ISSN 1991- 8690 الترقيم الدولي 8690 - 1991

A Novel Technique for Hiding Text and Image in Color Image

Aeada K. Al-Bedri

Dept. of Electrical Engineering - College of Engineering - University of Basrah

E-mail: aydaa74@yahoo.com

Abstract:

The growing possibilities of modern communications need the special means of security especially on computer network. The network security is becoming more important as the number of data being exchanged on the Internet increases. Therefore, the confidentiality and data integrity are requires to protect against unauthorized access and use. This has resulted in an explosive growth of the field of information hiding.

In this paper, a hiding technique by using modified third-least-significant-bit (LSB3) substitution to hide text and binary image in color image is proposed (text in the red color space and binary image in the green space of color cover image). Typically LSB method is used least-significant-bit-plane (LSB 1) but here we used LSB 3 plane to increase the robustness. The protection of the text and image can be greatly improved with use of stego-key which has been used to permute the text characters and binary image bits before embedding them. Experimental results show that the stego-image is visually indistinguishable from the original cover-image. Therefore, we can embed two different secret messages in one color cover image successfully.

Keywords: Steganography, Cover-image, LSBs, PSNR.

طريقة جديدة لإخفاء نص وصورة داخل صورة ملونة

عائدة كاظم عريبي البدري

قسم الهندسة الكهربائية - كلية الهندسة - جامعة البصرة

البريد الالكتروني: aydaa74@yahoo.com

الخلاصة:

النمو المتزايد المحتمل لوسائل الاتصالات الحديثة في حاجة إلى وسائل خاصة لأمن المعلومات على شبكات الكمبيوتر. أمن الشبكة أصبح أكثر أهمية نظرا لأن لتزايد عدد البيانات التي يتم تبادلها على الإنترنت. لذلك، موثوقية وتكاملية البيانات تتطلب حماية ضد الوصول للبيانات و الاستخدام غير المصرح به . وقد أدى هذا إلى النمو الهائل في حقل إخفاء المعلومات.

الطريقة المقترحة تعتمد على تخزين ثنائيات النص في فضاء اللون الأحمر من الطبقة الثالثة LSB3 من الصورة الملونة وتخزين ثنائيات الصورة الثنائية في فضاء اللون الأخضر من الطبقة الثالثة LSB3 لنفس الصورة الملونة. الطرق المعتادة لإخفاء البيانات عادة ما تستخدم تخزين البيانات في الطبقة الأولى LSB 1 من الصورة الملونة ولكن في طريقتنا المقترحة تم استخدام الطبقة الثالثة LSB3 وذلك لزيادة متانة بيانات النص المخفي والصورة الثنائية داخل الغطاء. وللمزيد من الحماية لبيانات النص والصورة الثنائية قمنا ببعثرتها باستخدام مفتاح– أخفاء قبل أن يتم إخفائها في صورة الغطاء. نتائج التجارب للطريقة المقرحة أظهرت أنه لا توجد اختلافات واضحة بين الصورة الملونة قبل إخفاء البيانات داخلها وبعد أخفاء البيانات. لذلك يمكن إخفاء نوعين من الرسائل في صورة ملونة واحدة.

الكلمات المفتاحية :كتابة مخفية، صورة–غطاء، البت الأقل أهمية، نسبة قمة الإشارة–إلى–الضوضاء.

1. Introduction:

1.1 Steganography

Steganography coming from the Greek word stegos, meaning covered and graphia, which means writing. Steganography is the art and science of hiding the information within information [1]. The goal of steganography is to hide messages inside other harmless messages in a way that does not allow any enemy to even detect that there is a second message present. In other words, steganography is used to avoid drawing suspicion to the existence of a hidden message. This approach of information hiding technique has recently become important in a number of application areas. Digital audio, video, and pictures are increasingly furnished with distinguishing but imperceptible marks, which may contain a hidden copyright notice or serial number or even help to prevent unauthorized copying directly [2].

1.2 Data Hiding in Image Files

Coding secret messages in digital images is by far the most widely used of all methods in the digital world of today. This is because it can take advantage of the limited power of the human visual system (HVS). Almost any plain text, cipher text, image and any other media that can be encoded into a bit stream can be hidden in a digital image. With the continued growth of strong graphics power in computers and the research being put into image based steganography, this field will continue to grow at a very rapid pace [3].

To a computer, an image is a collection of numbers that constitute different light intensities in different areas of the image. The numeric value representation forms a grid and the individual points are referred to as pixels. Image is the most popular cover objects used for steganography [4]. When dealing with digital images for use with steganography, 8-bit and 24-bit per pixel image files are typical. 8-bit images are a great format to use because of their relatively small size. But there is a drawback is that only 256 possible colors can be used which can be a potential problem during encoding. Usually a gray scale color palette is used when dealing with 8-bit images such as Graphic Interchange File Format (.GIF) because its gradual change in color will be harder to detect after the image has been encoded with the secret message. 24-bit images offer much more flexibility when used for steganography. The large numbers of colors (over 16 million) that can be used go well beyond the HVS, which makes it very hard to detect once a secret message, has be encoded. The other benefit is that a much larger amount of hidden data can be encoded into 24-bit digital image as opposed to an 8-bit digital image [3].

Chin-Chen Chang [8] has proposed a model in which the data is embedded into the cover image by changing the coefficients of a transform of an image such as discrete cosine transform. The high compression rate is one of the advantages of fractal image compression. Another advantage is the good image quality, after enough iteration for decompression. But the computation time required to encode an image might be very long due to an exhaustive search for the optimal code. Wang et al. [9] proposed to embed secret messages in the moderately significant bit of the cover-image. A genetic algorithm is developed to and an optimal substitution matrix for the embedding of the secret messages. They also proposed to use a local pixel adjustment process (LPAP) to improve the image quality of the stego-image.

1.3 LSB Steganography

Least Significant Bit (LSB) substitution method is a very popular way of embedding secret messages with simplicity. The fundamental idea here is to insert the secret message in the least significant bits of the images. This actually works because the human visual system is not sensitive enough to pick out changes in color whereas changes in luminance are much better picked out [5]. LSB steganography, in which the lowest bit plane of a bitmap image is used to convey the secret data, has long been known to steganographers. Because the eye cannot detect the very small perturbations it introduces into an image and because it is extremely simple to implement. LSB methods are commonly used among the many free steganography tools available on the Internet. There are two types of LSB steganography: *LSB replacement* can be uncovered relatively easily. In the particular case when the covers are grayscale images. The LSB matching embedding algorithm is as follows.

Convert the secret data into a stream of bits. Take each pixel of the cover image (possibly in a pseudo-random order generated by a shared secret key): if the LSB of the next cover pixel matches the next bit of secret data, do nothing; otherwise, choose to add or subtract one from the cover pixel value, at random. When the secret message is fewer bits in length than the number of pixels in the cover image, the pseudo-random permutation ensures that changes are spread uniformly throughout the image. The allowable range of pixel values will force the decision of whether to increment or decrement, when the cover pixel is saturated [6].

A 24-bit image provides the most space for hiding information. All color variations for the pixels are derived from three primary colors: red, green, and blue. Each primary color is represented by 1 byte. To hide an image in the LSBs of each byte of a 24-bit image, you can store 3 bits in each pixel. A 1,024 x 768 image has the potential to hide a total of 2,359,296 bits (294,912 bytes) of information. If you compress the message to be hidden before you embed it, you can hide a large amount of information. To the human eye, the resulting stego-image will look identical to the cover image.

For example, the letter A can be hidden in three pixels (assuming no compression) the original raster data for 3 pixels (9 bytes) may be

(00100111 11101001 11001000)

(00100111 11001000 11101001)

(11001000 00100111 11101001)

The binary value for A is **10000011**. Inserting the binary value for A in the three pixels would result in

(00100111 11101000 11001000)

(00100110 11001000 11101000)

(11001000 00100111 11101001)

The underlined bits are the only three actually changed in the 8 bytes used. On average, LSB requires that only half the bits in an image be changed. You can hide data in the least and second least significant bits and still the human eye would not be able to discern it [7].

2. The Proposed Technique:

In this paper, we propose a security mechanism (which previously used to embed text into color image [10]) that used LSB3 substitution scheme to hides text and binary image in the same pixel of color image (text in the red color space and image in the green color space), a 256 x 256 color image has been used. So, a message (text) up to 65536 bits (8192 bytes) can be hidden. Let the secret text is *T* = {t0, t1, t2, …, tN-1}, where 1 <= N < = 65536, N is the length of the message that is embedded and let the secret binary image *S* form a bit string *S* = *s*0*s*1…*sn*-1, where *n* is the number of secret bits.

Suppose that LSB 3 of the cover image is *C*r = {cr0 , cr1 , cr2 , …, cr65535},
where crj = {0, 1} for each j = 0, ..., 65535 for the red color space so that
*C*g = {cg0, cg1, cg2, …, cg65535}, where cgj = {0, 1} for each j = 0, ..., 65535 for the green color space.

To increase the secret of the message (text and image) Stego-Key is used, which is used to seed pseudo-random number generator (PRNG) which is used to generate permutated characters of the secret message (text Tp and image Sp). The bits in cover image *Cr*, which will be replaced by the bit secret text *Tp*, in the same time the bits in cover image *C*g will be replaced by the secret binary image *Sp* of the secret image, to obtain stego-image Z.

2.1 Embedding Algorithm

Embedding algorithm according to the proposed technique is as follows:

1- Read the binary image.
2- Input the text that is to embed.
3- Convert the text to binary form.
4- Read the color image.
5- Permute the text using stego-Key to compute Tp from T.
6- Permute the binary image using stego-Key to compute Sp from S.
7- Replaces the permutated characters of the message Tp by the red color space of LSB 3 (Cr), replaces the permutated characters of the message Sp by the green color of LSB 3(Cg) , set of the cover to obtain the new stego-image Z .
8- Output the stego-image Z.

Figure (1) shows the block diagram of embedding algorithm.

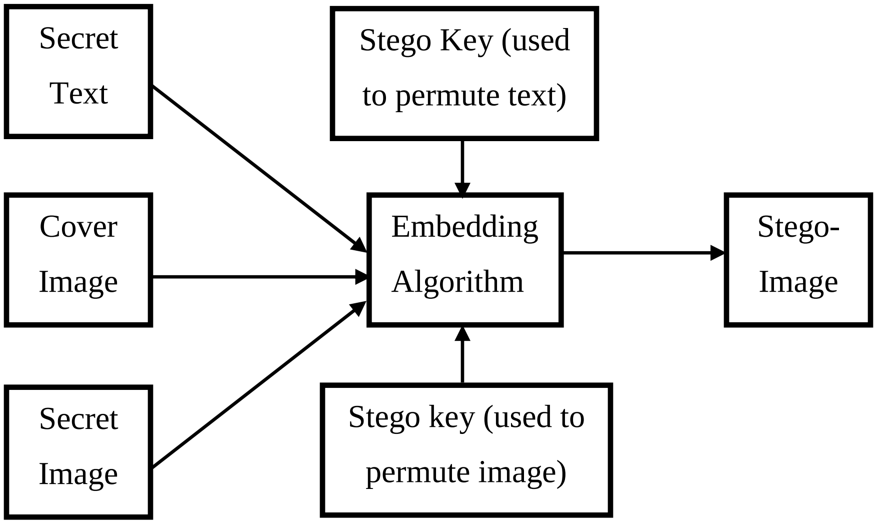
Secret Text
Stego Key (used to permute text)
Cover Image
Embedding Algorithm
Stego-Image
Secret Image
Stego key (used to permute image)

Figure (1) : Embedding Algorithm.

3. Experimental Results:

In this section, we show the experimental results of our scheme. We have written our approach in MATLAB version 7.1 and the running environment is a 1.73 GHz Genuine Intel (R) CPU and 1024 Mb RAM, with a Windows Vista operating system using three color images of size

(256 x 256) as a cover images and data secret (binary image and text) we use the following text as a message . Figure (2) explains the text we need to hide it with size of 6500 bits.

To a computer, an image is a collection of numbers that constitute different light intensities in different areas of the image. The numeric value representation forms a grid and the individual points are referred to as pixels. Image is the most popular cover objects used for steganography [4]. When dealing with digital images for use with steganography, 8-bit and 24-bit per pixel image files are typical. 8-bit images are a great format to use because of their relatively small size. But there is a drawback is that only 256 possible colors can be used which can be a potential problem during encoding. Usually a gray scale color palette is used when dealing with 8-bit images such as Graphic Interchange File Format (.GIF) because its gradual change in color will be harder to detect after the image has been encoded with the secret message. 24-bit images offer much more flexibility when used for steganography.

Figure (2): The text as message.

First of all, we permute the text (by using the stego-Key to compute Tp) that we want to hide in the cover image Figure (3) explains the message after permutation. In the same time, we permute the image (by using the stego-Key to compute Sp) that we want to hide in the cover image. Figure (4) explains the image we need to hide it before and after permutation.

,n daoelgefptaro f mi8Fessd ttsnljil-ire .ihtltt hesrtdl
eewoceihematciec.aegan ra tset ae gae mx ebpb nrihra nIuame cc. wl
giii bi naihmfcooyhie raeospyoea tg p e ewnrlr.aoa astm,t fggastonid uf
eGsa rai bneollewtafallwes2onmrlameroriclaebt htcouii.rmel Ush Itam
nsplasl4is 6aold taeoeu oeniteergimin i r s ettdh c o creieahseastp
gFosagnai ffmlcurcsi-ne ee sii hn vhso hl anlee ubgBtichde a
gchiinrihTu(idgie e4riebfl)m te wduthd t n roF ph2yxbieyvo d
aocctpfpbytnsoeri tagoiTegneeonsob]rsseraldc atanm bp tcdahuos
dt.lba. gtirs oitkivursgse oitgeytt ahdf pnr ei ea pherysutdtar er
hehealeo-seeure almihl[sn i stw.asiyr gene ais l dp nmiaf ih nrmn
dreaepbe tz atrtbIoia rofe i leetG wails-defee a oaiirrr eiif gr
ggebcuentbngboe 2a gsuws.rlgd uttlotlehsha fiupnclet f resetn t
8eathisulsan4treoied opv mcc smg eee daugfh uuiteaa oi et i nrteeps
axsa e csurtacg5mWcodoase m rcnn-ta8yrceha

Figure (3): Permutated text.

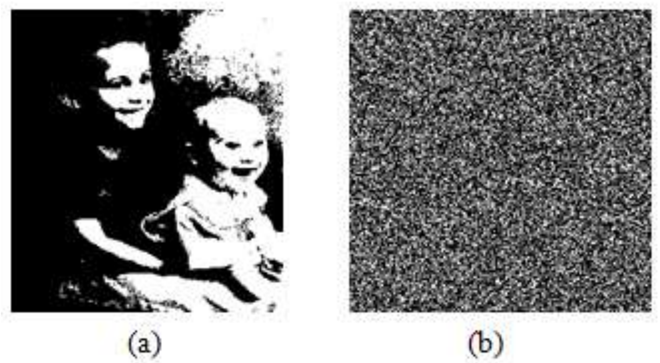

Figure (4) : hide image
a- Image before hide b- Image after hide

So, we used budgie, boys and fishes as color cover images for comparison. Figure (5) shows the three cover images.

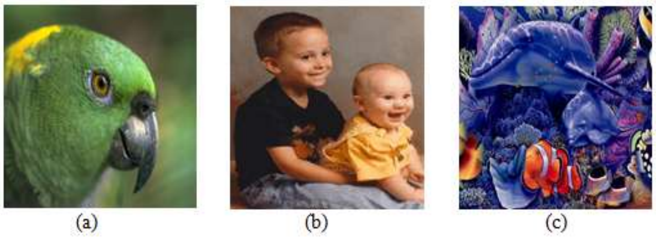

Figure (5) : Color covers images
a-Budgie cover image b- Boys cover image
c- Fishes cover images

The well known *peak-signal-to-noise ratio* (PSNR) which is classified under the difference distortion metrics can be applied on the stego-images, As a performance measurement for image distortion. It is defined as [8]:

$$PSNR = 10\log_{10}\frac{255^2}{MSE} \qquad \text{….(1)}$$

and Mean-Square Error (MSE) is defined as:

$$MSE = \left[\frac{1}{H\times W}\right]\sum_{i}^{H}\sum_{j}^{W}\left(x_{ij} - x'_{ij}\right)^2 \qquad \text{….(2)}$$

Where H, W are the size of the cover image (H = 256,W = 256 in this paper), xij: is the original cover image, and x'ij is the stego-image.

For color images, the reconstruction of all three color spaces must be considered in the PSNR calculation. The MSE is calculated for the reconstruction of each color space. The average of these three MSEs is used to generate the PSNR of the reconstructed RGB image.
The color PSNR equations are as follows:

$$PSNR = 10\log_{10}\frac{255^2}{MES_{RGB}} \qquad \text{…..(3)}$$

$$MSE_{RGB} = \frac{MSE_{red} + MSE_{green} + MSE_{blue}}{3} \qquad \text{...(4)}$$

Where MSE_{red} (or green or blue) is similar to Equation (2) for each color space.

Experiment 1

Now we implement proposed method to embed the message (Tp and Sp) in three covers images separately.

We obtain the following results, as shown in Table (1). Figure (6) explains the three stego-images after embedding the message (Tp and Sp).

Table (1): Results of embedding message (Tp and Sp) in three color

Color Cover images	PSNR (db)
Budgie	45.12
Boys	45.5744
Fishes	45.4252

Figure (6): Stego-images

Experiment 2

To make comparison for our results with the same method if we embed only one message, we embed the same text (Tp) in the fishes image and embed the image (Sp) in the fishes image separately. We obtain the following results, as shown in Table (2).

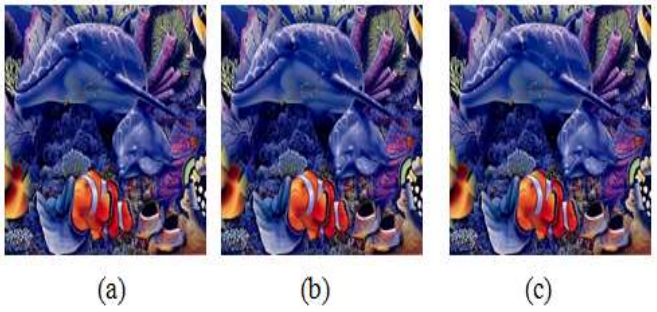

Figure (7): Fishes image with versus type of embedded messages
(a) Fishes stego-image with (text and image).
(b) Fishes stego-image with text only.
(c) Fishes stego-image with image only.

Table (2): Results of embed versus type messages in Fishes image

Color Cover images	PSNR (db)
Fishes with (Tp and Sp)	45.4252
Fishes with Tp	48.9614
Fishes with Sp	47.9664

4. Conclusion:

Steganography is useful for hiding messages for transmission. Our goal in this paper is to propose a steganography mechanism that allows the hiding of two types of messages in the LSB3 of one color image. In this paper, we focus our concern in image because of it is widely used in Internet and also in mobile system.

With the development of the Internet, information processing technologies and the rapid development of communication, it is necessary to share information resources, and the network has becoming the main means of communication. Also, the time of transmit it is very important our proposed technique save the time which need to transmit two hidden message. The proposed technique can easily be implemented and do not visually degrade the image to the point of being noticeable.

Table (2) shows that the results of the proposed technique have less noticeable changes in the value of PSNR when we embed text and binary image in the color cover image than when we embed only (text or binary image) in the same color cover image. Therefore, we can say the proposed technique it acceptable to embed these types of messages. Figure (8) shows the results of embed different secret message in fishes image.

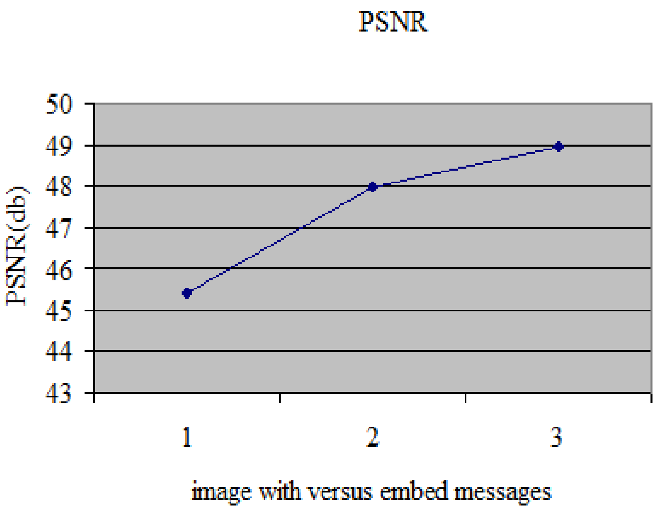

Figure (8): Fishes image with versus type of embedded messages

5. References:

[1] P. K. Singh, and R.K. Aggrawal., " *Enhancement of LSB Based Steganography for Hiding Image in Audio*", IJCSE International Journal on Computer

Science and Engineering, Vol. 02, No. 05, pp. 1652-1658, 2010.

[2] M.M. Amin, M. Salleh, S. Ibrahim, and M.R Katmin, "*Information Hiding Using Steganography*", 4th National Conference on Telecommunication Technology Proceeding (NCTT2003), pp. 21-25, 14-15 January 2003.

[3] S. M. Thampi, *"Information Hiding Techniques: A Tutorial Review*", ISTE-STTP on Network Security & Cryptography, LBSCE 2004.

[4] M. Sitaram Prasad, S. Naganjaneyulu, CH. Gopi Krishna and C. Nagaraju, *"A Novel Information Hiding Technique for Security by Using Image Steganography",* Journal of Theoretical and Applied Information Technology, pp.35-39, 2009.

[5] F. I. Alam, F. K. Bappee and F. U. A. Khondker *, " An Investigation into Encrypted Message Hiding Through Images Using LSB*", International Journal of Engineering Science and Technology (IJEST), Vol. 3 No. 2, pp. 948-960, Feb 2011.

[6] A. D. Ker, *"* Steganalysis *of LSB Matching in Grayscale Images* ", IEEE Signal Processing Letters, Vol. 12, No. 6, pp. 441-444, June 2005.

[7] N. F. Johnson and S. Jajodia, *" Exploring Steganography: Seeing the Unseen"*, Computing Practices, pp. 26-34,0018-9162/ 1998 IEEE.

[8] K. M. Prasad, V.Jyothsna, S.H.K. Raju and S.Indraneel, *"High Secure Image Steganography in BCBS Using DCT and Fractal Compression"*, IJCSNS International Journal of Computer Science and Network Security, Vol. 10 No. 4, pp. 162-170, April 2010.

[9] R. Wang, Ch. Lin, J. Lin,*"Hiding Data in Images by Optimal Moderately Significant –bit Replacement "*, IEE Electron. Lett, Vol. 36, No. 25, pp. 2069–2070, 2000.

[10] H. Younis, *" A Suggested Technique for Data Hiding "* , Journal of College of Education, Vol. 2, No.2 , Thi-Qar, Iraq, July 2010.