HidingFingerprint byUsing PIFS and DCT

Kadhim H. Kuban
Alshemlchy62@yahoo.com
Firas S. Miften
firassebar@yahoo.com
Wessam A.Hamed
wessam.abbas1980@yahoo.com
Thi_Qar University, College of Education for Pure Sciences,Computer Department

Abstract

In this paper, we propose method for hiding the fingerprint in the person's image where it can be used as authentication when failing to recognize the person's image. It compresses a fingerprint image using PIFS to get the transformation coefficients to be used later to hide in the image which applied DCT and hides the PIFS coefficient randomly in non-zero DCT coefficient using LSB, then applying inverse DCT.

Keywords: PIFS, DCT, LSB, PSNR, IDCT, Steganography.

الملخص:

نقترح في هذا البحث خوارزمية لإخفاء البصمة في الصورة الشخصية التي يمكن استخدامها للتحقق من الشخصية عندما نفشل في التعرف على الشخص من خلال صورته. وتتلخص هذه الخوارزمية بضغط صورة البصمة قبل إخفائه اباستخدام(PIFS) لنحصل على مجموعة من التحويلات التي تستخدم لاحقا لتخفى في الصورة الشخصية التي تقسم باستخدام متحول (DCT) حينها تخفى التحويلات بشكل عشوائي ضمن معاملات (DCT) بعدها يطبق عليها معكوس التحول (IDCT) للحصول على الصورة النهائية. التجارب المطبقة اثبتت قوة الطريقة المقترحة.

1. Introduction

Steganography is the art of information hiding [Johnson 2000]. An original image (cover-image) is changed by embedding secret information; the new image is calledstego-image. The modification should be clear in order to get perfectly secret communication. Here, perfect secrecy requires that in the stego-image, no detectable artifacts due to information embedding may be found to distinguish a stego-image from a valid cover-image. This implies that the stego-image should not move away much from a given original cover-image according to anappropriate distortion measure. Image steganographic techniques include least significant bit (LSB) embedding in spatial domain and discrete cosine transform (DCT) coefficients [Johnson 2000]. The goal of steganalysis is to defeat steganography methods by identifying the existence of hidden information. This may be done using detection methods if the distributions of the cover-image and stego-image are known to the steganalyzer [Cachin 1998], and various creative techniques otherwise [Fridrich 2002].

2. Partitioned Iterated Function Systems (PIFS)

According to [Fisher1995], Suppose we are dealing with a 256 x 256 pixel image in which each pixel can be one of 256 levels of grey (ranging from black to white) . Let R_1, R_2,..., R_{1024} be the 8x8 pixel nonoverlapping sub-squares of the image, and let D be the collection of all 16 x 16 pixel (overlapping) sub-squares of the imageFigure 1. The collection D contains 241 • 241 = 58,081 squares. For each R, search through all of D to find a D_i € D which minimizes (Equation 1); that is, find the part of the image that mostly looks like the image above R_i. This domain is said to cover the range. Also, a square in D has 4 times as many pixels as an R_i, so we must either subsample (choose 1 from each 2x2 sub-square of D_i) or average the 2 x 2 sub-squares corresponding to each pixel of R_i, when we minimize (Equation 1). Minimizing (Equation 1) means two things. First, it means finding a good choice for D_i (that is the part of the image that mostly looks like the image above R_i,). Second, it means finding good contrast and brightness settings s_i and o_i for w_i. For each D €**D**, we can compute s_i and o_i using least squares regression, which also gives a resulting root mean square (RMS) difference. We then pick as D_i the D € **D** with the least RMS difference. A choice of D_i, along with a corresponding s_i and o_i, determines a map w_i, of the form of (Equation 2). Once, we have the collection $w_1 \ldots . w_{1024}$ we can decode the image byestimating *xw*. Figure 1 shows four images: an initial image f_0 chosen to show texture; thefirst iteration $W(f_0)$, which shows some of the texture. The result is surprisingly good, given the naive nature of the encoding algorithm. Figure 1 shows how detail is added at each iteration. The first iteration contains detail at size 8x8, the next at size 4x4, and so on. [Jacobs 1990] originally encoded images with fewer grey levels using a method similar to this example but with two sizes of ranges. In order to reduce the number of domains searched, he also classified the ranges and domains by their edge (or lack of edge) properties. This is very similar to the scheme used by Boss et al. [Jacobs 1990] to encode contours.

$$d_{rms}(f \cap (R_i \times I), w_i(f)) \quad i = 1, \ldots, N. \qquad (1)$$

$$w_i \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} a_i & b_i & 0 \\ c_i & d_i & 0 \\ 0 & 0 & s_i \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} e_i \\ f_i \\ o_i \end{bmatrix}$$

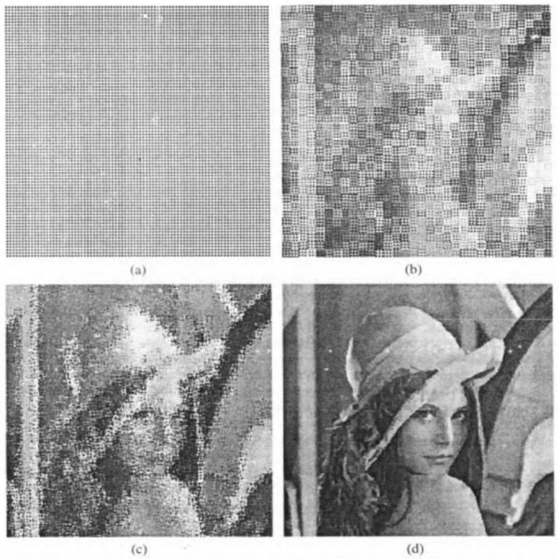


Figure 1: The initial image (a). and the first (b). second (c). and tenth (d) iteration at theencoding transformations

3. Discrete Cosine Transform (DCT)

There aremany techniques used to transformimage fromspatial domain to frequency domain and lossy image compression can be notion of an application of such transform coding. The most common frequency domain methods used in image processing are the 2D-DCT andWavelet [Lenti 2002] [Kharrazi 2006] [Morkel 2005]. In this work, the DCTas an example of the transform coding technique which can beused.

The DCT helps divide the image into parts of differingimportance. Inpractical, DCT can be carried out by partitioningthe image into equally size 2D blocks i.e., N × N grids(e.g., 8 × 8 grid containing 64 pixels per grid). With eachgrid a DCT coefficient for every component in the pixel iscalculated. The formula used to calculate the DCT coefficient S(u, v) (for u, v = 0, 1, 2, . . .,N − 1) of an image gridof pixels F(x, y) is given in Equation 3 [ITU 1992] [Provos 2003]:

$$S(u,v) = \frac{2}{N}C(u)C(v)\left[\sum_{x=0}^{N-1}\sum_{y=0}^{N-1} F(x,y) * cos\left(\frac{\pi u(2x+1)}{2N}\right)cos\left(\frac{\pi v(2y+1)}{2N}\right)\right] \quad (3)$$

where $C(k) = \frac{1}{\sqrt{2}}$, when $k = 0$; otherwise $C(k) = 1$,

and each F(x, y) pixel value has a level range from0 to 255in 8 bits monochromic image. It should be noted that formost images much of signal energy lies at low frequencies;these appear in the upper left corner of the grid of DCTcoefficients. Note that since these techniques modify onlynonzero DCT coefficients, message lengths are defined withrespect to the number of nonzero DCT coefficients in theimages [Kharrazi 2006].

To reproduce a grid of image pixels F(x, y), (for x, y =0, 1, 2 . . .N − 1), from the grid of DCT coefficients S(u,v), we can use the inverse of the DCT formula given in Equation 4:

$$F(x,y) = \frac{2}{N}\left[\sum_{u=0}^{N-1}\sum_{v=0}^{N-1} C(u)C(v)S(u,v) * cos\left(\frac{\pi u(2x+1)}{2N}\right)cos\left(\frac{\pi v(2y+1)}{2N}\right)\right] \quad (4)$$

4. Why Use PIFS

PIFS method be effective in image compression when there are similarities in the image then gives the best compression. Fingerprint images have frequently feature similarities because they have similar curves for this reason be PIFS method is useful and provides the greatest possible compression rate of other compression methods.(See Figure 2).

Figure 2: Similarity in fingerprint image

5. Proposed Hiding Fingerprint Method

Proposed method introduces a new method of embedding fingerprint within personal image.We applied a combination of PIFS, DCT transform and the notion of LSB technique of spatial domain steganography. The main idea of this method is to utilize significant bit of the DCT coefficients of a cover image to hide image bits. This method compresses the hide image (fingerprint) and modifies the bit of the coefficients randomly. The effect of this variation is distributed across the image by using the inverse of the discrete cosine transform (IDCT). This approach is illustrated in details in the following steps (algorithm):

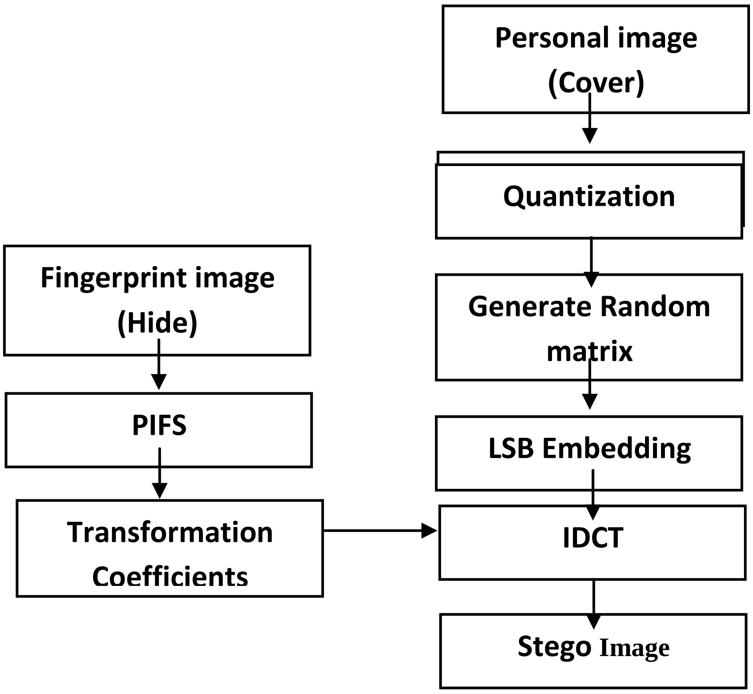
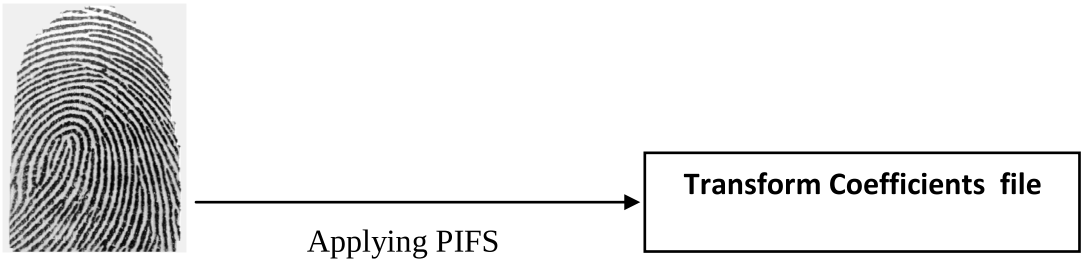


Figure 3: Block Diagram of Proposed Embedding Technique

Step 1: Applying PIFS on fingerprint image (hide).

Herein the image is compressed by applying PIFS on it to get small image in percentage of 20% of original image size.



a. Size: 40.4 KBb. Size: 6.7 KB

Figure 4: Show size of image **a**.before compression**b**.after compression

Step 2: Applying 2D DCT on personal image (Cover).

Herein the image is partitioned into 2D 8 × 8 blocks. Thus, each block consists of 64 values then each blocks transform by using Equation (3) to get DCT coefficients.

Step 3: Perform Quantization.

Step 4: Generating 8 × 8 (64 items) matrix randomly.

Using this matrix to distributed hide image bit on non-zero DCT coefficients. As example let R matrix is random matrix and F is DCT coefficients.

$$R = \begin{bmatrix} 25 & 51 & 12 & 46 & 15 & 35 & 42 & 57 \\ 2 & 18 & 53 & 31 & 29 & 5 & 26 & 59 \\ 1 & 45 & 16 & 64 & 20 & 8 & 33 & 22 \\ 62 & 49 & 4 & 27 & 44 & 7 & 55 & 11 \\ 43 & 37 & 40 & 61 & 14 & 50 & 39 & 9 \\ 23 & 58 & 36 & 24 & 52 & 41 & 63 & 32 \\ 10 & 6 & 38 & 48 & 30 & 21 & 34 & 47 \\ 3 & 54 & 17 & 13 & 28 & 19 & 56 & 60 \end{bmatrix} \quad F = \begin{bmatrix} 481 & -24 & 0 & -12 & 0 & 0 & 0 & 0 \\ -120 & 9 & -18 & -11 & 0 & 0 & 0 & 0 \\ 32 & 90 & -20 & -24 & 0 & 0 & 1 & 0 \\ 12 & 88 & 12 & 0 & 0 & 0 & 0 & 0 \\ -51 & 45 & 19 & 0 & 0 & 0 & 0 & 0 \\ -64 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -24 & 24 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Here, the first item in R matrix (25) that is meaning the next bit of hid image is embedding in DCT coefficient that have location (25) in F which is (12). In the same way for all DCT coefficients F.

Step 5: Embedding transform coefficients bits.

In this step, transform coefficients bits are embedded one by one in the successive non-zero DCT coefficients of the low frequency region upper left corner of the block of the DCT coefficients. If the value of the first bit of personal image and the PIFS bit are equal, nothing should be made. Otherwise, the first bit should be replaced by a PIFS bit.

Step 6: Apply the IDCT.

Now, we apply Equation 4 (i.e., the inverse of DCT) on the stego-matrix generated by Step 5. The result of this process will be stego-image.

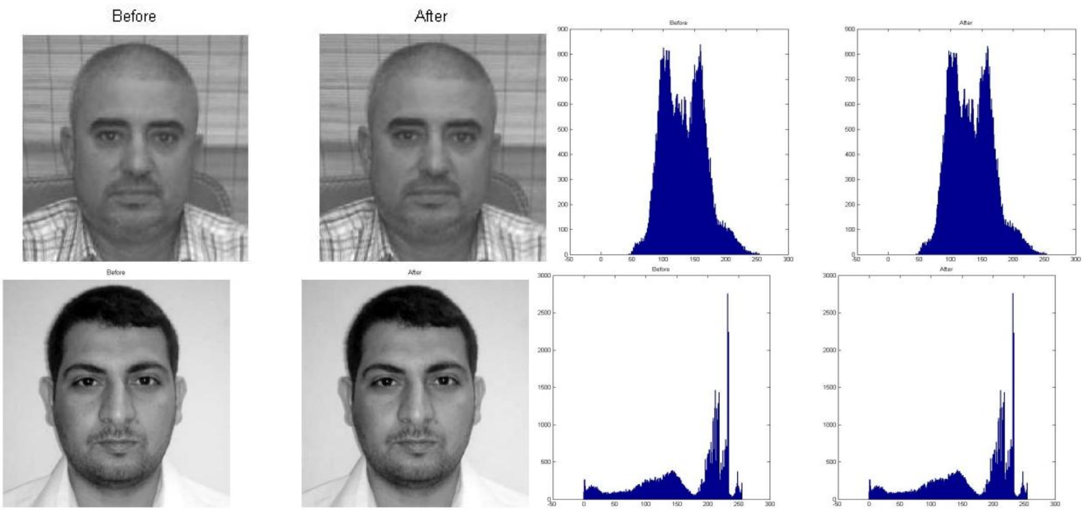
6. Experimental Results

Using the peak signal to noise rate (PSNR) as a measure to compute the quality of the stego image. In order to minimize the visible effect of changes to pixel values, the value of PSNR of stego image must be as high as possible.The five personal images and its fingerprint images in Table 1. they are used in simulation proposed method.

Table 1 :Person's Image and fingerprint

N.	Person1	Person2	Person3	Person4	Person5
Image					
fingerprint					

In Figure 4 shows the compare between image before and after steganography and its histogram. FromFigure 4 cannot observation the change between original image and stego-image that prove the effective of our algorithm. The obtained results of the experiments are summarized in the Table 2 by PSNR. Recall that the tested techniques are the proposed technique (PIFS with DCT), LSB, and LSB with DCT.Table 2 shows some of the obtained results: the PSNR of the different image that hide a PIFS. However, the table shows more precisely the decreasing of the PSNRs of stego images as the size of the embedded message increases. From the tables, we can see that all of the tested techniques produce acceptable reconstruction of the covering image.



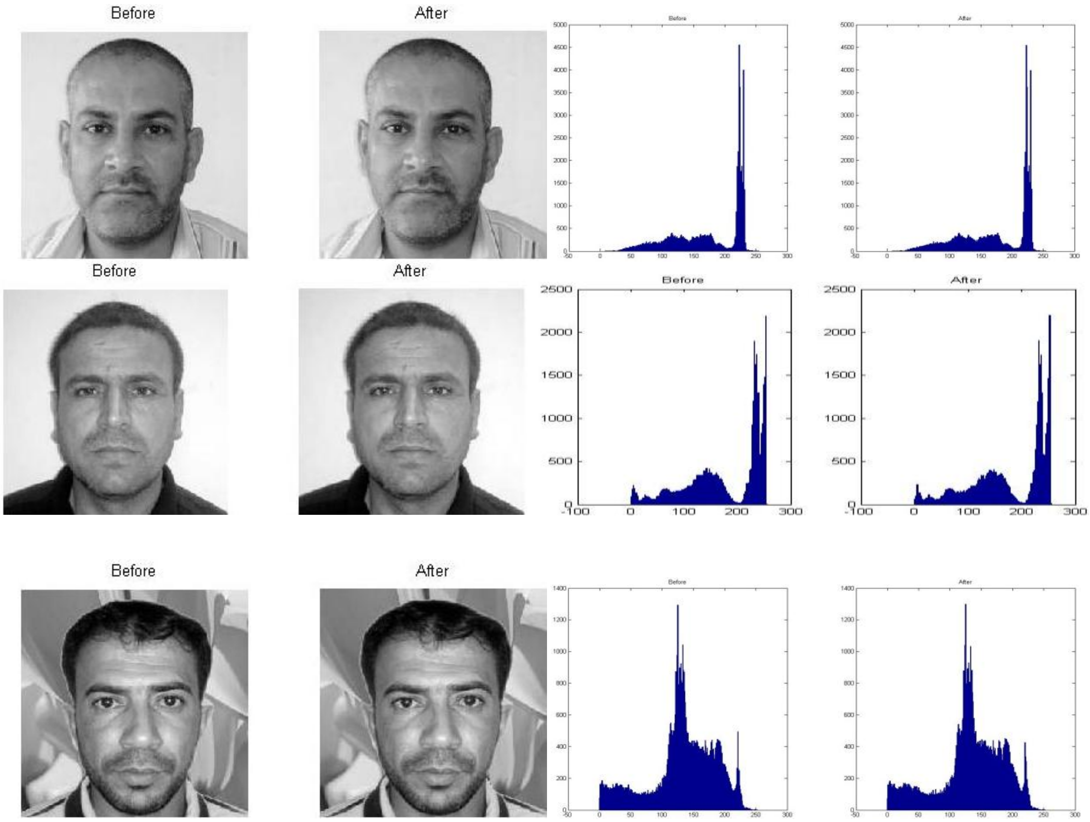


Figure 4: Shows the compare between image before and after steganography

Table 2: **PSNR(dB)** for LSB, LSB-DCT and PIFS-DCT

Persons	LSB	LSB-DCT	PIFS-DCT
Person 1	60.1084	68.8952	69.2114
Person 2	59.1058	68.0819	69.1023
Person 3	59.3278	68.9690	69.1106
Person 4	59.3731	68.0904	69.2029
Person 5	59.9903	68.8981	69.1272

From above table can see our proposed algorithm PIFS-DCT has high PSNR because it decreasing size of hide message by using PIFS compression, therefore the increase the PSNR.

7. Conclusions

In this paper, we suggest a hybrid approach that applies the PIFS with the DCT and LSB techniques. The idea is to utilize a LSB of the DCT coefficients of a cover image to hide message bits. After that, the information and the variation of the coefficients, affected by the embedding process, are random spread in the stego image by utilizing the inverse of the DCT process. The obtained experimental results show that, the proposed method will be a good and acceptable steganogaphymethod. Also, by imbedding information in the main significant bits of the DCT domain, the hidden message resides in more robust areas, spread across the entire stego image, and provides better resistance against stiganalysis process than other techniques.

8. References

Cachin, C., “An information-theoretic model for steganography”, Springer, 1998.

Fisher,Y., "Fractal Image Compression. Theory and Application", Springer-Verlag, New York, 1995.

Fridrich, J. and (Goljan M.), “Practical Steganalysis of Digital Images-state of the Art”, Pmc. SPIE Photonics West, San Jose, California, Jan. 2002.

ITU, “Information Technology - Digital Compression and Coding of Continuous-Tone Still Images Requirements and Specifications Recommendation T.81”, ITU Sept., 1992.

Jacobs, E.W., (Boss,R.D.,andFisher, Y.), "Fractal-based image compression ii", Technical Report 1362, Naval Ocean Systems Center, San Diego, CA, June 1990.

Johnson, N. F. and (Katzenheisser, S.), “A Survey of Steganographic Techniques”, Informotion Hiding, Norwood, MA. 2000.

Kharrazi, M., (Sencar, H. and Memon N.), “Performance Study of Common Image Steganography and Steganalysis Techniques,” Communications of the SPIE and IS&T, Oct-Dec., 2006.

Lenti, J., “Steganographic Methods", periodic polytechnic ser.el.eng 44, No.3-4 Jun 2002.

Morkel, T., (Eloff, J., and Olivier, M.), “An overview of image steganography”, In Proceedings of the Fifth Annual Information Security South Africa Conference, Sandton, South Africa, 2005.

Provos, N., and (Honeyman P.), “Hide and Seek: An Introduction to Steganography”, Security & Privacy, IEEE, May-June 2003.