A New Algorithm for Embedding Message in Image Steganography

Mohammad Ali Shamalizadeh Baei (PhD student),
Zeynolabedi Norozi
Faculty of Security and Cryptography
Imam Hossein University
Tehran, Iran

Mohammad Reza Karami Mollaei
Faculty of Electronic and Computer
Babol University of Technology,
Babol, Iran

***Abstract*— Steganography is the art of hiding information and an effort to conceal the existence of the embedded information. Paying no attention to file formats, image steganography is discussed in this paper. Least Significant Bit (LSB) insertion is a common, simple approach to embedding information in a cover image, but image manipulation such as noise, filtering, compression…etc, can destroy the hidden information in this image. The focus of this paper is designing a new algorithm for embedding message bites in an image file with more robustness against the external influences and more security. In this paper instead of using the first least significant bit or LSB of the cover image pixels for embedding the message bit, second least significant bit has been used to increase the robustness and SIHS[1] method is applied to produce more secured system that we call it RSIHS[2] technique.**

Keywords— Image Steganography, LSB2, SIHS, RSIHS

I. INTRODUCTION

Since the rise of the internet, one of the most important factors of information technology communication has been the security of information. Cryptography was created as a technique for securing the secrecy of communication and many different methods have been developed to encrypt and decrypt data in order to keep the message secret. Unfortunately it is sometimes not enough to keep the contents of a message secret, it may also be necessary to keep existence of the message secret. The technique used to implement this is called Steganography. Steganography is the art and science of invisible communication. Throughout history, Steganography has been used to secretly communicate information between people. Steganography plays an important role in information security [1, 2]. Steganography is a technology that hides a message within an object. Today‘s steganographic systems uses multimedia objects like image, audio, video etc., as cover media because people often transmit digital pictures over email and other Internet communication [3]. In modern approach, depending on the nature of cover object, Steganography can be divided into five types, Audio, Video, Text, Image and Protocol steganography.(See figure 1)An image is an array of M*N matrix. Each pixel has a numerical value which represents the color and light intensity of the pixel [4]. Images are more popular cover files for transmission over the internet due to harmlessness and attraction. In image steganography, data is to be inserted into the cover image that gives the resultant stego-image [5]. A possible formula of the process may be represented as:

Cover image + embedded message + stego key = stego image (See figure 2)

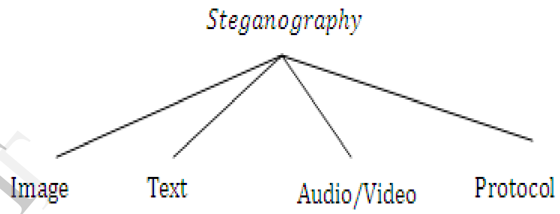
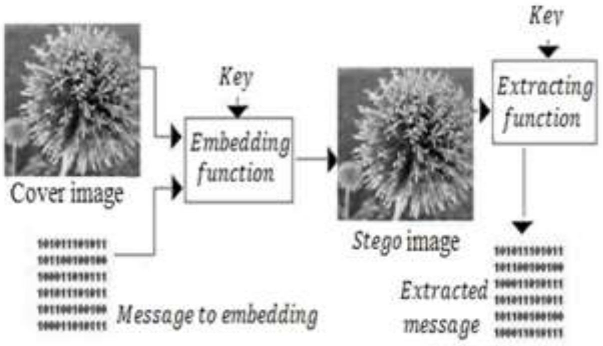


Figure 1: Type of Steganography



Figuer 2: Steganography procedure

There are various image steganographic [6], techniques:

1. Spatial domain technique.
2. Masking and filtering.
3. Transform techniques.

Spatial domain technique embeds the message directly in the intensity of the pixels. If message bit is embedded in least significant bit of each image pixel is called LSB or technique. A. E. Mustafa and et al, [7], for first time proposed an algorithm for embedding message bits in cover image pixels that called SIHS technique. In this method some of message bits are embedded in second least significant bite of cover image pixels. Jassim Mohammed Ahmed and

[1] Secure Information Hiding System
[2] Robustness Secure Information Hiding System

ZulkarnainMd Ali, [8], by using LSB1 technique, a stego-key and a random number generator, presented a secure system that is called LSB2 technique. In this paper we suggest a new method of embedding message bites, by improvement of technique and use of SHIS technique with considering robustness and security that we call RSIHS it technique.

II. THE LSB1 TECHNIQUE

Least significant bit (LSB) insertion is a common and simple approach to embedding message bits in cover image pixels. This technique works well for image steganography.

A. *Algorithm of LSB1*

Algorithm to embed text message using Grayscale Image
Step1: Read the cover image and text message, which is to be hidden in the cover image.
Step 2: Convert text message into binary.
Step 3: Calculate LSB1 of cover image pixels.
Step 4: Replace LSB1 of cover image pixels with each bit of secret message one by one.
Step 5: Write stego image.

Note that, on average, only half the LSB1 need to change. The difference between the cover (i.e. original) image and the stego image will be hardly noticeable to the human eye. Because the change on light intensity of each cover image pixel happens 0 or ±1 unit.

III. THE SIHS (SECURE INFORMATION HIDING SYSTEM) TECHNIQUE

As the LSB1 technique is used to hide the messages in images. However, it is decided to enhance the security system by introducing a new technique comprises of randomly dispersing the message bits in images. It is proposed in this enhancement that the embedding of message bits into the image is not only in the least significant bit but also the other bits in the pixel in the random manner. This can be achieved by comparing the message bit to the pixel bit randomly chosen from second to the last bit, r, which we produce with discrete logarithm algorithm or other technique. Based on this comparison, 1 is inserted in the least significant bit if the message bit identical to that of the image, whereas, 0 is inserted if the message bit didn't match with the chosen bit from the image(see figure 3)

We see that this algorithm uses LSB1 for embedding message bits but because of using random technique for coding original message bits it has more secure than LSB1.

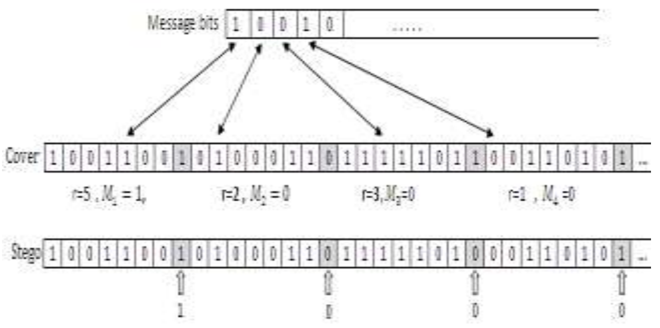


Figure 3: Embedding message bits in cover image pixels with *SIHS* algorithm

A. *Discrete logarithm algorithm and random numbers*

Discrete Logarithm calculation can be used to solve the sequence-mapping problem. The main idea here is to generate a series of random numbers of length equal to the message length, N, that ranging from 3 to 8. These series Numbers will be used in random-mapping. We defined discrete logarithm to produce random numbers. These numbers depend on the value of key (k). The values are computed from the following equation, and these numbers will be limited to the length of the message: x_i=a*x$_{(i-1)}$ (mod p)   (1), i=1,…,m.

Where,

x_0= is the sum of k digits.

a=3x_0

p=K

The numbers created from the above equation is then used to generate another numbers ranging from 3 to 8.The latter are used to locate the image bit (in the pixel) that will be used in the comparison with the message bit, as

Expressed as follow:

$$r = p_i = x_i(mod\ 7) + 2 \qquad (2)$$

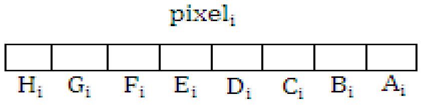
Recovering a message from a steg-image demands the corresponding decoding key, k that used during the encoding process. Hence, both the sender and receiver have to share the stego-key during the communication. The k key is then used for selecting the positions of the pixel where the secret bits had been embedded. For example consider message bits and cover image in figure4 and random numbers is generated with a secret key, k, and discrete logarithm algorithm.

IV. THE LSB2 TECHNIQUE

This technique is an enhancement of LSB1 algorithm. In this method, some of message bites are embedded in the second least significant bites of the cover image pixels to increase the robustness of the system and protect the message against the external influences such as noise, filter, compression…etc.

A. *Algorithm of LSB2*

Algorithm to embed text message using Grayscale Image is as follows:
Step1:Extract bit set of message. Bit={M_1,M_2,…,M_N}
Step 2: The pixels of cover image.
Pixel ={Pixel_1,Pixel_2,…,Pixel_N}

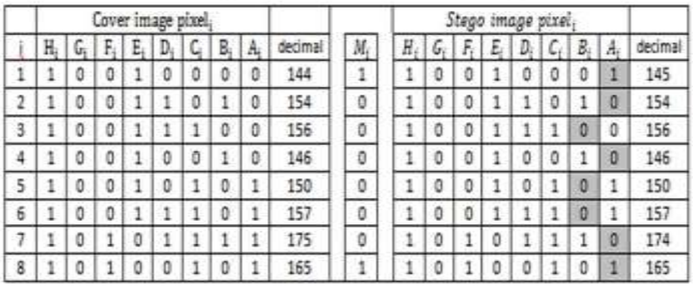


Step3:Extract LSB1 set of the cover image. LSB1={A_1,A_2,…,A_N}
Step 4: Extract LSB2 set of the cover image.
LSB2={B_1,B_2,…,B_N}
Step 5:

```
For i =1   to message length do
          {
 If ($M_i$ == $B_i$) Then
          do nothing
 Else
          {
  If ($M_i$ = = 1 and $B_i$ = = 0) Then
```

```
        {
     $B_i = M_i$;
    $A_i$=0;
    $pixel_i$=$Pixel_i$-1;
            }
 Else If ($M_i$ = = 0 and $B_i$ = = 1)Then
        {
     $B_i = M_i$;
    $A_i$=1;
     $pixel_i$=$Pixel_i$ +1;
            }
    }
```

Table1. Experimental result of LSB2 technique.



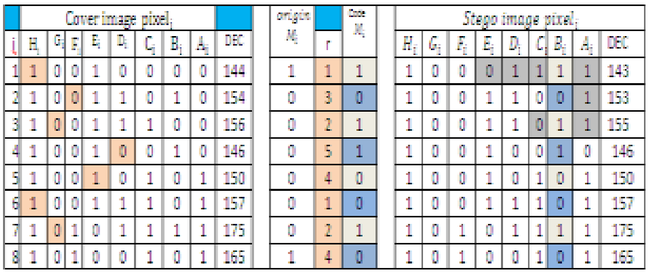
	Cover image $pixel_i$										Stego image $pixel_i$								
i	H_i	G_i	F_i	E_i	D_i	C_i	B_i	A_i	decimal	M_i	H_i	G_i	F_i	E_i	D_i	C_i	B_i	A_i	decimal
1	1	0	0	1	0	0	0	0	144	1	1	0	0	1	0	0	0	1	145
2	1	0	0	1	1	0	1	0	154	0	1	0	0	1	1	0	1	0	154
3	1	0	0	1	1	1	0	0	156	0	1	0	0	1	1	1	0	0	156
4	1	0	0	1	0	0	1	0	146	0	1	0	0	1	0	0	1	0	146
5	1	0	0	1	0	1	0	1	150	0	1	0	0	1	0	1	0	1	150
6	1	0	0	1	1	1	0	1	157	0	1	0	0	1	1	1	0	1	157
7	1	0	1	0	1	1	1	1	175	0	1	0	1	0	1	1	1	0	174
8	1	0	1	0	0	1	0	1	165	1	1	0	1	0	0	1	0	1	165

To apply this algorithm, consider that we have to hide the message bites 10000001 in a cover image that its primary pixels are shown in Table1. By using above algorithm and record result in Table1, we see that the LSB2 technique embeds some of message bits in second least significant bit of cover image pixels to increase robustness.

V. THE PROPOSED TECHNIQUE

The LSB1 is the simplest and most straight forward approach to embed or hide a message into a cover image and hides the message in a way that the humans do not distinguish it. On the other hand embedded message based on LSB1technique easily destroy with a noise or operation so compression and etc. Therefore, there is a need to enhance theLSB1. That's obvious that SIHS method is based on LSB1 technique. So here we design a new technique of embedding message based on LSB2 and SIHS which I call such a mechanism RSIHS. In this method we suggest an algorithm that embeds all of the message bits after coding with SIHS technique into second least significant bits of cover image pixel, so that the change on light intensity of each cover image pixel happens 0 or ±1 unit.

Table 2:Embeding message based on RSIHS technique



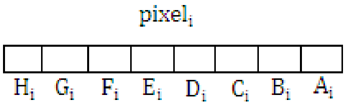
	Cover image $pixel_i$									origin M_i		Code M_i	Stego image $pixel_i$								
i	H_i	G_i	F_i	E_i	D_i	C_i	B_i	A_i	DEC		r		H_i	G_i	F_i	E_i	D_i	C_i	B_i	A_i	DEC
1	1	0	0	1	0	0	0	0	144	1	1	1	1	0	0	0	1	1	1	1	143
2	1	0	0	1	1	0	1	0	154	0	3	0	1	0	0	1	1	0	0	1	153
3	1	0	0	1	1	1	0	0	156	0	2	1	1	0	0	1	1	0	1	1	155
4	1	0	0	1	0	0	1	0	146	0	5	1	1	0	0	1	0	0	1	0	146
5	1	0	0	1	0	1	0	1	150	0	4	0	1	0	0	1	0	1	0	1	150
6	1	0	0	1	1	1	0	1	157	0	1	0	1	0	0	1	1	1	0	1	157
7	1	0	1	0	1	1	1	1	175	0	2	1	1	0	1	0	1	1	1	1	175
8	1	0	1	0	0	1	0	1	165	1	4	0	1	0	1	0	0	1	0	1	165

A. *Embedding algorithm(RSIHS algorithm)*

In this method, a N=h×w gray scale image has been used as a cover. So, we can hide a message up to bits. We suggest the following embedding algorithm:

Step 1: Extract Bit set of Message.
Bit={M_1,M_2,…,M_N}

Step2:The Pixels of cover image.
Pixel ={$Pixel_1$,$Pixel_2$,…,$Pixel_N$}



Step3: Extract LSB1 set of the cover image.

LSB1= {A_1,A_2,…,A_N}.

Step4: Extract LSB2 set of the cover image.

LSB2= {B_1,B_2,…,B_N}.

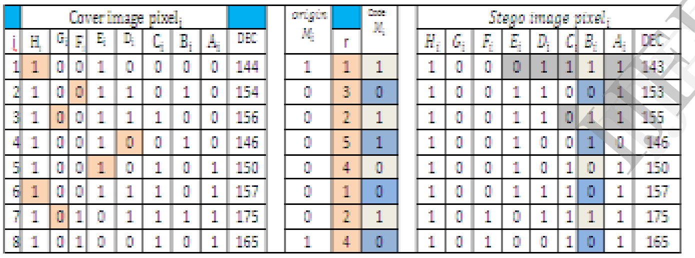
Step5: for i=1 to message length do

```
{
Generate a random integer number, $p_i$, with discrete logarithm algorithm. 〖(3≤$p_i$=r≤8)
If ($r^{th}$ bit of $pixel_i$ == $M_i$) then $M_i$=1;
Else Mi=0;
if ($B_i$== $M_i$ )  Then
do nothing
Else If ($M_i$==0 and $B_i$==1)
{
If ($A_i$=0)
{  $B_i$= 0;  $A_i$=1; }
Else If ($A_i$=1)  {
  If ($C_i$=0)
{$C_i$= 1;  $B_i$ =$A_i$=0 ;}
     Else If ($D_i$=0)
{$D_i$= 1;   $C_i$=$B_i$ =$A_i$=0 ;}
     Else If ($E_i$=0)
{$E_i$=1; $D_i$= $C_i$=$B_i$=$A_i$=0 ;}
     Else If ($F_i$=0)
{$F_i$=1; $E_i$= $D_i$= $C_i$=$B_i$=$A_i$=0 ; }
     Else If ($G_i$=0)  { $G_i$=1; $F_i$=$E_i$ =$D_i$=$C_i$=$B_i$=$A_i$=0 ;}
    Else If (H_i=0)
{H_i=1 ; $G_i$ = $F_i$ =$E_i$ =$D_i$=$C_i$=$B_i$=$A_i$=0 ;}
   }
 }
Else  If ($M_i$==1 and $B_i$==0)
    {
```

```
If   ($A_i$=1){   $B_i$= 1;  $A_i$==0; }
    Else If   ($A_i$=0){
If ($C_i$=1)
{$C_i$= 0; $B_i$=$A_i$=1 ;}
             Else If ($D_i$=1)
{$D_i$=0;   $C_i$= $B_i$=$A_i$=1;}
            Else If ($E_i$=1)
{$E_i$=0; $D_i$=$C_i$= $B_i$=$A_i$=1;}
            Else If ($F_i$=1) { $F_i$=0; $E_i$ =$D_i$=$C_i$=$B_i$=$A_i$=1 =1 ;}
             Else If ($G_i$=1) { $G_i$ =0; $F_i$=$E_i$ =$D_i$=$C_i$=$B_i$=$A_i$=1;}
            Else If ($H_i$=1){$H_i$=0 ;$G_i$=$F_i$=$E_i$ =$D_i$=$C_i$=$B_i$=$A_i$=1 ;}
                 }
         }
}
```

To apply this algorithm, again consider same data in table 1, namely, message bites 10000001 for embedding into a cover image that its primary light intensity pixels are shown in table 1. By using this algorithm and record results in table2.We have:

Table 2:Embeding message based on RSIHS technique



	Cover image pixel$_i$									origin M_i		Code M_i	Stego image pixel$_i$								
i	H_i	G_i	F_i	E_i	D_i	C_i	B_i	A_i	DEC		r		H_i	G_i	F_i	E_i	D_i	C_i	B_i	A_i	DEC
1	1	0	0	1	0	0	0	0	144	1	1	1	1	0	0	0	1	1	1	1	143
2	1	0	0	1	1	0	1	0	154	0	3	0	1	0	0	1	1	0	0	1	153
3	1	0	0	1	1	1	0	0	156	0	2	1	1	0	0	1	1	0	1	1	155
4	1	0	0	1	0	0	1	0	146	0	5	1	1	0	0	1	0	0	1	0	146
5	1	0	0	1	0	1	0	1	150	0	4	0	1	0	0	1	0	1	0	1	150
6	1	0	0	1	1	1	0	1	157	0	1	0	1	0	0	1	1	1	0	1	157
7	1	0	1	0	1	1	1	1	175	0	2	1	1	0	1	0	1	1	1	1	175
8	1	0	1	0	0	1	0	1	165	1	4	0	1	0	1	0	0	1	0	1	165

Table2 Shows that RSIHS technique embeds all of the message bits into second least significant bit of cover image pixels and changes light intensity of each cover image pixels like LSB1 as much as 0 or ±1 unit. So this technique must be robustness of others.

VI. EXPREMINAL RESULTS

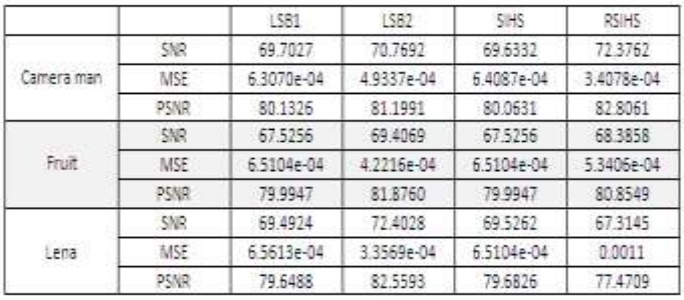
Experimental results evaluate the performance by hiding the data in LSB1, LSB2, SHIS and RSHIS method in gray scale image using MATLAB. Various performance parameters like SNR, PSNR and MSE have been used to evaluate the performance. The experiment has been taken out on figures 4,5 and 6 by embedding a message which includes ASCII characters. Table3 shows SNR, PSNR and MSE of each method for every algorithm.

Fig 4. Camera man

Fig 5. Fruit

Fig 6. Lena

Table 3: Exprimental results on 3 images



		LSB1	LSB2	SIHS	RSIHS
Camera man	SNR	69.7027	70.7692	69.6332	72.3762
	MSE	6.3070e-04	4.9337e-04	6.4087e-04	3.4078e-04
	PSNR	80.1326	81.1991	80.0631	82.8061
Fruit	SNR	67.5256	69.4069	67.5256	68.3858
	MSE	6.5104e-04	4.2216e-04	6.5104e-04	5.3406e-04
	PSNR	79.9947	81.8760	79.9947	80.8549
Lena	SNR	69.4924	72.4028	69.5262	67.3145
	MSE	6.5613e-04	3.3569e-04	6.5104e-04	0.0011
	PSNR	79.6488	82.5593	79.6826	77.4709

VII. CONCLUSION

In this paper we have presented an enhancement of the image steganography system using LSB approach to provide a more robustness and secure communication. In our proposed approach, all of the message bits are embedded randomly, by discrete logarithm algorithm, into the second least significant bit of cover-image pixels instead of first least significant bit. So the enhanced LSB technique namely RSISH algorithm, described in this paper helps us to successfully hide the secret data into the cover file with minimum distortion made to the cover file. Experimental results of the modified method (see Table.3.) show that, for this more robustness and secure method, SNR, MSE and PSNR are close to the other conventional method of LSB replacement or SIHS technique.

REFERENCES

[1] Mei-Yi, W., Yu-Kun, H., Jia-Hong. L. “An iterative method of palette-based image steganography”,*Journal of pattern recognition letters*, 2004, Issue 3, Vole (25), pp.301-309.

[2] Arvind Kumar**,** Km. Pooja. Steganography: “A data hiding technique”, *International journal of computer applications*, 2010, Vole (9), No.7, pp. 19-23.

[3] Cole, Eric. “Hiding in plain Sight: Steganography and the art of covert communication”, *Wiley*, 2003, Indianapolis.

[4] Gonzales, R. C., and R.E. Woods, “Digital image processing”, *Prentice Hall*, Inc., New Jersey, 2^{nd}, edition, 2002.

[5] Jagvinderkaur and sanjeevkumar. “Study and analysis of various image steganography techniques”, *International journal of computer science and technology*, 2011, Issue3, Vole (2), ISSN:2229-4333(Print)|ISSN:0976- 491(Online).

[6] Kavehahmadi. “A new method for image security and data hiding in image”, American *Journal of Scientific Research*, 2011, ISSN:1450-223X, Issue 38, pp. 41-49.

[7] A.E.Mustafa, A.M.F.ElGamal, M.E.ElAlmi and Ahmed.BD. “A proposed algorithm for steganography in digital image based on least significant bit”,*Research journal specific education faculty of specific education*, Mansoura University, 2011, Issue No. 21, pp. 751-766.

[8] Jassim Mohammed Ahmed and ZulkarnainMd Ali. “Information hiding using LSB technique”, *International journal of computer science and network security*, 2011, Vole (11), No.4, pp. 18-25.