

Secure Watermark Image Steganography by Pixel Indicator Based on Randomization

Zena Ahmed⁺ and M. Hamid Mohammed Farhan⁺⁺

College of Electrical and Electronics Technique

⁺ E-mail:zena782004@yahoo.com

⁺⁺ E-mail:hamd_farhan@yahoo.com

Abstract

Steganography is the art of hiding the fact that communication is taking place, by hiding information in other information. In this paper have being investigated a method to RGB image steganography based on pixel indicator technique and triple-A algorithm. They uses the same principle of Least Significant Bit (LSB), where the secret is hidden in the least significant bits of the pixels, with more randomization in selection of the number of bits used and the color channels that are used. This randomization is expected to increase the security of the system and also increase the capacity of information. These techniques can be applied to RGB images where each pixel is represented by three bytes (24 bit) to indicate the intensity of red, green, and blue in that pixel. This work showed attractive results especially in the capacity of the data-bits to be hidden with relation to the RGB image pixels. The effective of the proposed stego system has been estimated by Mean Square Error (MSE), Peak Signal to Noise Ratio (PSNR). This paper also illustrates how security has been enhanced using this algorithm.

Keywords: Steganography, Randomization, Triple-A Algorithm, Pixel Indicator Algorithm and Computer Security.

إخفاء الصورة المائية من المؤشر بكسل استنادا إلى التوزيع العشوائي

خلاصة

إخفاء المعلومات هو فن إخفاء حقيقي وإن الاتصالات تجري عن طريق إخفاء المعلومات في غيرها من المعلومات. في هذا البحث لدينا وسيلة لتحقيق إخفاء المعلومات صورة RGB على أساس تقنية مؤشر بكسل والثلاثي خوارزمية. أنها تستخدم نفس مبدأ LSB، حيث يتم إخفاء سر في بت على الأقل أهمية من بكسل، مع مزيد من العشوائية في اختيار عدد البتات المستخدمة وقنوات الألوان التي يتم استخدامها. ومن المتوقع أن هذا التوزيع العشوائي لزيادة أمن النظام، وكذلك زيادة قدرته. ويمكن تطبيق هذه التقنيات لصور RGB حيث يتم تمثيل كل بكسل من ثلاثة بايت للدلالة على كثافة اللون الأحمر والأخضر والأزرق في تلك بكسل. وأظهرت نتائج هذا العمل جذابا خصوصا في قدرة بت البيانات إلى أن تكون مخفية مع يتعلق بكسل صورة RGB. وقد قدر من فعالية نظام stego المقترحة عن طريق مربع الخطأ (MSE)، وإشارة إلى الذروة نسبة الضوضاء (PSNR). هذا البحث كما يوضح كيف تم تعزيز الأمن باستخدام هذه الخوارزمية.

1. Introduction

Steganography is the process of hiding of a secret message within an ordinary message and extracting it at its destination. Anyone else viewing the message will fail to know that it contains secret/encrypted data. The word comes from the Greek word “steganos” meaning “covered” and “graphei” meaning “writing” [1, 2, 3].

The LSB method insertion is a very simple and common approach to embedding information in an image in special domain. The limitation of this approach is vulnerable to every slight image manipulation [4].

Some techniques have been used for image steganography such as LSB, SCC and image intensity. In LSB, the least significant bit of each pixel for a specific color channel or for all color

channels is replaced with a bit from the secret data. Although it is a simple techniques, but the probability of detecting the hidden data is high. The Stego Color Cycle (SCC) technique is an enhancement [5].

2. Steganography

Steganography techniques require two files: cover media, and the data to be hidden [6]. When combined the cover image and the embedded message make a stegoimage, which is in our work for image steganography known as stego-image image [7]. One of the commonly used techniques is the LSB, its work by they pixel is replaced by bits of the secret till secret message finishes [5,8]. The risk of information being uncovered with this method as is, is susceptible to all ‘sequential scanning’ based techniques [7], which is threatening its security.

The random pixel manipulation technique attempts at overcoming this problem, where pixels, which will be used to hide data are chosen in a random fashion based on a stego-key. However, this key should be shared between the entities of communication as a secret key [9].

Image-based steganography techniques need an image to hide the data in. This image is called a cover media. Digital images are stored in computer systems as an array of points (pixels) where each pixel has three color components: Red, Green, and Blue (RGB). Each pixel is represented with three bytes to indicate the intensity of these three colors (RGB) [7].

The color channel, where the secret data will be hidden in, is cycling frequently for every bit according to a specific pattern [10]. For example, the first bit of the secret data is stored in the LSB of red channel, the second bit in the green channel, the third bit in the blue channel and so on. This technique is more secure than the LSB but still it suffers detecting the cycling pattern that will reveal the secret data. Also it has less capacity than the LSB. StegoPRNG is also a different technique that uses the RGB images. However in this technique, a pseudo random number generator (PRNG) is used to select some pixels of the cover image. Then, the secret will be hidden in the Blue channel of the selected pixels. Again this technique has the problem

of managing the key, and problem of capacity since it uses only the Blue channel out of the three channels of their available channels [5].

3. The Proposed Technique

The proposed method takes advantage of psycho visual redundancy and the dependency of a pixel. A color image is generally formed by three different bands, such as red, green, and blue, in a color coordinate system. In the proposed algorithm, the data hiding procedure is performed on R, G, and B bands, respectively. The step-by-step procedure is stated as follows and Table (1) shows the Meaning of indicator values. The proposed algorithm was implemented in Matlab ver 7.1. The devised method consists of two main processes. First one deals with the hiding data which passes some controls to matlab GUI for implementation of LSB hiding algorithm. The other process then returns back the reverse information in the cover RGB color image .we have implemented steganographic routines in matlab using the GUI toolbox. The suggested technique tries to solve the problem of the previous two techniques by using one of the channels as an indicator for data existence in the other two channels and the indicator is set randomly by nature.

Designing any stego algorithm should take into consideration the following three aspects (Figure 1):

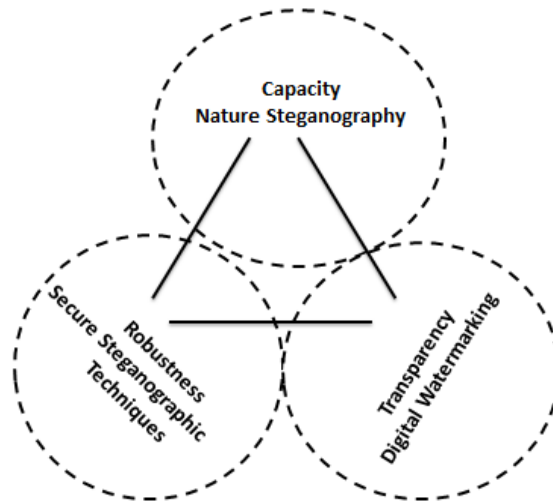


Figure 1. Steganography tradeoff parameters

- **Capacity:** The amount of data that can be hidden without significantly changing the cover medium.
- **Robustness:** the resistance for possible modification or destruction in unseen data.
- **Invisibility (Security or Perceptual Transparency):** The hiding process should be performing in a way that it does not raise any suspicion of eavesdroppers.

Figure 1, shows the relation between these three main parameters. If we increase the capacity of any cover to store more data than a practical possible threshold, then its transparency or robustness will be affect and vice versa. Similarly, transparency and robustness are related; if any of these two parameters are influenced, it can affect the performance in the other one. The capacity, robustness, and security parameters relation issues can be driven by the application need and its priorities [11].

Table 1. Meaning of indicator values.

indicator	$K=0, I_{LSB} \leftarrow I_{BIN}$	$K=1, I_{LSB} \leftarrow I_{BIN}$
KGB	G	B
RKB	R	B
RGK	R	G

3.1 Embedding Algorithm:

Algorithm for secret data embedding process:

Begin

Input : Color cover Image (I_c) and Secret Image (I_S)

Output : Stego Cover Image

Step 1 : Read the Color cover image (I_c) and the Image data to be embedded

Step 2 : Generate a randomize Binary map key
Step 3 : Split the cover into RGB planes
Step 4 : choose which RGB plane to save map key
Step 5 : Repeat step5 for all row and column of cover image
Step 6 : Read each pixel in map key
 If pixel =0 then embedded the secret pixel of image in channel one
 Else
 embedded the secret pixel of image in channel two
Step 7 : Combine the RGB plane to form stego cover
End

3.2 Reconstructed Algorithm:

Algorithm for secret data recovery process:

Begin
Input : Stego Cover Image (I_s)
Output : Secret Image (I_S)
Step 1 : Read the stego image (I_s) and split to RGB planes
Step 2 : Split the cover into RGB planes
Step 3 : determine where randomize Binary map key
Step 4: Repeat step5 for all row and column of cover image
Step 5 : Read each pixel in map key
 If pixel =0 then get LSB of the secret pixel of image in channel one
 Else
 Get LSB of the secret pixel of image in channel two
Step 6 : Save the secret Image (I_S)
End

3.3 Error Metrics

The effectiveness of the stego process proposed has been studied by estimating the following four metrics for both cover images. Bit Error Rate (BER) evaluates the actual number of bit positions which are replaced in the stego image in comparison with cover image. It has to be computed to estimate exactly how many bits of the original cover image (I_c) are being affected by stego process. The BER for the Stego image (I_s) is the percentage of bits that have errors relative to the total number of bits considered in I_c .

Let I_{cbin} and I_{sbin} are the binary representations of the cover image and stego cover then [12],

The total number of bit errors,

$$T_e = \sum_{i=1}^n |I_{cbin} - I_{sbin}| \dots \dots \dots (1)$$

and the bit error rate $BER = T_e / T_n$

T_n is the total number of bits considered for the gray image of size $M \times N$ pixels.
 T_n will be $M \times N \times 8$.

Peak Signal to Noise Ratio (PSNR):

The PSNR is calculated using the equation [12],

$$PSNR = 10 \log_{10} \left(\frac{I_{max}^2}{MSE} \right) \text{dB} \dots \dots \dots (2)$$

Where I_{max} is the intensity value of each pixel which is equal to 255 for 8 bit gray scale images. Higher the value of PSNR better the image quality

Mean Square Error (MSE)

The MSE is calculated by using the equation [12],

For Gray scale Images: $MSE = \frac{1}{MN} \sum_{i=1}^M \sum_{j=1}^N (x_{i,j} - y_{i,j})^2 \dots \dots (3)$

For color Images: $MSE = \left[\frac{MSE(R) + MSE(G) + MSE(B)}{3} \right] \dots \dots \dots (4)$

Where M and N denote the total number of pixels in the horizontal and the vertical dimensions of the image $x_{i,j}$ represents the pixels in the original image and $y_{i,j}$ represents the pixels of the stego-image.

4. Results and Discussion

By selected the three different images in (sizes, application field) to perform in the testing have shown in Table (2):

Table 2. The Size of Original Cover Image and the Binary Image of Three Tests.

Cover image	Size of cover image	Size of binary image
Lena	512x512	400x333
Sun	280x210	2100x2100
Colored	246x165	1138x1508

The Results of our method is shown in Table (3), and using Image Quality (In PSNR MSE) ,Then three digital images has been taken as cover images for the processed method are shown in the following Figures (2-4).

Table 3. Results in terms of Image Quality (In PSNR MSE) using RGB channel for different color images (for BPP=8/3).

Cover Image	Channel 1 Red		Channel 2 Green		Channel 3 Blue	
	MSE	PNSR	MSE	PNSR	MSE	PNSR
Lena	0.44056	51.6907	0.44042	51.6921	0.43972	51.6991
Sun	0.27425	53.7494	0.27464	53.7431	0.27322	53.7658
Colored	0.30999	53.2173	0.3098	53.22	0.31084	53.2054

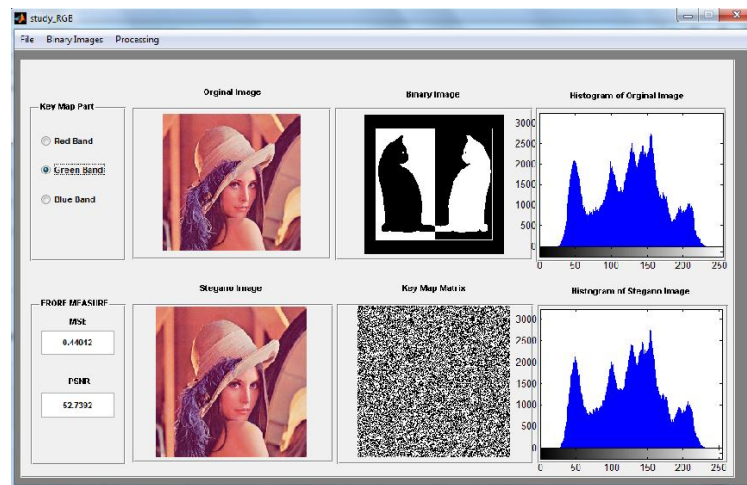


Figure 2. Original cover images and the binary image of Test1

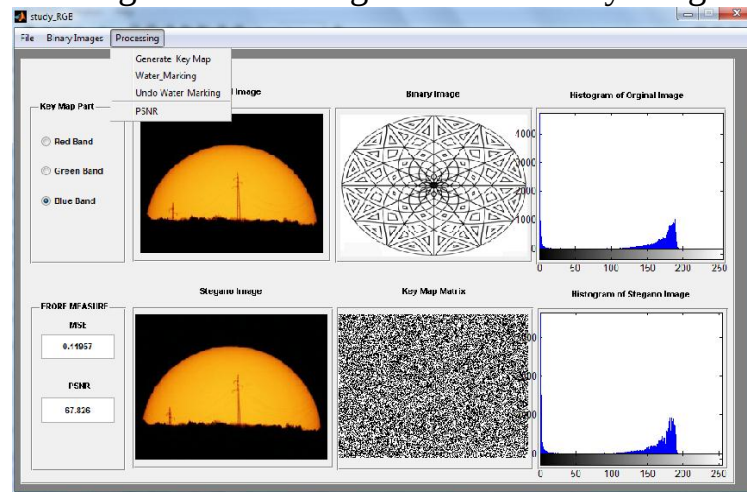


Figure 3. Original cover images and the binary image of Test2

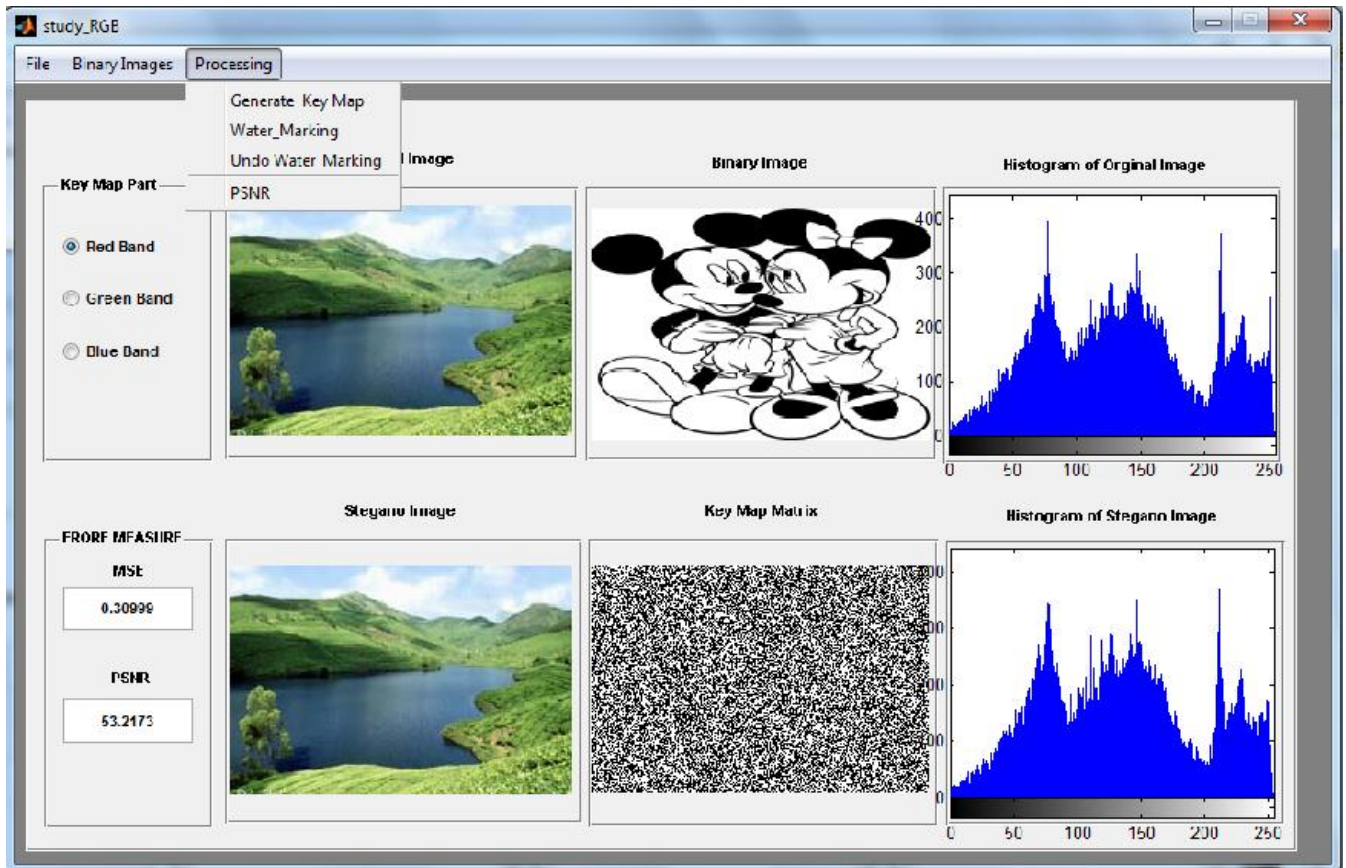


Figure 4. Original cover images and the binary image of Test3

5. Conclusions

Demonstrated a new watermarking technique that uses RGB band scheme to embed a copyright image or any binary image into original image. This technique works well with images of all sizes. This technique provides a Randomize key map embed in one channel and two other channels will content the copyright images or binary images are embedded into original image for copyright protection. Also, the embedded image is shared among n participants where all the n shares must be used to reconstruct the embedded image. This makes the system more secure. The method can with stand

attacks like JPEG compression, resize and adding noise with less loss in quality of the image. In this work we can be used either symmetric key or public key and embed the hash value inside the image so that at the receivers end the authentication and integrity of the image can be verified by recalculating the hash and verifying it. Similarly digital signatures can be generated for images and can be verified. Also (k,n) threshold secret sharing schemes can be implemented for much security.

Experimental results have shown that the proposed method provides an efficient way for embedding large

amount data into cover images without making noticeable distortions. Moreover, the proposed methods use less than half of the total number of pixels in an image where methods discussed in references [8,13] use almost all the pixels of an image for the same amount of hiding capacity.

References

1. N. F. Johnson and S. Katzenbeisser, "A Survey of Steganographic Techniques", in S. Katzenbeisser and F. Peticolas (Eds.): *Information Hiding*, Artech House, Norwood, MA, Page (43-78), 2000.
2. Lou, D. C. and Liu, J. L. "Steganography Method for Secure Communications", *Elsevier Science on Computers & Security*, 21, 5, Page (449-460), 2002.
3. J. Fridrich and M. Goljan, "Practical Steganalysis of Digital Images-state of the Art", *Proc. SPIE Photonics West*, Vol. 4675, San Jose, California, Page (1-13), January 2002.
4. C.K. Chan, L. M. Chen, "Hiding data in images by simple LSB substitution", *Pattern Recognition*, Vol. 37, Page (469-474), 2004.
5. K. Bailey, K. Curran, "An Evaluation of Image Based Steganography Methods", *Multimedia Tools & Applications*, Page (55-88), Vol. 30, No. 1, July 2006.
6. A. Gutub, L. Ghouti, A. Amin, T. Alkharobi, M.K. Ibrahim, "Utilizing Extension Character 'Kashida' With Pointed Letters for Arabic Text Digital Watermarking", *Inter. Conf. on Security and Cryptography - SECRIPT*, Barcelona, Spain, 28 - 31, July 2007.
7. N.F. Johnson, S. Jajodia, "Exploring Steganography: Seeing the Unseen", *IEEE computer*, Vol. 31, No. 2, Page (26-34), February 1998.
8. K. Hempstalk, "Hiding Behind Corners: Using Edges in Images for Better Steganography", *Proceedings of the Computing Women's Congress*, Hamilton, New Zealand, Page (11-19), February 2006.
9. A. Gutub, M. Ankeer, M. Abu-Ghalioun, A. Shaheen, and A. Alvi, "Pixel indicator high capacity technique for RGB image based Steganography", *WoSPA 2008-5th IEEE International Workshop on Signal Processing and its Applications*, University of Sharjah, Sharjah, U.A.E., Page (18-20), March 2008.
10. R. Chandramouli, N. Memon, "Analysis of LSB Based Image Steganography Techniques" *Proc. IEEE ICIP*, Page (1019-1022), 2001.
11. N. Tiwari¹, M. Shandilya, "Secure RGB Image Steganography from Pixel Indicator to Triple Algorithm-An Incremental Growth", *International Journal of Security and Its Applications*, Page (53-62), Vol. 4, No. 4, October, 2010.
12. R. Amirtharajan, R. John, "Constructive Role of SFC & RGB Fusion versus Destructive Intrusion", *International Journal of Computer Applications*, Page (0975 - 8887) Vol. 1, No. 20, 2010

13. G.C. Kessler, "An Overview of Steganography for the Computer Forensics Examiner", Forensic Science Communications, Vol. 6, No. 3, July 2004.