on the intended watermarking application [6]. In addition, digital watermarks should be difficult to be removed or modified without damaging the host signal.

In this paper, we propose a new digital watermarking algorithm that has the ability of embedding robust watermarks into digital images. Thereby, it is intended for the copyright protection of images. The embedded watermark will be robust against a verity of geometrical attacks. Many watermarking algorithms have been proposed to solve the problems of copyright protection. Each competes to accomplish higher robustness than others have. An intensive study and detailed analysis of the most promising proposed algorithms is performed, to find which of the proposed schemes can be adapted to meet our proposed algorithm requirements. According to their attractive properties (Secure, Blind, Reversible, and Easily Implemented); the idea of Chrysochos et al. algorithm [10] is chosen and developed to be used as a bases in building our proposed scheme.

The rest of this paper is organized as follows. In Section 2, we review the necessary background required to effectively implement our scheme. The related work for this paper is described in Section 3, while Section 4 proposes and discusses the new scheme. Experiments and analysis are presented in Section 5. Conclusion is given in the final section.

2. Background

2.1. Theoretical Model of the Digital Watermarking System

Digital watermarking is usually divided into two main processes: watermark embedding process and watermark extraction process.

The watermark embedding process embeds the watermark into the host signal. The watermark and the host signal are the inputs of this process. The watermarked signal is the output of this process. In some watermarking systems, a security key is used as additional input for the embedding process; it adds a level of security to the watermarking process and makes the watermark more robust against attacks. Figure 1 shows the watermark embedding process, which is adopted from [11].

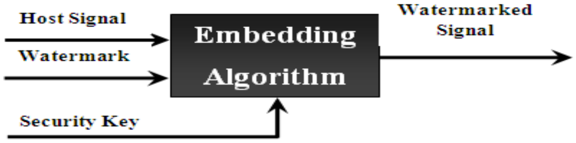


Figure 1. Watermark Embedding Process

The watermark extraction process extracts the watermark from the watermarked signal. The watermarked signal is the basic input of this process, the output of this process is the extracted watermark. If the embedding process used a security key, the same key is needed in the extraction process. The extraction process is exactly the reverse of the embedding process. Other inputs may be needed based on which type of watermarking algorithms is applied. For example, in the CA and CP systems the original watermark is also needed as an input to the extraction process for comparing with the extracted one in order to determine whether the host signal is authentic or not. Figure 2 shows the watermark extraction process, which is adopted from [11].

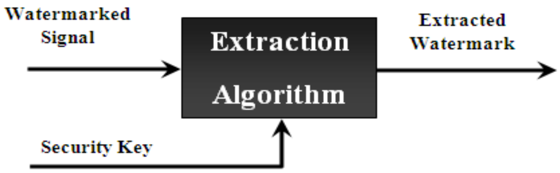


Figure 2. Watermark Extraction Process

2.2. Performance Measurements of a Watermarking Algorithm

2.2.1. Normalized Cross Correlation (NCC): NCC is an important performance parameter in any extracting module. Sometimes, it is needed to have a robust watermarking algorithm. Robustness means to have, approximately, undistorted extracted watermark even if the watermarked image is subjected to attacks. The NCC is used to verify the robustness of the watermarking systems by expressing the comparability between the extracted watermark and the original watermark quantitatively [12]. NCC is defined as seen in Equation 1 below [13].

$$\mathrm{NCC} = \frac{\sum_x \sum_y W(x,y)W'(x,y)}{\sqrt{(\sum_x \sum_y [W(x,y)^2]).(\sum_x \sum_y [W'(x,y)^2])}} \tag{1}$$

Where, W(x, y), W'(x, y) are the original watermark image and the extracted watermark image respectively. NCC is a value between 0 and 1. The larger the NCC values, the higher the watermark robustness.

2.2.2. Embedding capacity (EC): It is a measure to determine the ratio of information that can be embedded into the host image; it is defined in Equation 2 below:

$$\mathrm{EC} = \frac{\mathrm{Ne}}{\mathrm{N}} \tag{2}$$

Where, N and Ne denote the total number of the pixels and the total number of the embedded pixels respectively.

2.2.3. Mean Square Error (MSE): It is one of the simplest functions used to measure the distance between the host image and its watermarked version. Suppose we have an image I of size MxN and its watermarked version is I'. The MSE is defined in Equation 3 below:

$$\mathrm{MSE} = \frac{1}{M*N}\sum_{i=0}^{M-1}\sum_{j=0}^{N-1}[(R(i,j) - R'(i,j))^2 + (G(i,j) - G'(i,j))^2 + (B(i,j) - B'(i,j))^2] \tag{3}$$

Where $R(i,j)$, $G(i,j)$, $B(i,j)$, $R'(i,j)$, $G'(i,j)$, and $B'(i,j)$ are the pixels located at i^{th} column and j^{th} row of the host image I and watermarked image I' of the Red , Green, and Blue components respectively.

2.2.4. Peak Signal to Noise Ratio (PSNR): PSNR is used to measure how much the watermarked version of an image is similar to the original image. Suppose we have an image I and its watermarked version I'. Also suppose that the MSE values of the Red, Green, and Blue components are MSE_R, MSE_G, and MSE_B respectively. The PSNR is defined as shown in Equation 4 below:

$$\text{PSNR (dB)} = 10 \log_{10} \frac{\max I^2}{(\mathrm{MSE}_R + \mathrm{MSE}_G + \mathrm{MSE}_B)/3} \tag{4}$$

Where, maxI is the maximum pixel value of the original image. Typically, PSNR is measured in decibel units (dB). And the bigger the PSNR value, the better the watermark conceals [13]. In general, the processed image is acceptable to the human eyes if its PSNR is greater than 30 dB [1]. At that level, the processed or watermarked image will be visually very close to the original, un-watermarked, image.

3. Related Work

Coatrieux *et al.*, [14] presented a retina images watermarking algorithm, as an application of medical image watermarking. The watermark insertion process adds or subtracts at most one gray level to or from the pixels in the cover image. And hence, they claim that their algorithm has no distortions on the watermarked image. But Coatrieux *et al.*, algorithm has some inherent shortcomings, additive watermark insertion is not robust against image attacks, and transparency is also not enough high.

Wang *et al.*, [15], proposed a multipurpose watermarking scheme for both CP and CA of color images and videos. They embed a robust watermark, used for CP, and a reversible fragile watermark, used for CA, into different color components of a color image or video original frame. The robust watermark is embedded in the block's mean values of the green component of the color image, and then the digest (MD5 hash value) of the whole copyright-watermarked image is calculated to be used as fragile authentication watermark, which is reversibly embedded in the LSBs of blue component blocks of the copyright-watermarked image. Here, reversible embedding of the fragile watermark is used to eliminate its influence on the robust watermark, which is embedded before. Thus, in the extraction process, the bits of authentication (digest) are first extracted and then, the original copyright-watermarked image is perfectly recovered in order to generate the original digest, which is compared with the extracted one. If, by making a match, it is proved that the image is not tampered, the copyright watermark is then extracted and verified.

But, the payload capacity limit of the copyright watermark in this scheme is very low. For example, according to the division shown in their scheme, if we have an image of size 512x512, it is divided into 8x8 sub-blocks division. Then, the mean of each sub-block is calculated. Thus, now we get a matrix of 64x64 mean values. Now each 8x8 sized value group is embedded by 1 bit; thus we maximally can embed 8x8 sized binary watermark bits in the whole image. That is, 64 bits, which is very small to represent a copyright watermark. In our proposed work, we will allow larger copyright watermarks to be used. In addition, Wang et al. embed the copyright watermark in the block's mean values of the green component of the color image irreversibly. This will distort the overall behavior of the image permanently. This distortion might not be acceptable in applications working with sensitive imagery, like medical imaging [16]. By investigating some works in the literature, we found that algorithms that embed the watermark based on, quantization like in Kundur and Hatzinakos [17], bit-replacement like in Memon and Gilani [18], and truncation like in Karras [19], the original image could not be recovered from its watermarked version. In our work, we will find a completely reversible watermarking algorithm.

Tian [20], proposed a reversible watermarking method for gray scale images, which is based on difference expansion. In Tian's work, the difference between each neighboring pixel values of the image is calculated. Some of the calculated differences are selected for difference expansion and watermark bit embedding. Only expandable differences are selected, to avoid both overflow and underflow. The pixel's pairing could be horizontally, vertically, etc. If it is found that the difference h is expandable, it is replaced by a new difference h'=2h+bit, where bit is the watermark bit to be embedded. New pixel pair values are then calculated based on the new difference, h'. After embedding, a location map of all expanded differences is created to be used as a guide in the watermark extraction stage. The size of the location map is equal to the number of differences, or pixel pairs. A value of “1” is assigned in the location map to correspond to an expandable difference; otherwise, a value of “0” is assigned.

Li *et al.*, [21], described a reversible watermark embedding algorithm for CP of tongue color images, which is based on the prediction-error of the calculated four neighbor’s context prediction for both Red and Green components. To achieve reversibility, they used the same procedure followed by Tian [20]. The only difference is that, now they pass the prediction error of Red and Green components as input for difference expansion, rather than passing the values of the neighboring pixels. Also, they extend Tian’s method to deal with negative values.

From our point of view, the main drawback in the schemes of Tian [20] and Li *et al.*, [21] is that, extra information needs to be embedded other than the watermark, that is map of location. Additional drawbacks in Li *et al.*, [21] are: The watermark is embedded in a selected square area; same area is required to be selected in the extraction process. This requires that, the location and the dimensions of that square area must be published either by embedding or by transmission to the extractor. This will increase the payload and complexity of the algorithm. Also, Li *et al.*, algorithm exploits the high correlation that is inherent among the neighboring pixels of tongue images, in order to guarantee accurate prediction of pixel based on its neighbors, and to achieve small prediction-error. Thus, Li *et al.*, algorithm could not be used for other image types.

Chrysochos *et al.*, [10], present a reversible watermarking scheme aimed at embedding a binary watermark into a gray scale image. Watermark embedding in this scheme is mainly depends on the permutation of the histogram bins of the host image. Two keys are needed in this scheme. The first key is called the public key, which is a real number used to specify the watermark embedding area at the histogram of the host image. It is needed in both watermark embedding and extraction processes. The second key is called the private key, which is used for the restoration of the image. The integer part of the public key is called start, and it indicates the embedding starting point in the histogram of the host image. The decimal part of this key, multiplied by ten, is called step, which defines the minimum distance a couple of histogram bins may have. Each watermark bit, w, is embedded by first locating a couple (a, b) of intensity values, chosen according to start and step values, such that the corresponding histogram values, hist(a) and hist(b), are not equal. If the watermark bit, w, equals zero, then the histogram values, hist(a) and hist(b), are forced to be in an ascending order, otherwise, they are forced to be in descending order. In the watermark extraction stage, the public key is used to locate the same couple, (a, b), then, according to the values of hist (a) and hist(b), the previously embedded bit, w, is extracted. The private key is generated during the embedding process. For each intensity pair, (a, b), chosen for embedding, a bit, pk, of the private key is generated. The value of pk is set to zero if hist(a) and hist(b) are originally in an ascending order, otherwise, pk is set to one. The authors of this scheme claim that their scheme shows robustness against a variety of geometrical attacks.

But unfortunately, the maximum payload capacity of this scheme is very low. At the best case, where all histogram bin pairs are assumed as candidates for embedding, the payload capacity is 128 bits. If the scheme is applied for color images, in a way, such that each color component carries a portion of the watermark bits, the maximum payload capacity is 384 bits.

Yalman and Erturk [22], proposed a new histogram modification based data hiding technique, which modifies the histogram of the cover image for data hiding. The data hiding in their proposed scheme is based on the number of iterations (IN) of each brightness value (BV) of the cover image. Their scheme mainly depends on the arithmetic modulo operator. After determining the histogram of the cover image, the lowest and the highest BVs are determined and named, Lower Limit Value (LLV), and Upper Limit Value (ULV) respectively. These two values specify the area where the data is going to be embedded. Each histogram’s IN is used to embed a watermark bit. For each bit to be embedded, the modulo2 of the corresponding IN is checked, if it equals the value of the bit, the bit is assumed embedded successfully. Otherwise one pixel of the image with the corresponding BV to that IN is changed to the next following BV, to enforce both the value of the bit, and the value of module2 of the corresponding IN are equal. The maximum embedding capacity of this scheme is 256 bits, provided that the image histogram is uniformly distributed from “0” to “255”, and hence, the 256 bins are assumed as candidates for embedding. In addition, the maximum payload capacity is 768 bits if the scheme is applied for color images. Therefore, the embedding capacity of Yalman and Erturk scheme is approximately twice the embedding capacity of the aforementioned, Chrysochos *et al.*, scheme [10]. This is due to the embedding strategy followed in Yalman and Erturk scheme, in which each histogram bin is embedded with a watermark bit. But in Chrysochos *et al.*, scheme, each non-equal couple of bins is embedded by a watermark bit. But unfortunately, the scheme of Yalman and Erturk is not reversible. Also from our point of view, we see that their scheme is not that robust, because any change to a pixel value will lead to a different corresponding number of iterations, IN, and hence, leading to a different modulo2 value.

Finally, and after reviewing those recently proposed watermarking schemes, we conclude that we still need new watermarking schemes, to cope with today’s requirements. According to its attractive properties regarding robustness, low computational cost, blind, and reversible the scheme of Chrysochos *et al.*, [10], will be improved in our work, to solve its limitations. After refinement, it will contribute significantly in our proposed CP scheme.

4. The Proposed Scheme

4.1. Overview

In this paper, we propose a CP watermarking scheme. It is a reversible, secure, and blind watermarking scheme; the embedded watermark can be detected apart from the original image. In addition, it is a robust watermarking scheme. Thus, it aims at protecting the copyright of the color images. It will be robust against some geometrical attacks. Our proposed scheme is a development of the technique proposed by Chrysochos *et al.*, [10]. Their proposed scheme aimed at proving the ownership of digital images by embedding a robust watermark in it. The basic principle of their proposed scheme is based on the permutation of the histogram bins according to a specific rule. In other words, each bit is embedded by permuting a corresponding couple of the histogram bins according to that rule. The authors refer to an inherent drawback in their proposed scheme: The maximum payload capacity of the scheme is rather low. Usually, robustness to geometrical attacks is obtained at the expense of the payload capacity [23, 24]. However, we performed an intensive study, implementation, and testing to their proposed scheme, in an attempt to address this drawback. We have reached a new idea for embedding, which is employed in our proposed watermarking scheme. The next subsections demonstrate our achieved idea.

4.2. The Conceptual Model of the Proposed CP Watermarking Scheme

The conceptual model of the proposed CP technique is summarized in Figure 3, below. The main functionalities of Chrysochos *et al.*, watermarking scheme are developed to handle the previously mentioned drawback.

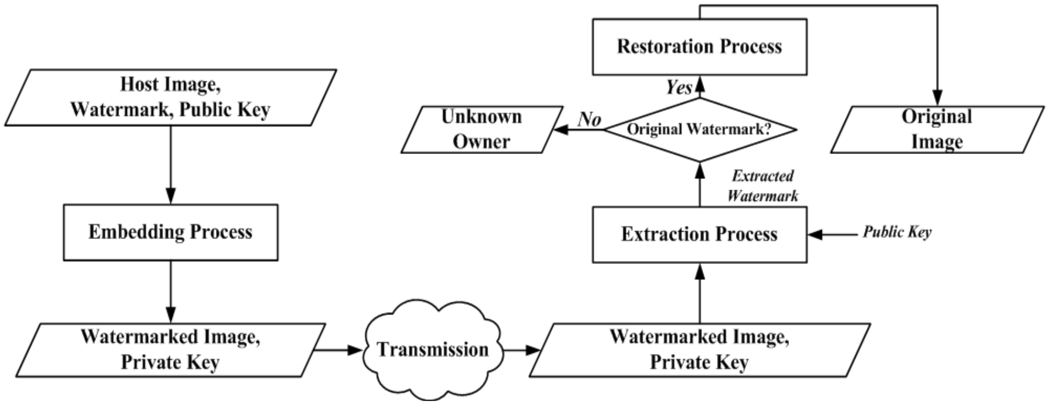


Figure 3. The Conceptual Model of the Proposed CP Watermarking Scheme

As seen in Figure 3, our proposed CP watermarking scheme consists of three main processes. The first main process is the embedding process, which is responsible for embedding the watermark in a way, such that it is embedded robustly. This process depends on three inputs, the original image to be watermarked, the watermark, and a public key. Two outputs are produced, the watermarked image, and the private key, which will be used later for image restoration. The second process is the extraction process, which is responsible for detecting the previously embedded watermark. The watermarked image and the same public key, used for embedding, are the two inputs used by this process. The extracted watermark is its output. Finally, the third process in our proposed scheme is the restoration process, which is responsible for removing the previously embedded bits from the watermarked image completely, and recovering the original, un-watermarked, image. This process runs only if the originality of the extracted watermark is verified. In addition to the public key, the restoration process is also based on the private key, which is generated during the embedding process. The next subsections discuss each of these three main processes individually.

4.3. The Watermark Embedding Process of the Proposed CP Watermarking Scheme

Aiming to increase the maximum payload capacity of Chrysochos *et al.*, scheme, a new embedding mechanism is proposed. The public key and the permutation of the histogram bins are still used in our proposed scheme. But the embedding rule is completely different. Now, a couple of bits are embedded at a time, by permuting three histogram bins. While, in Chrysochos *et al.*, scheme, one bit is embedded at a time, by permuting two histogram bins. In our scheme, the public key is still a real number, but it has a different use than that in Chrysochos *et al.*, scheme. The integer part of this key is called (begin), which refers to the point at the histogram where the embedding process will begin choosing triples, rather than couples, of intensity values (a, b, and c) with a corresponding histogram bins (Hist (a), Hist (b), and Hist (c)). Thus, the integer part of the public key, begin, could be any integer value in the interval [0, 255]. The decimal part of the public key is called (step), which is a single digit used to define the intensity interval the three intensity values may occupy. Equation 5, demonstrates how to calculate the value of begin, by extracting the integer part of a publicKey. And Equation 6 demonstrates how to calculate the value of step, by extracting the decimal part of that key.

$$\text{begin} = \text{publicKey}/1 \tag{5}$$

$$\text{step} = (\text{publicKey} \,\%\, 1) * 10 \tag{6}$$

Based on the calculated values of begin and step, the three intensity values, a, c, and b are calculated as shown in Equations 7, 8, and 9 respectively.

$$\mathrm{a} = \mathrm{begin} \tag{7}$$

$$\mathrm{c} = (\mathrm{begin} + \mathrm{step})\ \%\ 256 \tag{8}$$

$$\mathrm{b} = \left\lfloor \frac{(\mathrm{a}+\mathrm{c})}{2} \right\rfloor \tag{9}$$

As seen in the equations, the intensity value, b, is chosen as a midpoint between the two values a, and c. Therefore, a sufficient distance between a and c is needed. This is obtained by restricting the value of step in the interval [2, 9]. Moving to the next triple is simply achieved by cyclically incrementing each intensity value by one, as seen in Equation 10.

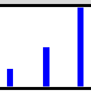
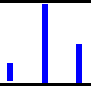
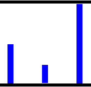
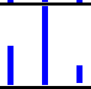
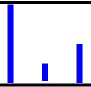
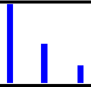
$$\mathrm{a}\ = (\mathrm{a} + 1)\%256;\ \mathrm{b} = (\mathrm{b} + 1)\%256;\ \ \mathrm{c} = (\mathrm{c} + 1)\ \%256 \tag{10}$$

At each triple calculation, we test whether c is less than b; to avoid the situation where a and b are at the end of the histogram while c is at the beginning. If that occurs, the triple is moved to the beginning of the histogram as seen in Equation 11.

$$\mathrm{a}\ = 0;\ \ \mathrm{c} = \mathrm{step};\ \mathrm{b} = \left\lfloor \frac{\mathrm{step}}{2} \right\rfloor \tag{11}$$

Each intensity values triple, (a, b, c), has a corresponding histogram bins triple, (Hist(a), Hist(b), Hist(c)). In our proposed scheme, to obtain a different pattern for each permutation of the bins triple, we are concerned only with intensity triples whose corresponding histogram bins are strictly non equal, Mean: Hist(a) ≠ Hist(b) ≠ Hist(c). Excluding triples those have equal histogram bins, the magnitude of each histogram bin within a triple might be higher than, lower than, or in between the other two histogram bins at the same triple. Consequently, each intensity triple will has a corresponding Bins Triple Pattern (BTP), which will be one of the six patterns shown in Table 1 below.

Table 1. List of Bins Triple Patterns (BTPs), used in our Proposed CP Watermarking Scheme

Bins Triple Pattern (BTP)	Shape of the pattern	Interpretation of the pattern
0		Low, Between, High
1		Low, High, Between
2		Between, Low, High
3		Between, High, Low
4		High, Low, Between
5		High, Between, Low

As mentioned before, in our proposed scheme, each couple of bits will be embedded by permuting a selected triple of histogram bins. Each couple of bits has a corresponding Bits Couple Pattern (BCP), which may equal "00", "01", "10", or "11". Embedding each selected couple is performed by the permutation of the corresponding triple of histogram bins according to the rule seen in Table 2. As seen in the table, the rule maps each BCP ("00", "01", "10", and "11") to a corresponding BTP (0, 1, 4, and 5). Also, to achieve lower computational complexity, the remaining two BTPs (2 and 3) are also exploited to represent the BCPs "00" and "11". Hence, if the current couple of bits to be embedded has a BCP = "00" and the corresponding triple of bins has a BTP = 2, no permutation is performed and the couple is assumed as embedded successfully. This also applies to the situation, when the current couple of bits to be embedded has a BCP ="11" and the corresponding triple of bins has a BTP = 3. This of course, will reduce the computational complexity of our proposed CP watermarking scheme, especially when we have noticed that the two BCPs, "00" and "11", are highly repeated in binary watermarks.

Because we embed the watermark in couples, it is required to have even watermark sizes. If the size of the watermark to be embedded is odd, the watermark is appended by a new bit, equals zero, from the right most side.

Table 2. The Embedding Rule of our Proposed CP Watermarking Scheme

Bits Couple Pattern (BCP)	The corresponding rule for embedding
"00"	IF (BTP = 0 OR BTP = 2), do nothing; Otherwise, permute the current bins triple until having a corresponding BTP = 0.
"01"	IF (BTP = 1), do nothing; Otherwise, permute the current bins triple until having a corresponding BTP = 1.
"10"	IF (BTP = 4), do nothing; Otherwise, permute the current bins triple until having a corresponding BTP = 4.
"11"	IF (BTP = 5 OR BTP = 3), do nothing; Otherwise, permute the current bins triple until having a corresponding BTP = 5.

We can embed the watermark in either histogram of the three color components of the host image. Or embed the watermark collectively in the histograms of the three color components, in a way, such that each color component carries a portion of the watermark bits. Anyway, the detailed steps for our histogram-based watermark embedding process of our proposed CP watermarking scheme are illustrated in Algorithm 1 bellow.

Algorithm 1: The embedding process of the proposed CP watermarking scheme.

Purpose: Watermark embedding, based on the permutation of the bins of the histogram of host color plane.

Input: The host color plane, the watermark to be embedded, and the publicKey.

Output: The watermarked image, and the private key, PK.

Procedure:

a) The histogram of the host color plane is computed.
b) Based on the publicKey, the values of begin and step are calculated using Equations 5, and 6 respectively.

c) If no intensity triple is selected yet, an intensity triple (a, b, c) is calculated based on the values of begin and step, and using Equations 7, 8, and 9; otherwise, the next intensity triple, (a, b, c), is calculated using Equation 10. In either case, the selected intensity triple must have a corresponding strictly not equal histogram bins, or reject the selected triple and select the next one using Equation 10. An attention must be paid at each triple selection; to avoid colloid with the previously selected triples, *i.e.*, to avoid using a previously embedded triple or we will lose that old embedding.

d) For each selected intensity triple, (a, b, c), having a corresponding triple of bins, (Hist(a), Hist(b), Hist(c)), with bins triple pattern, BTP; a corresponding watermark's bit couple with bits couple pattern, BCP, is embedded according to the rule shown in Table 2. If the rule asks to permute the current triple of bins until having a new corresponding BTP, the image pixels with intensities a, b, and c are interchanged accordingly.

e) The private key (PK), which is necessary for image restoration, is generated accumulatively during the watermark embedding. For each selected intensity triple, (a, b, c), the original BTP is appended to the PK. (This step is optional; it is done only if we are interested in restoring the original image, later)

f) Steps c, d, and e are repeated until all watermark bits are embedded in the color plane, or the algorithm reaches the capacity limit of the host color plane, *i.e.*, all the candidate triples of the histogram bins are already embedded.

END

4.4. The Watermark Extraction Process of the Proposed CP Watermarking Scheme

It is a blind process, which aims at detecting the previously embedded watermark from the watermarked image without the presence of the original image. This process mainly based on the extraction rule shown in Table 3 below, which is derived from the aforementioned embedding rule of Table 2.

Table 3. The Extraction Rule of our Proposed CP Watermarking Scheme

BTP	The corresponding extracted Bits Couple Pattern, BCP.
IF BTP = 0 OR BTP = 2	BCP = “00”
IF BTP = 1	BCP = “01”
IF BTP = 4	BCP = “10”
IF BTP = 5 OR BTP = 3	BCP = “11”

The detailed steps for extracting the watermark from a watermarked color plane are illustrated in Algorithm 2. As seen in the algorithm, the public key and the watermark size along with the watermarked plane are the parameters needed for watermark extraction process. The extracted watermark is the process output.

Algorithm 2: The extraction process of the proposed CP watermarking scheme.

Purpose: Watermark extraction, based on the histogram of watermarked color plane.

Input: The watermarked color plane, watermark size, S, and the publicKey.

Output: The extracted watermark, EW.

Procedure:

a) The histogram of the watermarked color plane is computed.

b) Based on the publicKey, the values of begin and step are calculated using Equations 5, and 6 respectively.

c) If no intensity triple is selected yet, an intensity triple (a, b, c) is calculated based on the values of begin and step, and using Equations 7, 8, and 9; otherwise, the next intensity triple, (a, b, c), is calculated using Equation 10. In either case, the selected intensity triple must have a corresponding strictly not equal histogram bins, or reject the selected triple and select the next one. An attention must be paid at each triple selection; to avoid colloid with the previously selected triples, *i.e.*, to avoid extracting from a previously used triple or we will duplicate the extracted bits.

d) From each selected intensity triple, (a, b, c), having a corresponding triple of bins, (Hist(a), Hist(b), Hist(c)), with bins triple pattern, BTP, the embedded couple of bits, with bits couple pattern, BCP, is extracted according to the rule shown in Table 3, The extracted couple of bits are appended to the extracted watermark, EW.

e) Steps c and d are repeated until all watermark bits are extracted, *i.e.*, until the size of the extracted watermark, EW, equals S.

END

Note that the extraction process can only be successful if its selected intensity triples are same as those selected during the embedding process.

4.5. The Restoration Process of the Proposed CP Watermarking Scheme

In some applications, it is needed to restore the watermarked image after being verified. Restoration is the process of recovering the original, un-watermarked, image from its watermarked version. The detailed steps of the restoration process of our proposed CP watermarking scheme are listed in Algorithm 3. This process mainly depends on the private key, PK, which was generated during the watermark embedding process; it carries the original list of bins triple patterns, BTPs.

Algorithm 3: The restoration process of the proposed CP watermarking scheme.

Purpose: Recovering the original, un-watermarked, image from the watermarked one.

Input: The watermarked plane, the publicKey, and the private key, PK.

Output: Restored Image.

Procedure:

a) The histogram of the watermarked plane is computed.

b) Based on the publicKey the values of begin and step are calculated using Equations 5 and 6 respectively.

c) If no intensity triple is selected yet, an intensity triple (a, b, c) is calculated based on the values of begin and step, and using Equations 7, 8, and 9; otherwise, the next intensity triple, (a, b, c), is calculated using Equation 10. In either case, the selected intensity triple must have a corresponding strictly not equal histogram bins, or reject the selected triple and select the next one. An attention must be paid at each triple selection; to avoid colloid with the previously selected triples.

d) The bins triple pattern, BTP, of the selected intensity triple, (a, b, c), is compared with the corresponding bins triple pattern, BTP, which was stored in the private key, PK. If a match is found, then the BTP of the selected intensity triple is original and left as it. Otherwise, the BTP of the selected intensity triple is permuted to match that in PK; and consequently, pixels with intensities (a, b, c) are interchanged according to that permutation.

e) Steps c and d are repeated until reaching the end of the private key, PK.

END

5. Experimental Results

To confirm its position among the others; our proposed CP watermarking scheme is compared with its counterpart, Chrysochos et al. watermarking scheme, which was developed in generating our proposed CP watermarking scheme. So, we implemented Chrysochos *et al.*, scheme, and then our tests are performed to evaluate our proposed scheme relative to their scheme. All tests are performed using a laptop running windows XP operating system, with a 2 GHz core 2 duo processor, 2 GB memory, and 384 MB display adapter. Matlab 7.8 and Visual Studio 2011 are the main software components used in our work.

The broad goal of our proposed scheme is to protect color images; therefore, a general purpose image database is obtained, which includes 1000 color images, those are given in IPEG format, with size 384 x 256 or 256 x 384. This database was previously used in [25], and we downloaded it from [26]. Database images are grouped into ten categories; including, African people, Beach, Buildings, Cars, Dinosaurs, Elephants, Flowers, Houses, Mountains, and Food. Also, because our proposed schemes may be directed to medical applications; a set of 50 colored medical images is established and joined to the aforementioned database under a new category named Medical. This medical set is obtained randomly from the Science Photo Library [27]; it is given in JPEG format in RGB color space. Therefore, our proposed schemes are evaluated using a dataset contains 1050 samples.

5.1. Comparing the Embedding Capacity

Theoretically, the maximum possible payload capacity per color plane using our proposed CP scheme is 170 bits. Particularly, we have (256/3) = 85 triples of histogram bins, each of which will be embedded by a couple of bits. On the other hand, the maximum possible payload capacity per color plane using Chrysochos *et al.*, scheme it is 128 bits. Particularly, we have (256/2) = 128 pair of histogram bins, each of which will be embedded with a single bit. If the three color planes are used for embedding, these capacities may rise to three times the old capacities, *i.e.*, it becomes 510 bits in our scheme, and 384 bits in their scheme.

Practically, each of Chrysochos *et al.*, watermarking scheme and our proposed CP watermarking scheme is used in turn to embed a watermark of size 512 bits into each of the samples in our dataset. The embedding capacity for each watermarked sample is recorded. Also, the average embedding capacity is calculated over each category in our dataset. Figure 4 shows the results of this experiment. As seen in the figure, in each category, the maximum payload capacity obtained when using our proposed CP watermarking scheme surpasses that of Chrysochos *et al.*, watermarking scheme. Numerically, the average payload capacity, over all samples in the dataset, using our proposed CP scheme is 417 bits, while it is 362 bits when using Chrysochos *et al.*, scheme. Thus, our proposed CP watermarking scheme increased the average payload capacity, per each sample, by about 55 bits.

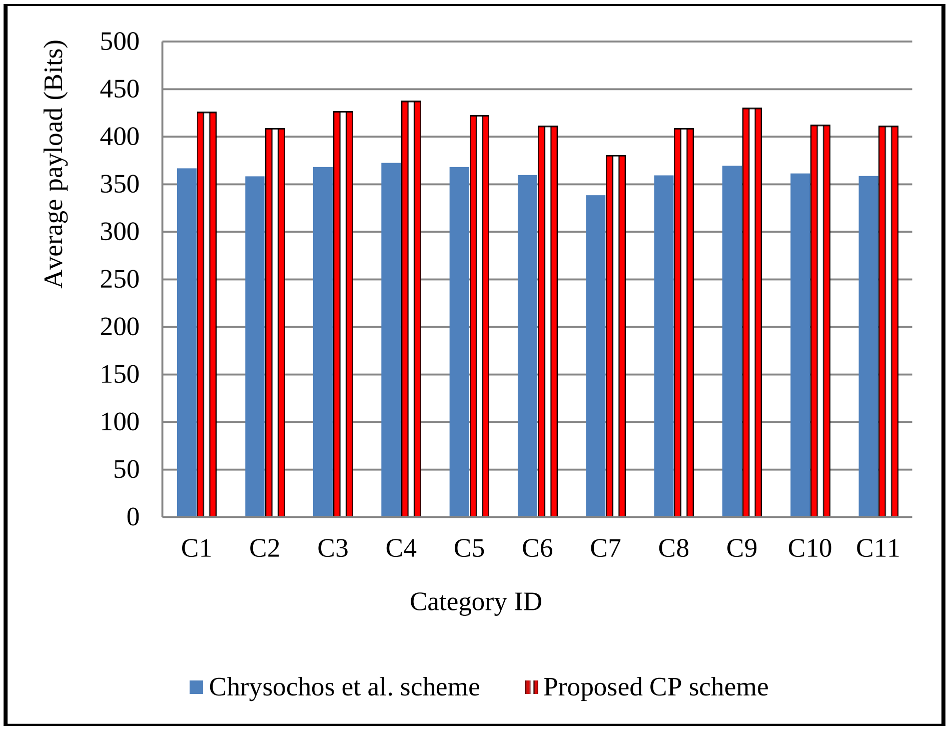


Figure 4. Comparing the Average Embedding Capacity between Chrysochos et al. Scheme and our Proposed CP Scheme

The interpretation for this increase at the payload capacity, obtained using our proposed CP watermarking scheme, is that, we embed every two watermark bits by permuting three histogram bins. While, Chrysochos *et al.*, embed each one bit of the watermark by permuting two histogram bins. Thus, our embedding mechanism exploits the histogram bins more than its counterpart. Finally, we noted that there is no direct relation between the ***step*** value of the public key and the embedding capacity of the two watermarking schemes.

5.2. Comparing the Effect of Watermark Embedding on the Host Image

Preserving the quality of the host image after being watermarked is one of the main success factors of any watermarking scheme. Therefore, the quality of the watermarked image is our main concern in this section. Measuring the quality of the watermarked image is an estimate of how much the watermarked image is similar to the original, un-watermarked, one. The PSNR, seen in Equation 4, is used as a measurement of the quality of the watermarked image. Both the Chrysochos *et al.*, watermarking scheme and our proposed CP watermarking scheme are used in turn to embed a fixed watermark into each of the samples in our dataset. The quality, PSNR, of each watermarked sample is calculated. Finally, the calculated values of the PSNR are averaged over every category of samples. Figure 5 shows the obtained results.

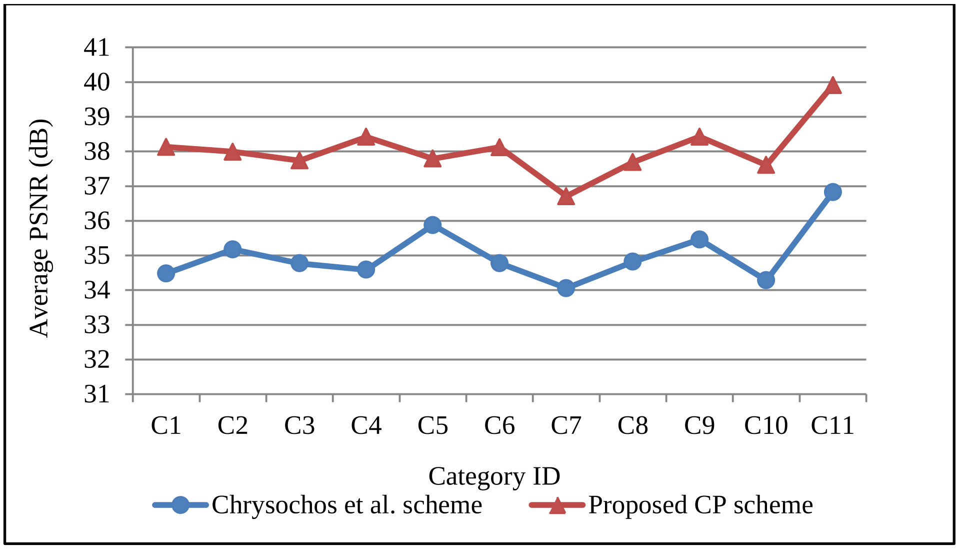


Figure 5. Comparing the Quality of the Watermarked Image for both Chrysochos et al. Scheme and our Proposed CP Scheme

As seen in Figure 5, for all categories, our proposed CP scheme generates higher quality watermarked images than those obtained when using Chrysochos *et al.*, scheme. The average PSNR over all watermarked samples using our scheme is, 38.05 dB, while using Chrysochos et al. scheme is, 35.01 dB. This increase in the quality is due to the embedding strategy of our proposed scheme, which performs a smaller number of histogram permutations than those performed using Chrysochos *et al.*, scheme, in embedding the same watermark. Embedding two bits using our proposed CP scheme requires at most permuting three histogram bins, while embedding two bits using Chrysochos *et al.*, scheme requires at most permuting four histogram bins. In addition, during watermark embedding in our proposed CP scheme, some Bins Triple Patterns, BTP, namely 2 and 3, are left unchanged when facing the Bits Couple Patterns, BCPs "00" and "11". Thus, our proposed scheme has fewer side effects on the host image if compared with its counterpart. Finally, it is noticed in this experiment that, increasing the step of the public key; decreases the quality of the watermarked image using Chrysochos *et al.*, scheme more quickly than when using our proposed scheme. The interpretation is that in Chrysochos *et al.*, scheme, when we increase the step of the public key, the distance occupied by the two selected histogram bins increased. Thereby, all the interchanged pixels, for embedding, will have far intensity values. But because our proposed CP scheme depends on three histogram bins, that occupied distance is divided into two smaller distances. Thereby, some of the interchanged pixels will have closer intensity values.

5.3. Comparing the Embedding Time

Chrysochos *et al.*, watermarking scheme and our proposed CP watermarking scheme, each is used in turn to embed a fixed watermark into each of the samples in our dataset. We recorded the embedding time at each sample in the dataset. Finally, the average embedding time is calculated for each category. Figure 6 shows the obtained results for the two schemes.

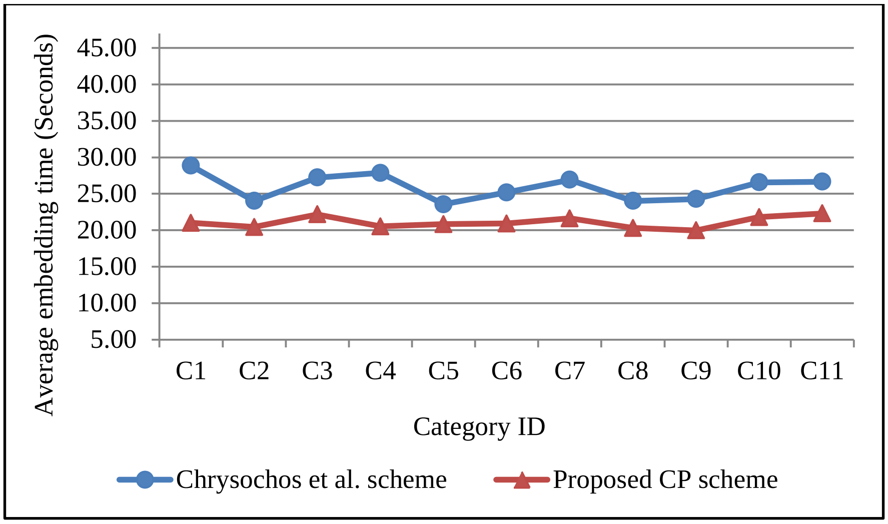


Figure 6. Comparing Watermark Embedding Time between Chrysochos et al. Scheme and our Proposed CP Scheme

As seen in Figure 6, it is evident that our proposed CP algorithm is faster than Chrysochos *et al.*, algorithm in embedding watermarks. The average embedding time over all samples using the proposed CP scheme is 21.08 seconds, while using Chrysochos *et al.*, scheme is 25.92 seconds. The interpretation of the superiority of our proposed scheme to its counterpart in obtaining less embedding time comes as follows: Assume that we seek to embed a watermark based on the histogram of a host image using each of the schemes in turn. The total number of the histogram bins permuted, and consequently, the total number of pixels interchanged using our proposed scheme is less than that when using Chrysochos *et al.*, scheme. That is because, three histogram bins are permuted in our proposed CP scheme to embed two bits, while two histogram bins are permuted in Chrysochos *et al.*, scheme to embed one bit. For example, if we seek to embed a binary watermark of 6 bits, at most we need to permute 9 bins using our proposed scheme, while 12 bins are permuted using Chrysochos *et al.*, scheme.

5.4. Comparing the Robustness Against some Geometrical Attacks

Robustness of the watermarking scheme against geometrical attacks means that the embedded watermark must survive against geometrical attacks those might be applied to the host image. In other words, despite of geometrically transforming the host image, the embedded watermark must still be detectable by the watermarking scheme.

Both Chrysochos *et al.*, scheme and our proposed CP scheme embed watermarks based on the histogram of the host image. Therefore, both schemes are robust only against geometrical attacks those change pixels positions, but not against those change the histogram of the host image. In this section, the robustness of the proposed CP watermarking scheme against such geometrical attacks is tested. In this test, each sample in the dataset is embedded with the copyright watermark shown in Figure 7, which is a binary watermark of size 20x20. The used public key is, 16.7.

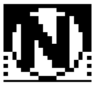


Figure 7. Binary Watermark (Size 20X20)

Now, to evaluate the robustness of our proposed CP watermarking scheme against some geometric attacks those change only pixels positions, each watermarked sample is subjected to some geometrical attacks; including, Flipping, Rotation, Scattering, Warping, and Skewing. For each attack application at a given watermarked sample, the embedded watermark is extracted and evaluated using the NCC. The NCC values are averaged over all the samples, those were modified by a given attack type. Table 4 summarizes the obtained results.

Table 4. The Experimental Results for Testing the Robustness of the Proposed CP Watermarking Scheme against some Geometric Attacks

Attack Name	Parameters	The NCC of the extracted watermark
Flipping	H, V, and Both	1
Rotation	90°, 180°, and 270°	1
Scattering	Any degree	1
Warping	Any degree	1
Skewing	Any degree	1

As seen in the table, our proposed CP watermarking scheme showed 100 % robustness against some geometrical attacks, *i.e.*, the detected watermark exactly matches the original embedded one. Such attacks are flipping (horizontally, vertically, both), rotation (90°, 180°, 270°), scattering, warping and skewing, as well as their combinations. The interpretation for this robustness is due to the fact that our proposed scheme embeds the watermark by modifying the histogram of the host image, which is mostly not affected after applying the aforementioned attacks to the host image. Those attacks change only the positions of the pixels. Mean, every pixel in the original image goes to a predefined point. This predefinition of new pixels positioning, specifies the type of the transformation. For example, at the horizontal flipping, the pixels of the original image are mirrored across a horizontal axis. The histogram of the image does not depend on the positions of pixels, but it depends on the number of pixels. Therefore, such image modifications will not change the histogram of the watermarked image. Thereby, we conclude that our proposed CP watermarking scheme is robust 100% against geometrical attacks those only change pixels positions, but not against those change the histogram of the watermarked image.

6. Conclusion

A copyright protection watermarking scheme is proposed in this paper. It is a robust, blind and reversible watermarking scheme for CP of images. It is a development of the scheme proposed by Chrysochos *et al.*, [10], which mainly depends on the permutation of the image's histogram bins. The authors referred to an inherent drawback in their proposed scheme; the theoretical maximum embedding capacity of their scheme is rather low (384 bits per color image). An attempt for adapting this drawback is presented in our proposed CP watermarking scheme. A new embedding mechanism is proposed, which is still using the permutation of the histogram bins for watermark embedding, but with a completely different embedding rule. Theoretically, the maximum payload capacity of our proposed CP watermarking scheme is 510 bits, which is larger than that of Chrysochos et al. scheme by 126 bits.

Our proposed scheme is evaluated and compared with its counterpart. Tests are performed based on a dataset of 1050 samples from eleven categories. Results are averaged over all the dataset samples. Our proposed scheme showed a noticeable superiority over its counterpart in terms of capacity, quality, time and robustness against some geometrical attacks. Finally, our proposed scheme is based on the spatial domain, which requires a lower computational cost than that required in transform domain based schemes.

References

[1] M. Prasad and S. Koliwad, “A Comprehensive Survey of Contemporary Researches in Watermarking for Copyright Protection of Digital Images”, International Journal of Computer Science and Network Security, vol. 9, no. 4, **(2009)** April, pp. 91-107.

[2] Y. Hu, H. Lee and H. Zeng, “Curve Watermarking Technique for Fingerprinting Digital Maps”, 8th International Conference on Intelligent Information Hiding and Multimedia Signal Processing, **(2008)**, pp. 223-226.

[3] S. Poonkuntran and R. S. Rajesh, “A Messy Watermarking for Medical Image Authentication”, Proceedings of 2011 International Confrence on Communications and Signal Processing, Kerala, **(2011)**, pp. 418-422.

[4] X. Ma and X. Shen, “A novel blind grayscale watermark algorithm based on SVD”, Proceedings of International Conference on Audio, Language and Image Processing, (ICALIP ‘08), Shanghai, **(2008)** July, pp. 1063-1068.

[5] N. Yadav, N. Pahal, P. Kalra, B. Lall and S. Chaudhury “A Novel Approach for Securing Forensic Documents Using Rectangular Region-of-Interest (RROI)”, Proceedings of the 2nd International Conference on Emerging Applications of Information Technology, (EAIT), Kolkata, **(2011)**, pp. 198-201.

[6] E. Brannock, M. Weeks and R. Harrison, “The Effect of Wavelet Families on Watermarking”, Journal of Computers, vol. 4, no. 6, **(2009)** June, pp. 554-566.

[7] Mrdjenovic and Ljiljana, “Digital watermarking in the generalized discrete cosine transform domain”, M.S. thesis, Dept. Comput. Sci., York University, Toronto, Ontario, Jan. **(2010)**.

[8] L. Fan, T. Gao, Q. Yang and Y. Cao, “A copyright-protection watermark mechanism based on generalized brain-state-in-a-box neural network and error diffusion halftoning”, IEEE International Conference on Multimedia and Expo, (ICME), China, **(2011)**, pp. 1-6.

[9] C. Xiaoling and Z. Huimin, “A Novel Video Content Authentication Algorithm Combined Semi-fragile Watermarking with Compressive Sensing”, 2nd International Conference on Intelligent System Design and Engineering Application, (ISDEA), **(2012)**, pp. 134-137.

[10] E. Chrysochos, V. Fotopoulos, A. Skodras and M. Xenos, “Reversible Image Watermarking Based on Histogram Modification”, Proceedings of the 11th Panhellenic Conference on Informatics with international participation, (PCI 2007), Patras, Greece, vol. B, **(2007)** May, pp. 93-104.

[11] S. Mohanty, “Digital Watermarking: A Tutorial Review”, Master Project Report, Dept. of Electrical Engineering, India, Institute of Science, DANGALORE, India, **(1999)**.

[12] M. Jiansheng, L. Sukang and T. Xiaomei, “A Digital Watermarking Algorithm Based On DCT and DWT,” Proceedings of the International Symposium on Web Information Systems and Applications, (WISA’09), Nanchang, P. R. China, **(2009)**, pp. 104-107.

[13] A. Zeki and A. Manaf, “A novel digital watermarking technique based on ISB (Intermediate Significant Bit)”, World Academy of Science, Engineering and Technology, vol. 50. **(2009)**, pp. 989-996.

[14] G. Coatrieux, M. Lamard, W. Daccache, J. Puentes and C. Roux, “A Low Distorsion and Reversible Watermark: Application to Angiogeraphic Images of the Retina”, Proceedings of the 27th IEEE Conference on Engineering in Medicine and Biology, Shanghai, China, **(2005)**, pp. 2224-2227.

[15] Z. Wang, B. Yang, X. Niu and Y. Zhang, “A Practical Multipurpose Watermarking Scheme for Visual Content Copyright Protection and Authentication”, Proc. 2006 Int. Conf. Intelligent Information Hiding and Multimedia Signal Processing, (IIH-MSP'06), **(2006)** China, pp. 461-464.

[16] S. C. Liew and J. M. Zain, “Reversible Medical Image Watermarking For Tamper Detection And Recovery”, Proceedings of the International Conference on Computer Science and Information Technology, (ICCSIT), Chengdu, **(2010)**, pp. 417-420.

[17] D. Kundur and D. Hatzinakos, “Digital Watermarking for Telltale Tampering Proofing and Authentication”, Proceedings of the IEEE, vol. 87, no. 7, **(1999)**, pp. 1167-1180.

[18] N. A. Memon, S. A. M. Gilani and A. Ali, “Watermarking of Chest CT Scan Medical Images for Content Authentication”, International Conference of Information and Communication Technologies, Karachi, **(2009)** August, pp. 175-180.

[19] D. A. Karras, “A Second order Spread Spectrum Modulation Scheme for Wavelet based Low Error Probability Digital Image Watermarking”, International Journal on Graphics, Vision and Image Processing, (GVIP), vol. 5, no. 3, **(2005)**.

[20] J. Tian, "Reversible Watermarking Using a Difference Expansion", IEEE Transactions on Circuits and Systems for Video Technology, vol. 13, no. 8, **(2003)**, pp. 890-896.

[21] L. Li, Y. Fan and C. Chang, "A Reversible Watermarking Algorithm Based on Four-Neighbors Context Prediction for Tongue Images", International Journal of Intelligent Information Processing, (IJIIP), vol. 2, no. 2, **(2011)**, pp. 22-28.

[22] Y. Yalman and I. Erturk, "A new histogram modification based robust image data hiding technique", 24th International Symposium on Computer and Information Sciences, (ISCIS), Guzelyurt, Turkey, **(2009)**, pp. 39-43.

[23] Z. Fang and Y. Zhao, "Image Watermarking Resisting to Geometrical Attacks Based on Histogram", International Conference on Intelligent Information Hiding and Multimedia, **(2006)**, pp. 79-82.

[24] M. Dainaka, S. Nakayama, I. Echizen and H. Yoshiura, "Dual-Plane Watermarking for Color Pictures Immune to Rotation, Scale, Translation, and Random Bending", International Conference on Intelligent Information Hiding and Multimedia, **(2006)**, pp. 93-96.

[25] J. Li and J. Z. Wang, "Real-time Computerized Annotation of Pictures", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 6, **(2008)**, pp. 985-1002.

[26] http://wang.ist.psu.edu/docs/related.shtml/, James Z. Wang Research Group, Accessed **(2013)** January 17.

[27] http://www.sciencephoto.com/, online image library, Accessed **(2013)** January 21.

Authors

Nader H. H. Aldeeb has received his B.Sc. degree in computer engineering and Master of Science degree in computer engineering from the Islamic University, Gaza, Palestine, in 2005 and 2012, respectively. Currently, he works as a computer engineer in the Programming and Development Department of the Central Computer Unit in Palestine. His research interests include data mining, image processing, machine learning, pattern recognition, artificial intelligence, and other fields.

Ibrahim S. I. Abuhaiba is a professor at the Islamic University of Gaza, Computer Engineering Department. He obtained his Master of Philosophy and Doctorate of Philosophy from Britain in the field of document understanding and pattern recognition. His research interests include computer vision, image processing, document analysis and understanding, pattern recognition, artificial intelligence, and other fields. Prof. Abuhaiba presented important theorems and more than 30 algorithms in document understanding. He published several original contributions in the field of document understanding in well-reputed international journals and conferences.