IOSR Journal of Engineering
May. 2012, Vol. 2(5) pp: 1007-1012

Concealing Encrypted Iris Templates in Images using Quantized DCT Coefficients

Rita Chhikara[1] Sunil Kumar[2]
[1]*ITM University, Gurgaon, Haryana, India*
[2]*Maharaja Agrasen Institute of Technology, Rohini, Delhi 110086, India*

***Abstract*—**This paper introduces a novel steganography-based approach to protect the iris data by hiding it into a digital image for personal identification purpose. Transformations are done to enrypt the biometric data before hiding. JPEG quantization blocks in diagonal sequence are investigated for exploring an efficient hiding algorithm with high transparency and strong robustness. The extraction of iris data does not require original image and provide high accuracy under different attacks. Experimental results show that this proposed method improves the security of the iris-feature with hardly detectable decrease in recognition performance.

Keywords- Biometrics, iris recognition, steganography, DCT, quantization.

I. Introduction

A biometric system provides automatic identification of an individual based on a unique feature or characteristic possessed by the individual. Traditional token-based or knowledge-based personal identification techniques (such as identification cards, passwords, etc) are unable to differentiate between an authorized person and an impostor who fraudulently acquires the access privilege of an authorized person[1]. Since biometrics characteristics (e.g. face, voice, fingerprint, gait, hand geometry, iris, gene, etc.) are inherently associated with a particular individual, they are uneasy to be stolen, forgotten, lost or attacked. However, the problem of security and integrity of the biometrics data in networked environments poses new issues [2]. Steganography is the ***art of concealing the existence of information within seemingly innocuous carriers***. It can be viewed as akin to cryptography. Both have been used throughout recorded history as means to protect information At times these two technologies seem to converge while the objectives of the two differ. A key application area of image embedding is in hiding vital medical or biometric information of employees in their pictures for ready access in case of an emergency, or for secure identification [3]. In these applications, biometrics-based identifying information, for example, may be hidden in the picture card of a person and the claimed identity of the card carrier can be verified by retrieving the hidden data and comparing them with the biometric data collected on the spot.

In the past few years, several researchers have made attempts on biometric data protection with the help of data hiding techniques. Pankanti and Yeung [4] proposed a fragile watermarking method for fingerprint image tampering detection. In a series of works [5-7], Jain and Uludag introduced several methods for combining data hiding with biometrics. In [5], Jain et al. presented a fingerprint image watermarking method that can embed facial information into host fingerprint images. Jing Dong and Tieniu Tan in [9] have studied the effects on iris recognition performance of iris watermarking. Using iris template as watermark ensures the iris template transmission while watermarking iris images could help to protect the database ownership as well as to detect iris image tempering. However, most of these works are on biometric data watermarking and involve the use of face and fingerprint images. In fact, biometrics data is bare to others when it is transmitted in the network, and if the hacker gets the data on the network, the security of the biometrics system is lost. So it is necessary to keep the security of the biometrics data when it transmitting in the suspicious channel. Because of the advantage of steganography, it can be used in the biometrics system for the security of the biometrics transmission.

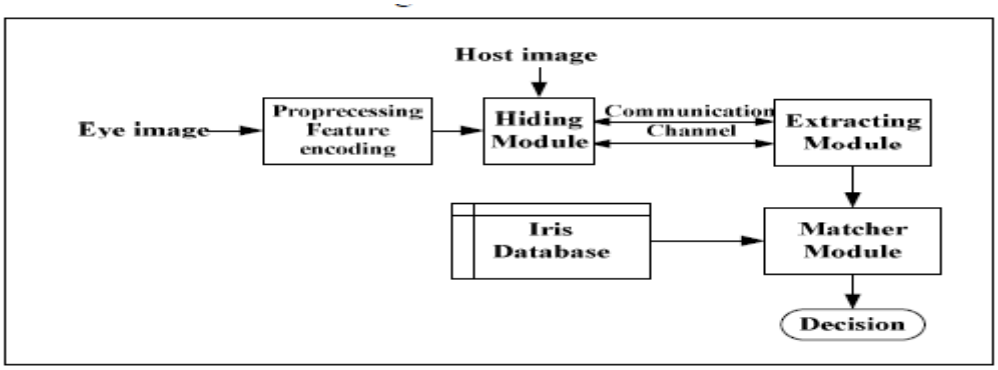

Fig 1 *Biometric system combined with steganography.*

In this paper, we attempt to present a new steganography based method based on LSB, JPEG standard quantization table in DCT domain for the purpose of protecting biometric data. Biometric image is captured from the sensor and image processing algorithms are performed to extract the features. These data are unique for different person and converted into a binary stream to generate the iris code. The hiding and extracting module separately implement the two steps as shown in Fig 1, hiding and extracting iris information. In the matching phase, the matcher module evaluate the scores by matching the data acquired with the database.

2. Feature Extraction

For feature extraction Libor Masek[14] method was followed with a gray scale image. It has an automatic segmentation algorithm, which would localise the iris region from an eye image and isolate eyelid, eyelash and reflection areas. Automatic segmentation was achieved through the use of the circular Hough transform for localising the iris and pupil regions, and the linear Hough transform for localising occluding eyelids. Thresholding was also employed for isolating eyelashes and reflections.
Next, the segmented iris region was normalised to eliminate dimensional inconsistencies between iris regions. This was achieved by implementing a version of Daugman's[11] rubber sheet model, where the iris is modelled as a flexible rubber sheet, which is unwrapped into a rectangular block with constant polar dimensions.
Finally, features of the iris were encoded by convolving the normalised iris region with 1D Log-Gabor filters and phase quantising the output in order to produce a bit-wise biometric template.

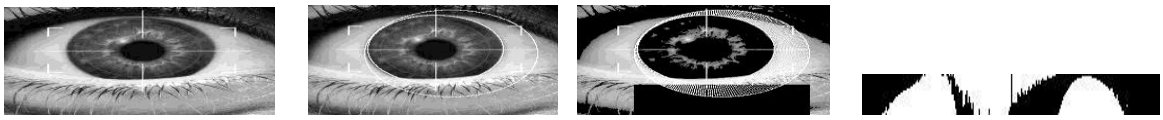

Fig 2 *Feature Extraction of an Iris*

3. Proposed Steganographic Method

The proposed method is a combination of DCT and LSB techniques with quantization using quality factor (α) 50 .

3.1 DCT - A discrete cosine transform (DCT) expresses a sequence of finitely many data points in terms of a sum of cosine functions oscillating at different frequencies. For each color component, the JPEG image format uses a *discrete cosine transform* to transform successive 8 x 8 pixel blocks of the image into 64 DCT coefficients each. The DCT coefficients *F(u,v)* of an 8 x 8 block of image pixels *f(x, y)* are given by

$$F(u,v) = \frac{1}{4} C(u)C(v) \left[\sum_{x=0}^{7} \sum_{y=0}^{7} f(x,y) \cos\frac{(2x+1)u\pi}{16} \cos\frac{(2x+1)v\pi}{16} \right] \quad (1)$$

$$C(u) = \begin{cases} \frac{1}{\sqrt{2}}, & \text{if } u \leq 0 \\ 1, & \text{if } u > 0 \end{cases} \quad (2)$$

The DCT block F consists of 64 DCT coefficients. The top-left coefficient F(1,1) correlates to lower frequency of the original image block, which is called DC coefficient. The coefficients away from F(1,1) in all directions correlates to higher and higher frequencies, where F(8,8) corresponds to the highest frequency.

3.2 Quantization - A sample 8 x 8 block of DCT coefficients is compressed by quantization. A useful feature in JPEG process is that image compression and quality is obtainable by selection of specific quantization table. Scalar multiples of JPEG standard quantization matrix may be used for compression. The scaled quantization matrix is then rounded and clipped to have positive integer values ranging from 1 to 255. For a quantity level

greater than 50, less compression and high image quality is obtained. For a quantity level less than 50, more compression and low image quality is obtained. The standard quantization matrix JPEG uses is quality factor (α) 50 that is shown in Fig. 1Q(u,v)=

16	11	10	16	24	40	51	61
12	12	14	19	26	5	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Fig 3. *Quantization matrix*

3.3 LSB - In the LSB(least significant bit) approach, the basic idea is to replace the Least Significant Bits (LSB) of the cover image with the bits of encrypted message to be hidden without destroying the statistical property of the cover image significantly. The LSB-based technique is the most challenging one as it is difficult to differentiate between the cover-object and stego-object if few LSB bits of the cover object are replaced. Least Significant Bit(LSB) Embedding Digital images are mainly of two types (i) 24 bit images and (ii) 8 bit images. In 24 bit images we can embed three bits of information in each pixel, one in each LSB position of the three eight bit values of RGB. Increasing or decreasing the value by changing the LSB does not change the appearance of the image.

3.4Algorithm

Problem definition: Given a cover image C and the message to be embedded M that is template for irisicode.
The objective is:

(i) To transform the stego-object from spatial domain to frequency domain using DCT.

(ii) To compress the frequency domain stego-object using Quantization matrix as shown in Fig 1 to generate a secure stego object.

(iii) To embed the encrypted message in the cover image by replacing LSB bits between 1 and 5. The combined image is called stego-image(S).

A. Hiding Phase

Step1: Encrypt the iris code

The message is encrypted by

i) Generating template for iris in binary form.

ii) Then rotating the binary bits by 90 degree

iii) Then flipping the bits upside down to ensure more security.

This is implemented using rot90 and flipud functions of matlab.

Step2: DCT and Quantization of image

Divide the host image into non-overlapping 8×8 blocks. The blocks are thereafter quantized using standard matrix Q(u) given in Figure 1. The coefficient matrix for the kth block is generated starting from 1st block and moving downwards diagonally. Thereafter the diagonal block in left most corner is considered and moving upwards. Check is implemented to ensure that a block is not considered again for embedding data.

For example

i) If i=1 and j=8

ii) Block(i:j.i:j) is selected for embedding

iii) i=i+8; j=j+8

iv) go to step ii until j < =width of image

The shaded region shows 8x8 blocks DCT coefficients selected for embedding in the sequence denoted by number on each block. The overlapping of block takes place if number of blocks are odd. In that situation only one of the blocks are taken into consideration.

1							16
	2					15	
		3			14		
			4	13			
			12	5			
		11			6		
	10					7	
9							8

Fig4*: Diagonal selection of blocks for embedding.*

Step 3: Embedding the secret code using LSB technique
i) The absolute value of selected pixel is selected.
ii) A flag is set if it is a negative number.
ii) It is converted to binary
iii) First bit of message to be hidden is written over LSB of this pixel
iv) It is converted back to decimal
v) It will change by +1 or -1
vi)The above procedure is repeated till all the bits are hidden in the image.

Step4: Reconstruct each block by inverse DCT. Then compose those new blocks and obtain a new approximate host image containing secret information.

B. Extracting Phase

Step1: Divide the received image into 8×8 blocks and transform them into DCT coefficient matrix. Quantize the blocks using standard quantization matrix shown in Fig 1.
Step2: Select the blocks in diagonal pattern as shown in Fig 2.
Step3: Select the right coefficients to extract the secret information. Convert the selected pixel to binary and obtain the LSB. Continue this process until all the bits of secret message have been extracted in a one dimensional sequence..
Step4: Get 2-D sequence of binary forrn with the related information stored in the image in predetermined pixels of the image.
Step5: Decrypt by flipping the acquired 2-D matrix upside down and thereafter rotating by 90 degrees to acquire the real iris code.

4 Performance Analysis

The binary coded template generated as described in section 2 is taken as the secret message to be embedded in the cover image as shown in the figure 5. The binary bits are encrypted by flipping upside down and then rotating by 90 degrees.
We consider the cover image as shown in Figure 5(a) for all the experiments and analysis performed in this paper. Message of bits 960 is taken to hide in the image.

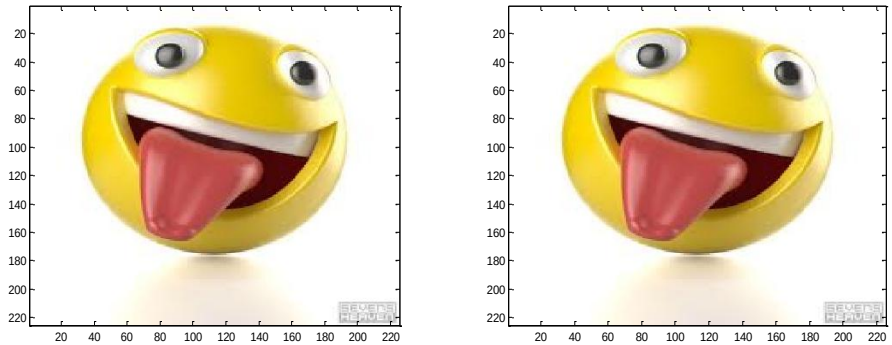

Figure 5. a) The cover image and b) stego-images

The cover image is converted into blocks of 8x8 DCT coefficients. The Discrete coefficient values F(u,v) of red color pixels of the image are as shown in the table below.

Table 1. 8x8 DCT coefficient of red color of the imageF(u,v)=

40678	-15.024	11911	-1440	1281.8	2776.7	68.789	-554.8
-3360.2	-94.017	2042.5	-1228.7	1043.4	1782.3	345.82	-520.46
8924.7	592.66	-4083.3	509.83	-4427.1	-1041.3	265.01	-604.48
1568.1	307.15	1420.7	403.79	-2671.3	-1552.1	-838.05	-79.646
558.74	-125.26	-2456.7	995.84	2249.3	-1368	96.921	851.29
2743.8	-726.73	-3003.1	801.63	116.78	-820.02	1871.5	801.29
453.12	51.467	-616.63	-199.42	316.46	585.83	-640.05	-38.559

Quantization is achieved by dividing each element in the DCT coefficient block by the corresponding value in the quantization matrix, and the result is rounded to the nearest integer. The quantized DCT coefficients FQ(u ,v) are computed by

$$FQ(u,v)=round(F(u,v)/Q(u,v)) \quad (3)$$

Table 2. Quantization table

2542	0	1191	-90	53	69	1	-9
-280	-9	146	-65	40	31	6	-9
637	46	-255	21	-111	-19	4	-10
112	18	65	14	-53	-19	-11	-1
31	-7	-67	18	33	-13	1	11
114	-20	-54	13	1	-8	17	9
9	1	-9	-2	3	5	-5	0
16	1	-14	-6	13	6	-5	3

The absolute values are selected from quantized matrix to hide the data using LSB technique. To extract the secret information from the selected DCT coefficient (i,j), Quantize the block and select pixels and take their absolute value , convert to binary and pick the least significant bit. The size of the hidden text is also hidden in the image which is extracted first and then the number of bits of hidden message is extracted. The qequnatized image is generated back by the formula

$$D(u,v)=FQ(u,v)*Q(u,v) \quad (4)$$

Table 3. Dequantized Matrix

40672	0	11910	-1440	1272	2760	51	-549
-3360	-108	2044	-1235	1040	1798	360	-495
8918	598	-4080	504	-4440	-1083	276	-560
1568	306	1430	406	-2703	-1653	-880	-62
558	-154	-2479	1008	2244	-1417	103	847
2736	-700	-2970	832	81	-832	1921	828
441	64	-702	-174	309	605	-600	0
1152	92	-1330	-588	1456	600	-515	297

Finally inverse DCT is applied to generate the stego image. As seen by the Fig 5 the two images appear to be similar and change is not noticeable to human eye. This proposed method provides high information hiding

capacity, increases the security and retains the image quality. The reverse procedure at receiver end is employed to extract the iris template and thereafter match the code with original iris.

5 Conclusion

With the development of biometric recognition techniques and the wide spread utilization of biometric identification systems, the security of biometric data itself has became a crucial issue because various risks have been discovered while using biometric system. In this paper, a new steganographic scheme is proposed which aims at reinforcing the security of iris feature. The experimental results show that it enhances the security of iris data greatly when transmitting in the communication channel while degrade little the recognition performance. Moreover, it is highly robust against different kinds of attacks and demonstrates good recognition performance.

6 References

[1] A. K. Jain, S. Pankanti, and R. Bolle(eds.), "BIOMETRICS: Personal Identification in Networked Society," Kluwer, 1999.
[2] B. Schneier, "The uses and abuses of biometrics," Comm.ACM, vol.42,no.8, pp.136, Aug.1999.
[3] Niels Provos, Peter Honeyman, "Hide and Seek: An Introduction to Steganography," IEEE computer society, 2003, pp. 32-44.
[4] S.Pankanti and M.M.Yeung, "Verification Watermarks on Fingerprint Recognition and Retrieval", Proc.SPIE, vol.3657, pp.66-78, 1999.
[5] A.K.Jain, U.Uludag, and R.K.Hsu, "Hiding a face in a fingerprint image," Proc. of Intl Conf. on Pattern Recognition, vol.3, pp. 756-759, Aug. 2002.
[6] A. K. Jain, A. Ross, and U. Uludag, "Biometric template security: challenges and solutions," Proc. of the European Signal Processing Conference (EUSIPCO '05), Sep.2005
[7] A.K.Jain and U.Uludag,"Hiding biometric data," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 25, no. 11, pp. 1494-1498, Nov. 2003.
[8] Fridrich. J., Goljan. M., and Du. R: "Steganalysis Based on JPEG Compatibility" Proc. SPIE Multimedia Systems and Applications IV, Vol. 4518, Colorado, 2001, pp. 275-280.
[9] K B Raja, Venugopal K R and L M Patnaik, "A Secure Stegonographic Algorithm using LSB, DCT and Image Compression on Raw Images", Technical Report, Department of Computer Science and Engineering, University Visvesvaraya College of Engineering, Bangalore University, December 2004.
[10] Hsien-Wen Tseng, Chin-Chen Chang, "Steganography using JPEG Compressed Images", Proceedings of IEEE Fourth International conference on computer and information technology, 2004,pp. 12-17.
[11] J. Daugman. "High Confidence Visual Recognition of Persons by a Test of Statistical Independence", IEEE Tans. Pattern Analysis and Machine Intelligence, vol.15, pp.1148-1161,1993.
[12] Y. Z. Li, H. Zhu, R. Q. Yu, G. Yang, and J. Xu, "An Adaptive Blind Watermarking Algorithm Based on DCT and Modified Watson's Visual Model," Proc. IEEE Symp. Electronic Commerce and Security, pp.904- 907, Aug. 2008.
[13] Institute of Automation, Chinese Academy of Sciences. CASIA Iris Image Database(ver.1.0), http://www.sinobiometrics.com
[14] http://www.csse.uwa.edu.au/~masekl01/