International Journal of Electronics Communication and Computer Engineering
Volume 3, Issue 6, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

Application of Biometric Image as a Key in Ensuring Security of Data using Steganographic Approach

Sabyasachi Samanta
Haldia Institute of Technology
Haldia, WB, INDIA
sabyasachi.smnt@gmail.com

Saurabh Dutta
Dr. B. C. Roy Eng. College
Durgapur, WB, INDIA
saurabh.dutta@bcrec.org

Gautam Sanyal
National Institute of Technology,
Durgapur, WB, INDIA,
nitgsanyal@gmail.com

***Abstract* – In this paper, we have proposed a new image steganographic technique capable of producing a secret-embedded image that is entirely indistinguishable from the original image by the human eye. Data bits from textual message are embedded about the entire image to some suitable pixel positions. Initially the root pixel position is selected depending on the key value and the remaining is considered as the child nodes are calculated in binary tree. As a result, we get a watermarked image which is almost imperceptible to human survey. Subsequently we have taken a biometric image as a key image. Then we have done the bitwise XOR operation pixel by pixel by using both the watermarked image and key image. Thus, we get a visible watermarked image. At the decryption end, again the hidden data bits are retrieved from the watermarked image.**

Keywords – Watermarking, Information hiding, Binary tree, Biometric authentication.

I. INTRODUCTION

Digital Watermarking describes the way or technology by which anybody can hide information, for example a number or text, in digital media, such as images, video or audio. A binary tree is a tree-like structure that is rooted and in which each vertex has at most two children and each child of a vertex is designated as its left or right child. Biometrics or biometric authentication refers to the identification of humans by their characteristics or traits. Biometrics is used as a form of identification and access control. It is also used to identify individuals in groups that are under surveillance. A pixel with 32-bit color depth consists of α value, R (Red), G (Green) and B (Blue) value. α value is the value of opacity. If α is 00000000, the image will be fully transparent. Each of three(R, G & B) 8-bit blocks can range from 00000000 to 11111111(0 to 255) [6] [17] [19].

In this paper, we have proposed a technique to embed a message about the entire image at arbitrary pixel positions. The text taken from the keyboard or special characters encoded into its ASCII-8 (American Standard Code for Information Interchange) binary equivalent. The length of the text is also converted into its 8-bit binary equivalent. The data bits from length and text are stored into an encrypted array. Then the number of required pixel positions i.e. where to embed data, is calculated from the array length. The initial or root pixel [k(x, y)] position is generated from the six digit key value. From the root pixel position the embedding process become initiated. The remaining pixel positions are calculated as the left and right child are being calculated in binary tree. The left child is calculated as 2*k in binary tree representation. Here the left child (pixel position) is as [k(2*x,2*y)]. The right child (pixel position) is as [k(((2*x)+1),((2*y)+1))]. If nay value become more than the width or height of the image, the modulus operation is being calculated. The root pixel position is notified as 1. The left and right child are as 2(2*1) and 3(2*1+1) respectively. The process continues up to required number of pixel position vertically from root to higher level and horizontally left to node. The data from encrypted array is being embedded from left child to right child in each and every level starting from root. Moreover, we get a watermarked image.

Then we have taken a biometric image as a key image. Here we have taken a fingerprint image as a key image. The bitwise XOR operation (pixel by pixel) between two images is initiated from the initial pixel position of both images. The XOR operation may vary from process to process depending on the size of images. Here we have taken the equal size images. Finally, we get a visible watermarked image.

In our work, we have targeted any one bit of last four significant bit of each R, G and B of any selected random pixel positions. Here the bit position is selected depending on the value of pixel position. The replacement of all the bits have done in nonlinear pixel and bit positions, in any one of last four significant bit of R,G and B at selected pixels about the entire image using the private key cryptography technique taking the α value as 255 or as in the original image [1] [4] [6] [18] [21].

Example: A text with 9-characters will form an array with 80 (8+9*8) bits of stream (first 8-bits for length). An image with 560 X 864 dimension has 4, 83,840 pixels. In our work, only we have altered any one bit of last four significant bits. If any bit generated from text become same to the targeted bit of image then there will be no change i.e. it will produce the same to original image.
Section 2 represents the related work. Section 3 represents the scheme followed in the encryption technique. Section 4 represents an implementation of the technique. Section 5 gives you an idea about the experimental results. Section 6 is an analytical discussion on the technique. Section 7 draws a conclusion.

II. RELATED WORKS

In this section we have discussed various steganographic data hiding methods both in spatial domain and transform domain.

International Journal of Electronics Communication and Computer Engineering
Volume 3, Issue 6, ISSN (Online): 2249–071X, ISSN (Print): 2278–4209

A. Spatial Domain Steganographic Method

1) Data hiding by using LSB method: Various techniques about data hiding have been proposed so far. One of the common techniques is based on manipulating the least-significant-bit (LSB) method. Using this method the message bits are directly replaced with the LSBs of the cover-image [7] [8].

2) Data hiding by using PVD method: Wu and Tsai proposed a novel steganography technique using the pixel-value differencing (PVD) method to distinguish edge and smooth areas of the cover image. The PVD technique can embed more data in the edge area which guarantees high imperceptibility. Later on Chang et al. proposes a new method using tri-way pixel value differencing which is better than original PVD method with respect to the embedding capacity and PSNR [9] [10].

3) Data Hiding by GLM: In 2004, Potdar et al. proposes GLM (Gray level modification) technique which is used to provide a new platform for grey level modification steganography system. It embeds binary data within the spatial domain of the grey scale images by modifying the grey level values of the pixels. It is a one-to-one mapping between the binary data (or message bits) and the selected pixels in an image [5] [11] [12].

B. Transform Domain Steganographic Method:

Many transform domain methods exist. The commonly used transformation functions include Discrete Cosine Transformation (DCT), Fast Fourier Transform (DFT), and Wavelet Transformation.

1) DCT based Data Hiding: DCT method is used in the JPEG compression algorithm. It transforms a signal or image (taking 8x8-pixel blocks of the image) from the spatial domain to the frequency domain. It can separate the image into high, middle and low frequency components. DCT based steganography embed the text message in LSB of the Discrete Cosine (DC) coefficient of digital images. Some DCT based steganographic work has been given in [13] [14] and [20].

2) DWT based Data Hiding: Digital wavelet transform (DWT) represents an image as a sum of wavelet functions with different locations and scales. Unlike the spatial domain, secret messages may be embedded in the high frequency coefficients resulted from Discrete Wavelet Transform i. e. the information is stored in the wavelet coefficients of an image, instead of changing bits of the actual pixels. Some DCT based steganographic work has been given in [15] and [16].

III. THE SCHEME

This section represents a description of the actual scheme used during "Application of Biometric Image as a Key in Ensuring Security of Data using Steganographic Approach" technique. Section 3.1 describes the encryption technique using five algorithms 3.1.1, 3.1.2, 3.1.3 & 3.1.4 while section 3.2 describes the decryption technique using algorithm 3.2.1 [2] [3] [6].

3.1 Encryption of data bits about the image

3.1.1 Create an array from message data

Step I: Take input from keyboard or special characters and compute the length (chlen).

Step II: Convert the length (chlen) into its 8 bit binary equivalent. Store that data bits to earr[bit] as LSB (Least Significant Bit) to earr[1] and MSB (Most Significant Bit) to earr[8] respectively.

Step III: Convert each character to 8-bit (using ASCII-8) binary equivalent and store to earr [ ] as LSB to earr[1+(i*8)] and MSB to earr[8+i*8].

Step IV: Repeat Step III for i=0 to (N-1).

Step V: Stop.

3.1.2 Selection of pixel positions using key

Step I: Calculate number of pixels (as three following data bit replaced in R, G and B of every pixel). So p= (ceil (bit /3)).

Step II: Root pixel [k(x, y)] position is generated from the 6-digit key value. The left child (pixel position) is as [k(2*x,2*y)]. The right child (pixel position) is as [k(((2*x)+1),((2*y)+1))]. If nay value become more than the width or height of the image, the modulus operation is being calculated.

Step III: The root pixel position is taken as 1. The left and right child are as 2(2*1) and 3(2*1+1) respectively.

Step IV: Repeat Step II to Step III up to p.

Step V: Stop.

3.1.4 Replacement of array elements with R, G & B values of pixels

Step I: Calculate the width (w) and height (h) of the image.

Step II: Set x=arrx[p] and y=arry[p].

Step III: To select the pixel position into image, compare the value of x and y with the value of w and h (where addressable pixel position is (0, 0) to (w-1, h-1)).

a) If (x >(w-1)) or (y >(h-1)) then
Set P (x, y) = P (0+(x %(w-1)), (0 +(y %(h-1)))
Otherwise Set P (x, y) = (x, y).

Step IV: To select the bit position (b) of selected pixel i.e. with which bit the array data will be replaced. Set z =arrz[p].

i) If (z%4=0) then b=1st LSB.

ii) If (z%4=1) then b=2nd LSB.

iii) If (z%4=2) then b=3rd LSB. Otherwise b=4th LSB of each R, G & B of a pixel.

Step V: To replace the array elements with the selected bit position of selected pixel and to reform as a pixel

a) Alter each of R, G &B to its equivalent 8-bit.

b) Replace data bit of earr[bit] by following Step III to Step V.

c) Reform it's to pixel value and place it to its position of the image.

Step VI: Replace the array element to pixels using the above mentioned process starting from the 0th element up to the end of the array.

A) If bit%3 = 0
Go to Step VII.

B) If bit%3 = 1 for 0th element to (bit-1)th element of the array repeat Step VI (A). For (bit)th element to R, value for G and B will be remain same. And go to Step VII.

C) If bit%3=2 for 0^{th} element to $(bit-2)^{th}$ element of the array repeat Step VI (A).For $(bit-1)^{th}$ element to R, $(bit)^{th}$ to G and B will be remain same. And go to Step VII.

Step VII: Repeat Step II to Step VI for i=1 to p.

Step VIII: Stop.

3.1.4 Creation of visible watermarked image using biometric image

Step I: Take the biometric key image and compute the width (w_b) and height (h_b).

Step II: Take the invisible watermarked image and perform the bitwise XOR operation (pixel by pixel) between two images starting from initial pixel position. We get the visible watermarked image.

Step III: Stop.

3.2 Decryption of the data bits from the image

3.2.1 Regain of replaced bits from the watermarked image

Step I: Take the biometric key image as input and compute the width (w_b) and height (h_b).

Step II: Read the values of R, G and B of each and every pixel of both the key image and visible watermarked image. Then to get the invisible watermarked image go through Step II to Step III of Algorithm 3.1.4.

Step III: Take key input to get the pixel and bit positions in R, G and B of selected pixels, go through Step I to Step VI of Algorithm 3.1.2 and Step I to Step VII Algorithm 3.1.3.

Step IV: Retrieving the encrypted bits from the selected pixels store it to decrypted array from darr[1] to darr[bit] respectively.

Step V: To get the length repeat Step II to Step IV for i= 1 to 3 times (as every pixel contain three data bits).

Step VI: Taking data bits of darr[1] as LSB and darr[8] as MSB calculate the length (chlen) of message.

Step VII: Retrieving the encrypted bits from the selected pixel and bit positions, store it to decrypted character array from darr[10] to darr[bit] respectively. Step VIII: Repeat Step VII for i=4 to chlen.

Step IX: Take data values from the decrypted array darr[ ], LSB as darr[8*i+1] and MSB as darr[8*(i+1))] respectively. Converting the values into its equivalent ASCII characters, restore the message.

Step X: Stop.

VI. AN IMPLEMENTATION

Let the message to be encrypt is "NONLINEAR".

So the length of the message =09

=00001001(8 Bit Binary equivalent).

First the bits from length and then from text are being stored to the array earr[bit] respectively as,

Table 4.1: Data bits in encrypted array

Bit for Length	Bit for Character
earr[1]=1	earr[9]=0
:	:
Earr[8]=0	Earr[80]=0

Let the key (K) =6359.

The image size of both the host image and biometric key image= 150 X 150 (w x h).

Number of effected pixel required for character (p) = [ceil (80/3)] =27.

In the Table 4.2, how the array elements are replaced with R, G & B values in selected nonlinear pixels and bit position about the image is described (as described in Algorithm 3.1.3 & 3.1.4).

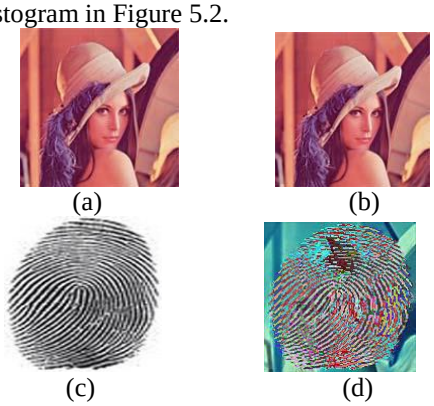
Table 4.2: Replacement of bits about image

Key (K)	Position in tree	Value of pixel P(x,y)	Bit position =(x+y)% 4	Array data to replace
6359	Root	P(63,59)	3^{rd} LSB	earr[1] earr[2] earr[3]
6359	Left Child	P(126,118)	1^{st} LSB	earr[4] earr[5] earr[6]
6359	Right Child	P(127,119)	3^{rd} LSB	earr[7] earr[8] earr[9]
:	:	:	:	:

In this way, we may generate an indistinguishable watermarked image embedding the entire message. Then taking the biometric key image we have produced a visible watermarked image. Afterward, we are able to transmit the encrypted visible watermarked image through any communication channel. Applying the decryption technique as described in Algorithm 3.2.1 also we will be able to get back the encrypted message from that watermarked image at the decryption end.

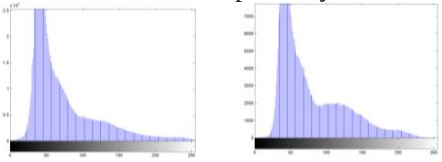
V. EXPERIMENTAL RESULT

An experimental result is given in Figure 5.1 and the histogram in Figure 5.2.

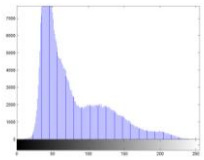


(a) (b) (c) (d)

Figure 5.1(a) is the cover image, (b) the watermarked-image, (c) the fingerprint image (d) the final XOR image respectively



(a) (b)



(c)

Figure 5.2: (a), (b) and (c) the histogram for cover image, watermarked image and final XOR image respectively

A. *Peak Signal to Noise Ratio (PSNR)*

PSNR measures the quality of the image by comparing the original image or cover image with the stego-image, i.e. it measures the percentage of the stego data to the image percentage. The PSNR is used to evaluate the quality of the stego-image after embedding the secret message in the cover. Assume a cover image C(i,j) that contains N by N pixels and a stego-image S(i,j) where S is generated by embedding the message bit stream. Mean squared error (MSE) of the stego-image as follows: $MSE = 1/[NxN]^2 \sum_{i=1}^{N} \sum_{j=1}^{N} [C(ij) - S(ij)]^2$

The PSNR is computed using the following formulae:

$$\text{PSNR} = 10\log_{10} 255^2 / \text{MSE db}.$$

Table 5.1: PSNR values after embedding text in host image

Cover Image Size	Message Size (in character)	PSNR of Respective Coefficient
150*200	10	72.2765
	20	72.2455
	30	72.1966
300*400	10	74.1441
	20	74.1380
	30	74.0994
600*800	10	76.0305
	20	76.8733
	30	76.8111

VI. Analysis

Here, initially we have produced an invisible watermarked image. We have not used any compression and/or encryption technique before the creation of array (earr[ ]). Anybody may employ the compression and/or encryption technique(s) at the time of creation of array (earr[ ]). In that case, the length of array will be less and the strength of encryption will be higher than present. In addition the number of affected pixel will also be fewer than now. Here we have generated root pixel position depending on key value. And the remaining positions are calculated as the child nodes are calculated in binary tree. Binary values generated from the textual information may be replaced at arbitrary pixel positions of the image using any other tree structure. Hare we have used 4-digit key for encryption process. Anybody may use, key using more digits. As bits are placed any one bit in lower four bits of each R, G and B, the change of color of the targeted pixels will be less and so forth it becomes invisible to human eye at the time to make the invisible watermarked image. The number of targeted pixels proportionally varies to size of text. Later we also have taken a biometric image as a key image. Anybody may use any other biometric image like face, eye of any one. At the time of encryption the sender may use the biometric image of himself or as he/she required a key image. After embedding data into host image we have done bitwise XOR operation (pixel by pixel) with the key image. The XOR operation between two images differs from process to process. If, dimension of key image become less than the host image, then the resultant visible watermarked image will also be different as it happens against the initial pixel position of the images[3] [7].

VII. Conclusion

This paper represents a robust hybrid watermarking technique. First, the data bits from the textual message are embedded to the host image and we get an invisible watermarked image. Then we have taken a biometric image as a key image and performed bitwise XOR (pixel by pixel) operation with the previous resultant image. Finally, we get a visible watermarked image. Here, we have used private key cryptographic technique to place the data bits in unlike pixel and bit positions starting from nonlinear pixel position depending on key about
the entire image. Moreover, initially it produces the similar image to see in naked eye at the time of watermarking using this method. If the key become unknown to anybody who wants to attack the information, we think, it will be quite impossible to him or her to find out the information from the watermarked image. Next, we have taken a biometric image as a key image and have performed bitwise XOR (pixel by pixel) operation with the previous resultant image. If the key image becomes mysterious, it will also be quite impractical to attackers to get the invisible watermarked image.

References

[1] P. S. Revenkar, Anisa Anjum, W .Z. Gandhare "Secure Iris Authentication Using Visual Cryptography", International Journal of Computer Science and Information Security, Vol. 7, No.3, 2010, ISSN 1947-5500, pp.-217-221

[2] Souvik Bhattacharyya, Goutam Sanyal "A Data Hiding Model with High Security Features Combining Finite State Machines and PMM method", International Journal of Electrical and Computer Engineering 5:2 2010, pp. 78-85

[3] Sabyasachi Samanta, Saurabh Dutta, Goutam Sanyal, "An Enhancement of Security on Image Applying Asymmetric Key Algorithm", International Journal of Computer Applications (0975 – 8887), Volume 25– No.5, July 2011, pp. 19-23

[4] Thi Hoang Ngan Le, Kim Hung Nguyen, Hoai Bac Le "Literature Survey on Image Watermarking Tools, Watermark Attacks, and Benchmarking Tools", 2010 Second International Conferences on Advances in Multimedia, pp. 67-73.

[5] M. Kameswara Rao, Sushma Yalamanchili, "Copyright Protection of Gray Scale Images by Watermarking Technique Using (N, N) Secret Sharing Scheme", Journal of Emerging Technologies in Web Intelligence, Vol. 2, No. 2, May 2010, pp. 101-105

[6] Mahmoud A. Hassan, Mohammed A. Khalili, "Self Watermarking based on Visual Cryptography", World Academy of Science, Engineering and Technology 8, 2005

[7] V. Lokeswara Reddy, Dr. A. Subramanyam, Dr. P. Chenna Reddy, "Implementation of LSB Steganography and its

Evaluation for Various File Formats", International Journal Advanced Networking and Applications Volume: 02, Issue: 05, Pages: 868-872 (2011)

[8] Mrs. Kavitha, Kavita Kadam, Ashwini Koshti, Priya Dunghav, "Steganography Using Least Significant Bit Algorithm", "International Journal of Engineering Research and Applications (IJERA)" ISSN: 2248-9622, Vol. 2, Issue 3, May-Jun 2012, pp. 338-341

[9] Chung-Ming Wang , Nan-I Wu , Chwei-Shyong Tsai , Min-Shiang Hwang, "A high quality steganographic method with pixel-value differencing and modulus function", The Journal of Systems and Software (2007), pp. 1-9

[10] J. K. Mandal, Debashis Das, "Steganography Using Adaptive Pixel Value Differencing (APVD) of Gray Images through Exclusion of Overflow/Underflow", CS & IT-CSCP 2012, pp. 93-102

[11] Muhammad A Khan, Vidyasagar Potdar, Elizabeth Chang, "An Architecture Platform for Grey Level Modification Steganography System"

[12] Rajkumar Yadav, Ravi Saini, Kamaldeep, "A New Image Steganography Approach for Information Security Using Gray Level Images in Spatial Domain", International Journal on Computer Science and Engineering (IJCSE), Vol. 3, No. 7, July 2011, pp. 2679-2690

[13] Dr. Ekta Walia, Payal Jain, An Analysis of LSB & DCT based Steganography", Global Journal of Computer Science and Technology, Vol. 10, Issue 1 (Ver 1.0), April 2010, pp. 4-8

[14] Chia-Chen Lin, Pei-Feng Shiu, "DCT-based Reversible Data Hiding Scheme", Journal of Software, Vol. 5, No. 2, February 2010, pp. 214-224

[15] Po-Yueh Chen, Hung-Ju Lin, "A DWT Based Approach for Image Steganography", International Journal of Applied Science and Engineering 2006. 4, 3: pp.275-290

[16] M. Abolghasemi, H. Aghaeinia, K. Faez, "Data Hiding Detection Based on DWT and Zernike Moments" 4th International Conference: Sciences of Electronic, Technologies of Information and Telecommunications, March 25-29, 2007 – TUNISIA

[17] Linfeng Guo Yan Meng, "Psnr-Based Optimization of Jpeg Baseline Compression on Color Images", ICIP 2006, pp.1145-1148

[18] Sunita Barve, Uma Nagaraj, Rohit Gulabani, "Efficient and Secure Biometric Image Stegnography using Discrete Wavelet Transform", " International Journal of Computer Science & Communication Networks", Vol 1(1),September-October 201, pp. 96-99

[19] Linfeng Guo, Yan Meng,"PSNR-Based Optimization of JPEG Baseline Compression on Color Images", ICIP 2006, pp. 1145-1148

[20] Yedla Dinesh, Addanki Purna Ramesh, "Efficient Capacity Image Steganography by Using Wavelets", International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622, Vol. 2, Issue 1, Jan-Feb 2012, pp.251-259

[21] Atallah M. Al-Shatnawi, "A New Method in Image Steganography with Improved Image Quality", Applied Mathematical Sciences, Vol. 6, 2012, pp. 3907 - 3915

AUTHOR'S PROFILE

Sabyasachi Samanta

is working as Assistant Professor at Dept. of IT, Haldia Institute of Technology Haldia, WB, India. He has received M.Tech. Degree in IT and currently pursuing Ph.D. at National Institute of Technology, Durgapur, WB, India. His main research interest includes watermarking, steganography and cryptography.

Saurabh Dutta

is a professor in Dr. B. C. Roy Engineering College. He holds a Ph. D. Degree in Computer Science. His research domain is information security and cryptology.

Gautam Sanyal

is a member of the IEEE. He has received his B.E and M.Tech degree from National Institute of Technology (NIT), Durgapur, India. He has received Ph.D. (Engg.) from Jadavpur University, Kolkata, India, in the area of Robot Vision. He possesses an experience of more than 25 years in the field of teaching and research. He has published nearly 68 papers in International and National Journals / Conferences. Three Ph. Ds (Engg.) have already been awarded under his guidance. At present he is guiding six Ph. Ds scholars in the field of steganography, Cellular Network, High Performance Computing and Computer Vision. He has guided over 10 PG and 100 UG thesis. His research interests include Natural Language Processing, Stochastic modeling of network traffic, High Performance Computing, Computer Vision. He is presently working as a Professor in the department of Computer Science and Engineering and also holding the post of Dean (Students' Welfare) at National Institute of Technology, Durgapur, India.