A PROPOSED ALGORITHM FOR STEGANOGRAPHY IN DIGITAL IMAGE BASED ON LEAST SIGNIFICANT BIT

BY

A. E.Mustafa
Prof of Fundamentals of Education
Faculty of Specific Education
Mansoura University

A.M.F.ElGamal
Assistant Prof of Computer Science
Faculty of Specific Education
Mansoura University

M.E.ElAlmi
Assistant Prof of Computer Science
Faculty of Specific Education
Mansoura University

Ahmed.BD
Demonstrator of Computer
Faculty of Specific Education
Mansoura University

Research Journal Specific Education
Faculty of Specific Education
Mansoura University

ISSUE NO. 21, APRIL. 2011

مجلة بحوث التربية النوعية – جامعة المنصورة
العدد الحادي والعشرون – أبريل ٢٠١١

A PROPOSED ALGORITHM FOR STEGANOGRAPHY IN DIGITAL IMAGE BASED ON LEAST SIGNIFICANT BIT

BY

A. E.Mustafa
Prof of Fundamentals of Education
Faculty of Specific Education
Mansoura University

A.M.F.ElGamal
Assistant Prof of Computer Science
Faculty of Specific Education
Mansoura University

M.E.ElAlmi
Assistant Prof of Computer Science
Faculty of Specific Education
Mansoura University

Ahmed.BD
Demonstrator of Computer
Faculty of Specific Education
Mansoura University

Abstract

Data hiding is the art of hiding data for various purposes such as; to maintain private data, secure confidential data and so on. There are lots of techniques used for data hiding and the well known technique is the Steganography. In contemporary terms, Steganography has evolved into a digital strategy of hiding a file in some form of multimedia, such as an image, an audio file or even a video file. This paper presents a new Steganography method based on the spatial domain for encoding extra information in an image by making small modifications to its pixels. The proposed method focuses on one particular popular technique, Least Significant Bit (LSB) Embedding. Instead of using the LSB-1 of the cover for embedding the message, LSB-2 has been used to increase the robustness. LSB-1 may be modified according to the bit of the message, to minimize the difference between the cover and the Stego-cover. For more protection to the message bits a Stego-Key has been used to permute the message bits before embedding it. Experimental results of the modified method shows that the Peak Signal to Noise Ration (PSNR) is grater than the conventional methods of LSBs replacement.

Keywords :

Steganography, Data hiding, Embedding Data, Information Security, Least Significant Bit.

A PROPOSED ALGORITHM FOR STEGANOGRAPHY IN DIGITAL IMAGE BASED ON LEAST SIGNIFICANT BIT

BY

A. E.Mustafa
Prof of Fundamentals of Education
Faculty of Specific Education
Mansoura University

A.M.F.ElGamal
Assistant Prof of Computer Science
Faculty of Specific Education
Mansoura University

M.E.ElAlmi
Assistant Prof of Computer Science
Faculty of Specific Education
Mansoura University

Ahmed.BD
Demonstrator of Computer
Faculty of Specific Education
Mansoura University

1. Introduction

The hiding of data is frequently called steganography .Steganography is a technology that hides a message within an object. Steganography plays an important role in information security [1, 2]. The origin of steganography is traced back to ancient civilizations. The ancient Egyptians communicated covertly using the hieroglyphic language, a series of symbols representing a message. The message looks as if it is a drawing of a picture although it may contain a hidden message. After the Egyptians, the Greeks used steganography, "hidden writing" where the name was derived [3]. The goal of steganography is to hide the fact that any form of communication is occurring by embedding messages into an innocuous-looking cover medium such as digital image, video, audio and so on, while steganalysis focus on revealing the presence of the secret messages and extract them [4-5] .

In general, steganography approaches hide a message in a cover e.g. text, image, audio file, etc., in such a way that is assumed to look innocent and there for would not raise suspicion. Fundamentally, the steganographic goal is not to hinder the adversary from decoding a hidden message, but to prevent an adversary from suspecting the existence of covert communications[3].

There are many ways (methods) to hide information in images. Any text, image, or anything that can be embedded in a bit stream can be hidden in an image. Image steganography has come quite far in recent years with the development of fast, powerful graphical computers. An image in a computer is an array of numbers that represent light intensities at various points (pixels). These pixels make up the image's raster data. Digital images are stored in either 24-bit (true color images) or 8-bit per pixel files. A common image size is 640 × 480 pixels and 256 colors (or 8 bits per pixel). Such an image could contain about 300 Kb of data . Such large size images should be avoided since the attention when sending over a network or the Internet. Hence 8-bit color images, like GIF files, can be used to hide information. Here, each pixel is represented as a single byte, and the pixel's value is between 0 and 255. Grey-scale images are preferred because the shades are changed very gradually between palette entries. This increases the image's ability to hide information .

The most well known techniques to data hiding in images are least significant bit (LSB) substitution, and masking & filtering techniques. LSB is a simple approach to embedding information in an image. But image manipulation can destroy the hidden information in this image. Applying LSB technique to each byte of a 24-bit image, three bits can be encoded into each pixel, as each pixel is represented by three bytes. Applying LSB technique to each byte of an 8-bit image, only one bit can be encoded into each pixel, as each pixel is represented by one byte[6].

2. Principles of Steganography

There are three categories of steganography :

pure steganography, secret key steganography, and public key steganography [2]. Pure steganography requires no prior exchange of information between the two parties communicating and relies on secret through obscurity. This means that the algorithms not publicly known, and therefore the level of testing is also unknown, making the tool unproven. One has to go on faith alone in those involved in the tool's creation to be assured covert communication. numerous instances of the false sense of security through obscurity can be cited [7].

Secret key steganography usual uses a publicly known algorithm, and relies on a secret key chosen beforehand by the two parties communicating. This key is needed to both embed and extract the hidden

information, and if the proper key is not used, it cannot be known if data is actually hidden in a given cover object [8]. If prior secure or, if desired, covert communications cannot be conducted to share the secret key before covert communications, another possibility is public key steganography. It entails the sender using the recipient's public key to embed the information, which can only be detected using the recipient's private key. This is analogous to how the public key infrastructure works in cryptography. The interesting characteristic with public key steganography is that even the sender should not be able to detect the secret message in the resulting stego object. As another alternative, proposes a steganographic key exchange protocol, where the communicating parties exchange a sequence of messages that look like normal communications, and at the end of the sequence each party is able to compute a shared key. This shred key can then be used for secret key steganography. No matter how it carried out, steganography is not useful if the existence of secret information can be proven by outside parties.[7-2]. Steganalysis is the method by which to detect the presence of a hidden message and attempt to reveal the true contents of this message. This technology has also substantially evolved throughout history [2].

3. Image Steganography Methods

Image steganography has been widely studied by researchers. There are a variety of methods used in which information can be hidden in images. In the following section, we present the most common methods. There are three common methods of steganography: Replacing Moderate Significant Bit, Transformation Domain Techniques, and Replacing Least Significant Bit. Replacing Moderate Significant Bit, Chan et al. showed how to use the moderate significant bits of each pixel in the cover image to embed the secret message. This method improves sensitivity to modification, but it degrades the quality of stego-image[9-10]. Other familiar data hiding techniques use the transformation domain of digital media to hide information discussed by Chang et al. and Hsu et al.. Functions such as the discrete cosine transform (DCT) and the discrete wavelet transform (DWT) are widely applied by Chang et al., and Hsu et al. These methods hide the messages in the significant areas of the cover image, which makes them robust against compression, cropping and other image processing attacks [9]. The last method is Replacing Least Significant Bit the concept of LSB Embedding is simple. It exploits the fact that the level of precision in many

image formats is far greater than that perceivable by average human vision. Therefore, an altered image with slight variations in its colors will be indistinguishable from the original by a human being, just by looking at it. By using the least significant bits of the pixels' color data to store the hidden message, the image itself will seem unaltered [11, 12].

4. Least Significant Bit (LSB-1) Replacement

This is the simplest of the steganography methods based in the use of LSB, and therefore the most vulnerable. The embedding process consists of the sequential substitution of each Least Significant Bit (LSB-1) of the image pixel for the bit message. For its simplicity, this method can camouflage a great volume of information. The following steps illustrate how this method is used to hide the secret data "A" in cover image " Mansoura.bmp ".

1st step : Convert the data from decimal to binary.

[Message] —Dec2Bin→ [1000001]

2nd step : Read Cover Image "Mansoura.bmp" as shown in figure 1:

144	**142**	**146**	**152**	**156**	**147**	**151**	**157**
160	**155**	**159**	**162**	**133**	**123**	**133**	**145**
144	**141**	**141**	**138**	**61**	**55**	**65**	**79**
120	**123**	**131**	**144**	**50**	**61**	**74**	**92**
170	**167**	**167**	**166**	**61**	**59**	**56**	**59**
120	**125**	**131**	**132**	**61**	**59**	**59**	**59**
124	**133**	**139**	**131**	**88**	**76**	**77**	**76**
138	**153**	**167**	**154**	**139**	**…..**	**…..**	**…..**

Figure 1 : The cover image " Mansoura.bmp"

3Th step : Convert the Cover Image from decimal to binary.

10010000	**10011010**	**10011100**	**10010010**	**10010110**	**10011101**	**10101111**	**10100101**
10100000	**10011011**	**10011111**	**10100010**	**10000101**	**01111011**	**10000101**	**10010001**
10010000	**10001101**	**10001101**	**10001010**	**00111101**	**00110111**	**01000001**	**01001111**
01111000	**01111011**	**10000011**	**10010000**	**00110010**	**00111101**	**01001010**	**01011100**
10101010	**10100111**	**10100111**	**10100110**	**00111101**	**00111011**	**00111000**	**00111011**
01111000	**01111101**	**10000011**	**10000100**	**00111101**	**00111011**	**00111011**	**00111011**
01111100	**10000101**	**10000111**	**10000011**	**01011000**	**01001100**	**01001101**	**01001100**
10001010	**10011001**	**10100111**	**10011010**	**10001011**	**…….....**	**…….....**	**…….....**

4Th step : Break the byte to be hidden into bits.

Thus [10000001] is divided into 8 bits → [1 0 0 0 0 0 0 1].

5Th step : Take first 8 byte of original data from the Cover Image .

10010000	**10011010**	**10011100**	**10010010**	**10010110**	**10011101**	**10101111**	**10100101**

6Th step : Replace the least significant bit by one bit of the data to be hidden.

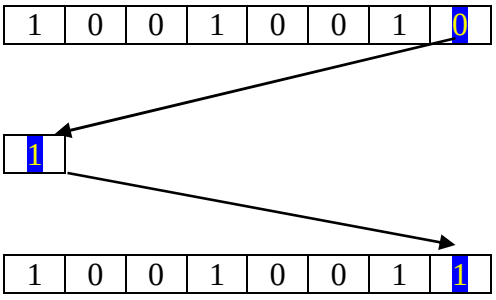
- First byte of original data from the Cover Image :

1	0	0	1	0	0	0	0

- First bit of the data to be hidden :

1

- Replace the least significant bit :



- Repeat the replace for all bytes of Cover Image :
- Finally the cover image before and after steganography is shown in figure 2.

Cover Image before steganography

Cover Image after steganography

Figure 2: The cover image before and after steganography

5. The Proposed Method

In this method, a 256*256 color image has been used as a cover. So, we can hide a message up to 65536 bytes. The message is embedded in the LSB-2 of the cover to increase the robustness of the system and protect the message against the external influences such as noise, filter, compression,…etc. The embedding process is very easy, which only replaces the permutated bits of the message (M) by the LSB-2 set of the cover to obtain the new stego-image $Z=\{new\_pixe_{l0}, new\_pixe_{l1},..,new\_pixel_{65535}\}$. To minimize the difference between the old value (pixel) in the cover and the new value (new_pixel) in the stego-image, we suggest the following embedding algorithm:

Embedding Algorithm:

Step 1: Extract Bit set of Message , Bit={M_0,M_1,……, M_{65535} }
Step 2: The Pixels of cover image , Pixel ={$pixel_0$, $pixe_l$,…, $pixel_{65535}$}
Step 3: **Extract LSB-1 set of the cover image, LSB1={A_0, A_1,…,A_{65535}}.**
Step 4: **Extract LSB-2 set of the cover image, LSB2={B_0, B_1,…, B_{65535}}.**
Step 5:

```
For i=1 to message length do
{
        If Mi= =Bi Then
                do nothing
        Else
        {
                If Mi= =1 and Bi= =0 Then
                {
                        Bi=Mi;
                        Ai=0;
                        Pixel(i)-=1
                }
                Else If Mi= =0 and Bi= =1 Then
                {
                        Bi=Mi;
                        Ai=1;
                        Pixel(i)+=1
                }
        }
}
```

6. Experimental results and discussions

To apply the proposed algorithm, consider that we have to hide the secret data "A" in cover image " Mansoura.jpg " :

1st step : Convert the data from decimal to binary.

[Message] Dec2Bin → [1000001]

2nd step : Read cover image "Mansoura"

144	142	146	152	156	147	151	157
160	155	159	162	133	123	133	145
144	141	141	138	61	55	65	79
120	123	131	144	50	61	74	92
170	167	167	166	61	59	56	59
120	125	131	132	61	59	59	59
124	133	139	131	88	76	77	76
138	153	167	154	139	…..	…..	…..

3Th step : Convert the cover image from decimal to binary.

10010000	10011010	10011100	10010010	10010110	10011101	10101111	10100101
10100000	10011011	10011111	10100010	10000101	01111011	10000101	10010001
10010000	10001101	10001101	10001010	00111101	00110111	01000001	01001111
01111000	01111011	10000011	10010000	00110010	00111101	01001010	01011100
10101010	10100111	10100111	10100110	00111101	00111011	00111000	00111011
01111000	01111101	10000011	10000100	00111101	00111011	00111011	00111011
01111100	10000101	10000111	10000011	01011000	01001100	01001101	01001100
10001010	10011001	10100111	10011010	10001011	…….....	…….....	…….....

4Th step : Break the byte to be hidden into bits.

Thus [10000001] —is divided into 8 bits→ [1 0 0 0 0 0 0 1].

5Th step : Take first 8 byte of original data from the cover image .

10010000	10011010	10011100	10010010	10010110	10011101	10101111	10100101

6Th step : Replace LSB2 by one bit of the data to be hidden.

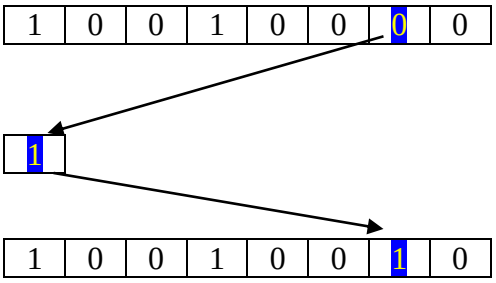
- First byte of original data from the cover image is:

1	0	0	1	0	0	0	0

- First bit of the data to be hidden is:

1

- Replace the LSB2:



- In our proposed method if the bit of the data to be hidden = = 1 and LSB2= =0 then

1-we change LSB1 of image to 0 after replacement.

1	0	0	1	0	0	1	0

2-we subtract 1 .

1	0	0	1	0	0	0	0

So we have no change in cover image

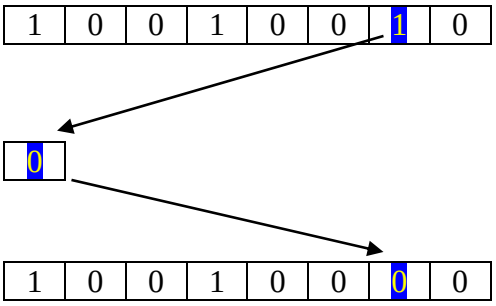
- Second byte of original data from the cover image :

1	0	0	1	0	0	1	0

- Second bit of the data to be hidden :

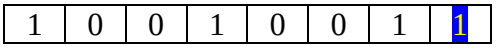
0

- Replace the LSB2 :

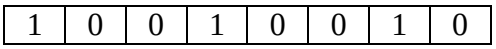


In our Proposed Method if the bit of the data to be hidden = = 0 and LSB2= =1 then

1-we change LSB1 of image to 1 after replacement.



2-we increase 1 .



So we have No change in cover image

- Repeat the replace for all bytes of cover image.

The cover image before and after applying the proposed steganography is shown in figure 3.

Cover image before steganography

Cover image after steganography

Figure 3: The cover image before and after applying the proposed steganography

Thus, the difference in LSB-2 replacement no change in cover image or less or equal one as in LSB-1 but in more robust. The comparison between the LSB method and the proposed method is shown in table 1. The experimental results demonstrate that the proposed method uses a multilevel hiding strategy to achieve large hiding capacity and keep distortion low.

Table1: the comparison between the LSB method and the proposed method

Moment	LSB Method		Proposed Method	
	Before	After	Before	After
Mean	402.195	402.188	402.195	402.187
Standard eviation	32798.4	32798.8	32798.4	32798.7
Median	417	417	417	417
Kurtosis	5.0284	5.02831	5.0284	5.02836
Skewness	-0.892541	-0.892626	-0.892541	-0.89256
MSE	**0.400061264**		**0.300106651**	
PSNR	**52.10953858**		**53.35804740**	

8. Conclusion

The enhanced LSB technique described in this paper helps to successfully hide the secret data into the cover file with minimum distortion made to the cover file. This method are essential for construction of accurate targeted and blind steganalysis methods for JPEG, BMP and PNG images. In this paper we have identified the use the concept of LSB2 to hide the given text into the cover. The most commonly used technique, the least significant bit technique causes higher distortion to the cover file in many cases. Experimental results of the modified method shows that PSNR is grater than the conventional method of LSBs replacement.

References

1. Mei-Yi, W., Yu-Kun, H. , Jia-Hong, L. (2004): An Iterative Method of Palette-Based Image Steganography, **Journal of Patern Recognition Letters**, Vol (25).

2. Alain, C. Brainos (), A Study Of Steganography And The Art Of Hiding Information, East Carolina University.

3. Desoky, A. (2009):A novel Noiseless Steganography Paradigm, **Ph.D**, Department of Computer Science and Electrical Engineering, Faculty of the Graduate School, University of Maryland, Baltimore County.

4. Christopher, T. (2007):Compression Aided feature Basedsteganalysis of Perturbed Quantization Steganography in Jpeg image **,Ms.C** s, Department of Science in Electrical and Computer Engineering, University of Delaware.

5. Xiang-yang, L. , Dao-shun,W., Ping, W., Fen-lin, L.((2008): A review on Blind Detection for Image Steganography, **Journal of Signal Processing,** Vol(88),Issue(9).

6. Samer, A.(2006):A New Algorithm for Hiding Gray Images using Blocks,Information , Security Journal, **The Hashemite University**, Jordan, Volume (15), Issue (6).

7. Gerad, G.(2006) : An Investigation of Scalable Vector Graphics as Cover Medium for Steganography, **Ms.C**, faculty of college of arts and science, American University.

8. Kaushal M. Solanki, 2005, Multimedia Data Hiding:From Fundamental Issues to Practical Techniques, **Ph.D**, Electrical and Computer Engineering, university of california, Santa Barbara.

9. Sanjeev, M.,et.al (2008): Customized and Secure Image Steganography**, Journal of Signal Processing ,** Vol(1), Issue (1).

10. Hengfu, Y., Xingming S., Guang S.,(2009): A High-Capacity Image Data Hiding Scheme Using Adaptive LSB Substitution**, Journal of radio engineering,** VOL. (18), NO. (4).

11. Lee, L.(2004) : LSB Steganography :Information Within Information**, Journal of Computer Science**,Vol (265), No (5).

12. Chi-Kwong,C., Cheng, L.(2004): Hiding data in images by simple LSB substitution, **Journal 0f Pattern Recognition**, Vol (37).

خوارزم مقترح لإخفاء البيانات في الصور الرقمية قائم على البت الأقل أهمية

A. E.Mustafa
Prof of Fundamentals of Education
Faculty of Specific Education
Mansoura University

A.M.F.ElGamal
Assistant Prof of Computer Science
Faculty of Specific Education
Mansoura University

M.E.ElAlmi
Assistant Prof of Computer Science
Faculty of Specific Education
Mansoura University

Ahmed.BD
Demonstrator of Computer
Faculty of Specific Education
Mansoura University

الملخص

علم إخفاء البيانات هو فن لإخفاء البيانات السرية لأغراض متعددة , منها على سبيل المثال الحفاظ على البيانات الخاصة ، وتأمين البيانات السرية ....الخ , وهناك الكثير من التقنيات المستخدمة لإخفاء البيانات ومن أشهرها تقنية إخفاء المعلومات Steganography . و تعتمد تقنية إخفاء المعلومات على إخفاء البيانات السرية داخل ناقل من الوسائط المتعددة مثل الصور الرقمية والملفات الصوتية وملفات الفيديو . الهدف الرئيسي في هذا البحث هو تحليل واختبار الطريقة التقليدية لإخفاء البيانات ثم تقديم تقنية مقترحة جديدة لإخفاء البيانات السرية في المجال المكاني لغطاء الصورة. التقنية المقترحة تعتمد على أشهر الأساليب استخداما , وهو أسلوب البت الأقل أهمية LSB , ولكن يتم تخزين ثنائيات الرسالة في الطبقة الثانية (LSB-2) من ثنائيات الصورة بدلا من الطبقة الأولى LSB-1 , وذلك لزيادة متانة بيانات الرسالة داخل الغطاء. ولمزيد من الحماية لبيانات الرسالة قمنا ببعثرتها باستخدام مفتاح إخفاء قبل أن يتم إخفائها في الغطاء. وقد أثبتت النتائج أن الطريقة المقترحة أعطت قيمة تشابه (PSNR) بين الغطاء قبل وبعد الإخفاء أكثر من قيمة التشابه في الطريقة التقليدية للإخفاء.