# Information Hiding using Linear Recursion

Ruchika Sharma
Assistant Professor, JaganNath Institute Of Management Sciences,
JIMS, New Delhi-110085
E-mail: msgruchi2@gmail.com, ruchika.sharma@jimsindia.org

Dr.Vinay Kumar
Ex Scientist, GOI and Ex Dean & Professor, VIPS, GGSIPU, Delhi, India
E-mail: vinay5861@gmail.com

**Abstract**

Preserving the privacy of information has been always a challenge to the researcher community. Since the advent of Information Communication Technology, it has become even more challenging to maintain the privacy of information while transferring it over the public network. Many techniques have been developed and are being used for the purpose. Onlookers are also working side by side to break the techniques to breach privacy. Encryption provides a level of security however, it is not meant for hiding the very communication channel. Information hiding, also known as Steganography, helps in hiding the communication channel itself by placing the information in other digital media. How randomly the place for hiding a single bit of message is selected in digital media to hide the message, determines the strength of steganography. This paper proposes an information hiding approach based on a recursive equation to randomize the selection of pixels in digital media while hiding the information. We have used a 24 bit BMP image as cover digital media. We also analyzed the strength of the approach against the three parameters: Perceptibility, Robustness and Capacity.

**Keywords** - Steganography, BMP file, linear recursion, steganalysis, Information hiding, perceptibility, robustness, capacity.



## 1. INTRODUCTION

Information Hiding is the process of hiding secret information in an appropriate carrier [11, 13]. A carrier can be any digital file e.g., audio, video, image, text, etc., [10, 12, 14]. To hide information, Steganography is used. Steganography is a process of embedding secret information in digital media [1, 2, 8, 9]. This paper intends to generate pixel locations in the BMP image file to hide secret information.

The goal of steganography is to hide the very fact that communication is taking place. Steganography can be classified in two ways :

### 1.1 Spatial Domain Steganography

Spatial domain steganography works directly on pixel values[16].In Spatial Domain Steganography, secret information directly embeds in pixel values. The flowchart of spatial domain steganography is shown in Fig. 1

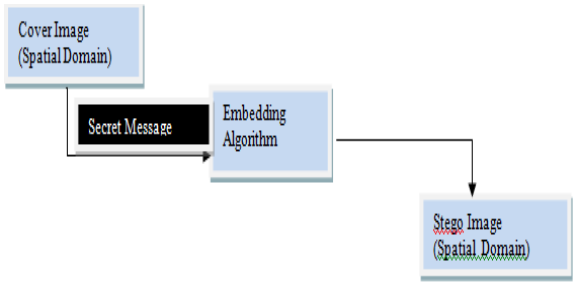


Fig.1 Representation of Spatial Domain Steganography

Least Significant bit(LSB) substitution is the most basic example of spatial domain steganography.

### 1.2 Frequency Domain Steganography

In Frequency domain steganography, images are first converted into the frequency domain and then secret information is embedded in their transformed coefficients [17]. The flowchart of spatial domain steganography is shown in Fig. 2.

Fig.2 Representation of Frequency Domain Steganography

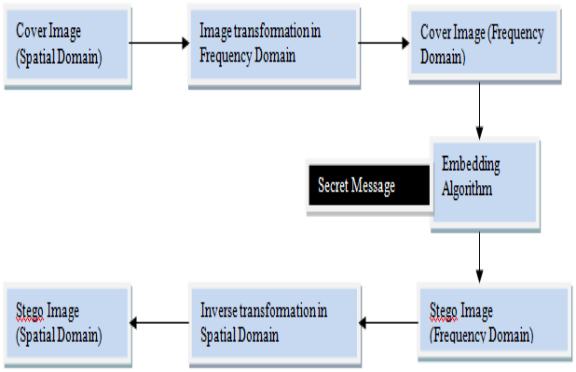


Fig.2 Representation of Frequency Domain Steganography

Discrete Cosine Transform (DCT) and discrete wavelet transform (DWT) are examples of the Frequency domain steganography technique.

This paper proposes a new way of hiding information in digital cover using Linear Recursive approach. The main advantage of using the Linear Recursive approach is its Robust feature. In the proposed approach, the bit places are randomized to hide secret information, so it becomes more difficult for any intruder to detect the very presence of secret information.

The paper is organized into 7 sections. Section 2 contains a theoretical description of the Linear recursive approach. Section 3 contains a description of bitmap file format. In Section 4, the conceptual approach of Linear Recursion is explained with an algorithm to find randomize bit locations to hide secret information is generated. Section 5 explains the implementation process of the proposed approach. In section 6, Steganalysis is done to check the strength of embedded algorithm against three parameters capacity, robustness and perceptibility. The paper is concluded in Section 7.

## 2. LINEAR RECURSIVE APPROACH

Recursion is the process of repeating items in a self-similar way. The most common application of recursion is in computer science, in which it refers to a method of defining functions in which the function being defined is applied within its own definition [7].

When the Linear Recursive approach is used to hide information, we interpret it as a "Recursive Approach to Data Hiding". Linear Recursive approach is used to send secret messages embedded inside digital cover like pictures, videos, etc.

Using the linear recursive approach, one may find places (bit positions) where secret information can be hidden. The recursive approach is used to find the location where we can make changes in bit positions. We find places where information can be changed and later we can retrieve the information by using the same recursive approach.

## 3. BMP FILE FORMAT

BitMap files are uncompressed; hence they are large in size. The advantage is that they are simple and got wide acceptance in Windows programs.

Bitmap files are structured in four parts:

**3.1.** The bitmap file header i.e., the size of the bitmap file

**3.2.** The bitmap information header i.e., the color format, compression if used and the file dimensions.

**3.3**..The bitmap color table i.e., which contains details of all the colors present in the bitmap

**3.4**. The bitmap document array, which contains the actual bits of the bitmap image in a consecutive line-by-line pixel array [2].

Bitmap file formats can be saved in four different bit depths:

a. 1-bit (monochrome)
b. 4-bit (16 colours),
c. 8-bit (256 colours)
d. 24-bit (16 million colours). Even in 24-bit colour, the colours are restricted to RGB combinations

The BMP file format stores colour information for each and every pixel in the image without any compression. For example, a 60x60 pixel BMP image will include color data for 3600 pixels. This way of storing image information allows crisp, high-quality graphics, but it generates files having large file sizes. The JPEG (Joint Photographic Expert Group) and GIF (Graphical Interchange formats) are also bitmaps, but they use image compression algorithms which can significantly decrease their file size. For this reason, JPEG and GIF images are used on the Web, while BMP images are often used for printable images [3, 4].

## 4. CONCEPTUAL APPROACH

In this paper, the linear recursive approach is used. The concept focuses on using a 24 bit BMP image format to hide information. Linear recursive approach generates different pixel locations in a 24 bit BMP file to hide secret information. A 24 bit BMP file format is best for hiding secret information because of its size. The concept considers a 24 Bit BMP image, Linear recursive approach and initial value which is used with a linear recursive approach.

The algorithm to generate pixel locations to hide secret information is as follows:

**4.1** The initial value finds the first-pixel position to hide information and then using this initial value in linear recursive approach, the next pixel position is generated and so on. For example, if 24 bit BMP is of 40 *50 size then the total number of pixels in the image is 2000. Each pixel is of 3 bytes so total image size is 6000 bytes. The total space for hiding secret information is 6000 bytes.

**4.2** This paper uses a modulo mathematical approach to generate new pixel locations if the generated pixel position is beyond the total size. If the recursive approach generates pixel position which is beyond 6000, for example, 7345 then the Modulo mathematical approach is used. Modulo mathematical approach takes out the mod of the number and generates the remainder. In the above example, 7345 mod 6000 will generate 1345th location. So the secret information will be stored at 1345th pixel location.

**4.3** Incase if the modulo approach generates the same pixel location then increment the pixel location by 1 and secret information will be stored at a new incremented location. For example, if the modulo approach generates 1345th location again then the new incremented pixel location is 1345+1=1346. So secret information will be hidden at this new location.

## 5. IMPLEMENTATION

Suppose the secret message to be hidden, is "Linear recursive equation". To hide the above sentence, we have to generate ASCII code of all alphabets and then their corresponding binary equivalents

| Alphabets | ASCII Code | Binary |
|---|---|---|

| | | Equivalent |
|---|---|---|
| l | 108 | 01101100 |
| i | 105 | 01101001 |
| n | 110 | 01101110 |
| e | 101 | 01100101 |
| a | 97 | 01100001 |
| r | 114 | 01110010 |

Total characters in secret message =23
Each character takes one byte in memory.
Total number of bits to be hidden= 23* 8= 184 bits
Consider a Linear Recursive Equation
tn = a tn-1 + b
The constant value of a=2
The constant value of b=5
Initial value (t0) = 1
Using this recursive Equation, we will find out the pixel locations where the first alphabet of secret information can be hidden.
Let us take a 24-bit BMP image of 300*300 size i.e 300 columns and 300 rows.
The total number of pixels=300*300 =90000 pixels.
Each pixel takes 3 bytes of storage, so the total number of bytes=90000*3=270000 bytes
Total space occupied by 24-bit BMP image in bytes = 270000.
First alphabet is l whose binary equivalent is 01101100. Starting from LSB, the first bit is 0.
This means the first bit i.e., 0 of the secret message will be stored at t0 location.
t0=1, so 0 will be hidden at pixel location 1
Now,
t1=a t0 + b
t1=2*1+5=7
This means the second bit i.e., 0 of the secret message will be stored at 7th pixel.
t2= a t1 + b
t2= 2* 7 + 5=19
This means the third bit i.e., 1 of the secret message will be stored at 19th pixel.
t3= a t2 + 5
t3 = 2* 19 + 5=43
This means the fourth bit i.e., 1 of the secret message will be stored at 43rd pixel.
t4 = 91, so fifth bit i.e., 0 will be hidden at 91th pixel
t5 = 187, so sixth bit i.e., 1 will be hidden at 187th pixel
t6 = 379, so seventh bit i.e., 1 will be hidden at 379th pixel
t7 =765, so eighth bit i.e., MSB 0 will be hidden at 765th pixel
We found the places to store the binary equivalent of l at different pixel locations. To hide next i, whose binary equivalent is 01101100, we find another pixel locations to store LSB i.e., 0 of the secret message, we again generate pixel locations as follows
t8 = at7+5
t8=2*765+5=1535
t9=2*1535+5=3075
t10=2*3075 + 5=6155
t11=2*6155+5=12315
t12= 2*12315+5=24635
t13=2*24635+5=49275
t14=2*49275+5=98555
t15=2*98555+5=197115
Second alphabet i, whose MSB, the last bit will be stored at 197115th pixel location.
Now to store the next character n, whose binary equivalent is 01101110, we will again find pixel locations. To store 0 (the LSB of the secret message), the calculated pixel location is
t16 = 394235 but we have total 27000 locations.
So to store t16 we use the modulo Mathematical Approach.
The modulo approach generates remainders. So using this approach, we can generate new locations to hide information.
As we have generated t16=394235,
We use the mod to take out new location
t16= t16 mod 270000=124235 so the LSB 0 of secret information 01101110 will be stored at a new calculated byte location is 124235.
t17=788475, t17=t17 mod 270000=248475, so the 18th bit of secret information '1' will be stored at 2,48,475th byte location.
t18=1576955, t18=t18 mod 270000=226955, so the 19th bit of secret information '1' will be stored at 226955th byte location.
t19=3153915, t18=t18 mod 270000=183915
When we take out mod operation, it is possible to get the same location again. This will create a problem as already some information is hidden at that location.
To resolve this issue, we increment the byte location by 1 and store the hidden information at a new location.
For example, t19= 183915 and t23 =183915.
20th bit of hidden information will be stored at 183915th location but t23 also generated the same location so 24th bit cannot be stored at that location.
The solution is
t23=183915+1=183916, so the 24th bit will be stored at a new location, which is generated by incrementing the byte location by 1.

## 6. STEGANALYSIS

Steganalysis refers to the detection of hidden content. It identifies the existence of hidden information. The objective of steganalysis is to determine accurately whether a secret message is hidden in the stego cover or not [5]. Steganography and steganalysis go side by side to ensure the strength of the embedded algorithm of a steganographic communication system. The embedded

algorithm is tested on three parameters: Capacity, Robustness and Perceptibility.

Capacity: This factor determines the amount of information that can be hidden in the given digital cover. The more information is hidden the more it is likely to be detected.

Robustness: It is the amount of modification the stego cover can withstand before an adversary can destroy hidden information [6].

Perceptibility: It is the factor that determines whether the stego is maintaining the similarity with the original digital cover. More similarity implies it is less likely to be perceived as stego.

These three factors are represented in Fig. 3 and used to determine the quality of the embedding algorithm in any steganographic system [15]. Since these factors are contradictory, a balanced approach needs to be adopted. We propose to carry out the planned work to overcome such limitations and achieve an optimized result.

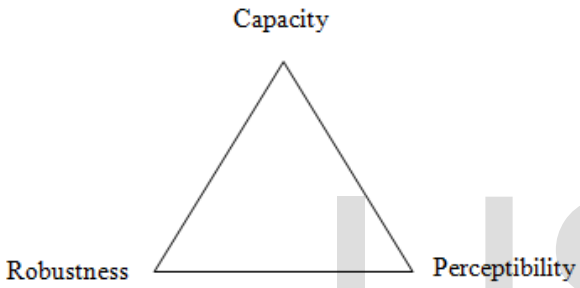


Fig.3 Representation of parameters of Steganalysis

## 7. CONCLUSION

We have introduced a new technique of hiding information in BMP image files using the Recursive Equation Approach. A recursive equation is used to find the location of the next pixel in the cover image while hiding the next bit of the secret message. The strength of the proposed technique is also analyzed using different parameters of steganalysis.

## ACKNOWLEDGEMENT

Encouragement, support and suggestions for content improvement from all colleagues, peers and seniors are gratefully acknowledged, which will remain essential for present and future scientific thought processes beyond scheduled professional endeavors.

## REFERENCES

*[1] Anderson, R. and Petitcolas, F., (1998). 'On the limits of steganography'. IEEE Journal on Selected Areas in Communications, Vol. 16, No. 4, pp.474–481.*

*[2] Cole, E., (2003). Hiding in Plain Sight: Steganography and the Art of Covert Communication. USA: Wiley Publishing.*

*[3] W. Brown and B.J. Shepherd, Graphics File Formats: Reference and Guide, Manning Publications, Greenwich, Conn, 1995*

*[4] Kirkby, D., (2001). BMP Format, Available at http://atlc.sourceforge.net/bmp.html*

*[5] Niels Provos and Peter Honeyman, Hide and seek: An introduction to steganography, IEEE Security and Privacy, vol. 1, no.3, pp. 32-44, 2003.*

*[6] Lin T. and Delp J., "A Review of Data Hiding in Digital Images," in Proceedings of the Image Processing, Image Quality, and Image Capture Conference, Georgia, pp. 274-278, 1999.*

*[7] Kumar, V., (2002). Discrete Mathematics. New Delhi, India: BPB Publication.*

*[8] Stallings, W., (1999). Cryptography & Network Security: Principles and Practice. NY, USA: Prentice Hall.*

*[9] Schneier, B., (1996). Applied Cryptography: Protocol, Algorithms, and Source Code in C, 2nd ed., John Wiley & Sons, New York*

*[10] Bender, W., Gruhl, D., Morimoto, N. and Lu, A. (1996). 'Techniques for data hiding'. IBM Systems Journal, Vol. 35, Nos. 3–4, pp.313–336.*

*[11] F.A.P Petitcolas, R.J. Anderson and M.G. Kuhn ; "Information Hiding a Survey", Proceedings of the IEEE, vol.-87, issue 7, pp. 1062-1078, 1999.*

*[12] K. Ahsan, & D. Kundur , "Practical data hiding in TCP/IP", Proceeding of the workshop on multimedia security at ACM multimedia,2002.*

*[13] K..M. Singh, L.S. Singh, A.B. Singh and K.S. Devi,"Hiding Secret Message in Edges of the Images", Information and Communication Technology, 2007. ICICT '07, pp. 238-241.*

*[14] Jagvinder Kaur and Sanjeev Kumar, " Study and Analysis of Various Image Steganography Techniques" IJCST Vol.2, Issue 3, September 2011*

*[15] Johnson, N.F. and Jajodia, S. (1998b), 'Steganalysis of images created using current steganography software', Proceedings of the 2nd Information Hiding Workshop, Portland, OR, April 14-17, pp. 273-289.*

*[16] A. Westfeld, "F5-A steganographic algorithm: High capacity Despite Better Steganalysis", Proc.4th Int'l Information Hiding. Workshop, Springer ,verlag vol . 2137 , New York,2001.*

*[17] I. Cox, J. Kilian, T. Leighton and T. Shamoon, "Secure spread spectrum watermarking for multimedia". IEEE Transaction On Image processing, Vol 6, issue 12 , pp1673-1687,1997.*

***Biographical Notes***

*Ruchika Sharma is an Assistant Professor in Jagannath Institute of Management Sciences, Delhi. She completed MBA(IT) from Symbiosis and M.Phill in Computer Science from Global Open University.*

*Dr. Vinay Kumar did his Ph. D. from the University of Delhi, MCA from Jawaharlal Nehru University, Delhi. He worked as a scientist in National Informatics Centre (NIC), Government of India for approximately 21 years. He also worked with Vivekananda Institute of Professional Studies (VIPS), affiliated to GGSIPU, as Professor and Dean of the IT Department and Research and Publication. He has extensively contributed to the knowledge arena through his books and research publications. He has published over 100 research papers in refereed journals and authored a book of Discrete Mathematics and a memoir Killed Instinct.*