Hiding Encrypted Message in the Features of Images

Kh. Manglem Singh, S. Birendra Singh and L. Shyam Sundar Singh
DOEACC Centre, Imphal – 795001, India

Summary
This paper proposes a novel least significant bit embedding algorithm for hiding encrypted messages in nonadjacent and random pixel locations in edges of images. It first encrypts the secret message, and detects edges in the cover-image. Message bits are then, embedded in the least significant bits and random locations of the edge pixels. It ensures that the eavedroppers will not have any suspicion that message bits are hidden in the image and standard steganography detection methods can not estimate the length of the secret message correctly.

1. Introduction
Steganography is the art and science of writing hidden messages in such a way that no one apart from the intended recipient knows of the existence of the message [1]. Unlike cryptography, where the existence of the message is clear, but the meaning is obscured, the steganographic technique strives to hide the very presence of the message itself from an observer. Steganography simply takes one piece of information and hides it within another. Computer files (images, sounds recordings, even disks) contain unused or insignificant areas of data. Steganography takes advantage of these areas, replacing them with information. One can replace the least significance bit of the original file (audio/image) with the secret bits and the resultant cover is not distorted. It is not to keep others from knowing the hidden information, but it is to keep others from thinking that the information even exists. If a steganography method causes someone to suspect that there is a secret information in the carrier medium, then this method fails [2]. The noise or any modulation induced by the message should not change the characteristics of the cover and should not produce any kind of distortion. Stego methods for digital media may be either in the spatial domain or in transform domain. Least significant bit (LSB) embedding is very frequently used in the spatial domain. The transform domain tools include those that involve manipulation of algorithms such as discrete cosine and wavelet transformations. These methods hide messages in more significant areas of the cover and may manipulate image properties such as luminance. There are many applications of steganography. Some of the applications for digital images include watermarking in copyright protection, feature tagging and secret communication. An author can embed a hidden message in a file so that he can later claim his ownership of intellectual property and copyright.
The paper is organized as follows. Section 2 gives a brief introduction on the historical development of steganography. Section 3 deals with the proposed technique. Section 4 mentions some of the detection techniques. Section 5 is on the experimental results, followed by conclusions at Section 6.

2. Historical Background
The earliest record of steganography is found from Histories of Herodotus (484 BC- 425 BC) [3,4]. Herodotus told how Demeratus, a Greek at Persian court, wanted to warn Sparta that Xerxes intended to invade Greek. To avoid capture by the enemy, he scrapped the wax of the tablets and wrote the message on the underlying wood. In one incident, Histiaeus was held by the Greek

Manuscript received April 5, 2007
Manuscript revised April 25, 2007

tyrant king Darius in Susa as a prisoner around 440 BC. He shaved the head of his most trusted slave and tattooed it with a message, which disappeared after the hair had grown. When the slave reached his destination, his hair was shaved and the message was recovered. In another incident, Herodotus described how a man named Harpagus killed a hare and hid a message inside its belly. Then, he sent the hare with a messenger disguised as a hunter.

Ancient Romans used invisible inks that were prepared from readily available substances like fruit juice, urine and milk to write between lines on innocent letters. The secret message reappeared on heating the letter. With the advancement of the science of chemistry, the history has shown the use other chemical as invisible inks. Both Axis and Allied spies used invisible inks during the World War II. Copper sulfate solution was used to write hidden message on handkerchief. The message would become visible when it was exposed to ammonia fume.

Microfilm was a popular medium during the Franco-Prussian War (1870-1871). With the discovery of photography, it allowed to reduce message greatly. During Russo-Japanese war of 1905, microscopic images were hidden in ears, nostrils and under fingernails. During the World War I, messages to and from spies were reduced to microdots by several stages of photographic reduction and then, stuck on top of printed periods or commas in innocuous cover material such as magazines. It was possible with the advancements in photography, lens making and film processing. This method was used to send information around with spies avoiding detection. Intensive steganographical experimentation was seen during the World War II. A Japanese spy, named Velvalee Dickinson, known as the "Doll Woman" used her dolls business as an analogical code [5]. Such a message was "Doll in a hula skirt is in the hospital and doctors are working around the clock" which translates as "Destroyer USN Honolulu is badly damaged and in Seattle undergoing around the clock repairs".

The null ciphers (unencrypted message), Cardin Grille and Semagrams were used to hide message inside an innocent looking container [6-9]. For example, a German spy at the German Embassy in Washington D.C. sent in the telegram the following messages during the World War II to their headquarters at Berlin: "Apparently neutral's protest is thoroughly discounted and ignored. Isman hard hit. Blockade issue affects pretext for embargo on by-products, ejecting suets and vegetable oils." Taking the second letter from each word the following message emerges: "Pershing sails from NY June 1". A Cardan grille is an important tool for reading of a message obfuscated through steganography. It was introduced by Renaissance mathematician Gerolamo Cardano in 1550. The grille is usually a card perforated with holes at selected places. To read a message, the card is laid over the page of texts that contains a hidden message. Only the letters that appear through the holes in the grille are read. Semagram does not use writing to hide message [8,9]. It is a picture or glyph associated with a concept. Jargon code is a secret language or phrases expressed in it, used to communicate secretly. It is often used by the military. "Tora! Tora! Tora!" , for example, is a famous jargon code used by Imperial Japanese Navy, denoting an order to "carry out the attack on Pearl Harbor".

By the end of twentieth century, the government began to use steganography for protecting their currency from being counterfeited. They have employed special inks, dyes, embedded threads and microstrips that denote the face value of the bill. Recent articles in national media sources such as the USA Today pointed that members of terrorist organizations use steganography as a tool to attack against the western interests [10,11].

3. Proposed Technique

The simplest way to hide data on an image is to replace the least significant bits (LSB) of each pixel sequentially in the scan lines across the image in raw image format with the binary data. The portion, where the secret message is hidden is degraded while the rest remain untouched. An attacker can easily recover the hidden message by repeating the process. To add better security, the message to be hidden is first encrypted using the simplified data encryption standard (S-DES) and then is distributed randomly by a pseudo random number generator (PRNG) across the image. The following is one form of the PRNG:

$$y(n+1) = (ay(n)+b) \bmod M \quad \text{for } n \geq 0 \tag{1}$$

where the parameters a, b, M and $y(0)$ are referred to as the multiplier, increment, modulus and seed respectively. These values must be chosen carefully in advance.

This approach may raise suspicion that the image contains the secret message, because the resulting stego-image appears as speckles at the point of message embedding. A better approach is to hide the message in the regions that are least like their neighboring pixels. Such regions contain edges, corners, thin lines, ends of lines, textures etc. with fast varying pixel values. Majority of images contain edges dominantly. An attacker has less suspicion the present of message bits in edges, because pixels in edges appear to be either much brighter or dimmer than their neighbors. Edges from the image can be detected easily by applying the appropriate edge detection filter and many such standard filters are available. For a 2×2 window shown in Figure 1, Roberts cross-gradient operator [12] has the following form:

$$D = |G_x| + |G_y| \tag{2}$$

where $G_x = x_3 - x_2$, $G_y = x_1 - x_4$ and x_1, x_2, x_3 and x_4 are the pixels in the 2×2 window, scanning from the top left to bottom right.

x_1	x_2
x_3	x_4

Figure 1 Pixels in a 2×2 window.

All edge detection filters are threshold- based technique. A pixel that is detected as an edge point before LSB embedding may not be detected as an edge point after embedding, because the value of D may change after embedding. Therefore, LSB embedding in edges may require the original cover-image for the extraction of the secret message. LSB embedding creates an imbalance between neighboring gray-scales, because gray-scale values cover byte of the pixel flips.

The proposed algorithm does not require the original cover-image for the extraction of the secret message. It ensures that the edge pixels are detected before embedding the secret message are not different from the points are detected after embedding it. It uses LSB embedding algorithm in the edges randomly distributed across the image, flipping the gray-scale values among $2i-1$, $2i$, $2i+1$ and $2i+2$, depending on the values of D, G_x, G_y, cover byte (CB) and secret message bit (*MB*). The new value of the *CB* after LSB embedding is given by:

$$x_1 = \begin{cases} x_1+1, & \text{if } D \geq \theta \,\&\, G_x \geq 0 \,\&\, x_1 \text{ is even} \,\&\, MB = 1 \\ x_1-1, & \text{if } D \geq \theta \,\&\, G_x < 0 \,\&\, x_1 \text{ is even} \,\&\, MB = 1 \\ x_1+1, & \text{if } D \geq \theta \,\&\, G_x \geq 0 \,\&\, x_1 \text{ is odd} \,\&\, MB = 0 \\ x_1-1, & \text{if } D \geq \theta \,\&\, G_x < 0 \,\&\, x_1 \text{ is odd} \,\&\, MB = 0 \\ x_1, & \text{otherwise.} \end{cases} \tag{3}$$

where x_1 is the cover byte and θ is a pre-defined threshold. The same equation is used to embed

the message bit in x_3, replacing x_1 by x_3 and G_x by G_y. The following equation is used to embed the message bits to the cover bytes x_2 and x_4:

$$x_2 = \begin{cases} x_2 - 1, & \text{if } D \geq \theta \,\&\, G_y \geq 0 \,\&\, x_2 \text{ is even} \,\&\, MB = 1 \\ x_2 + 1, & \text{if } D \geq \theta \,\&\, G_y < 0 \,\&\, x_2 \text{ is even} \,\&\, MB = 1 \\ x_2 - 1, & \text{if } D \geq \theta \,\&\, G_y \geq 0 \,\&\, x_2 \text{ is odd} \,\&\, MB = 0 \\ x_2 + 1, & \text{if } D \geq \theta \,\&\, G_y < 0 \,\&\, x_2 \text{ is odd} \,\&\, MB = 0 \\ x_2, & \text{otherwise.} \end{cases} \quad (4)$$

Truth tables of Equations 3 and 4 are shown in Tables 1 and 2. It is assumed that gray-scale values of neighboring pixels remain same after embedding the secret message bit in the edge point in the window. This condition enforces us to apply the edge detection filter in non-overlapping window only, such that the secret message bits are stored in nonadjacent edge pixels. The absolute value of *D* after embedding the message bit does not change for all blocks that are not detected as edge blocks. However, the absolute value of *D* increases for edge blocks after embedding the message bits.

4. Detection Techniques

Many algorithms were proposed for the estimating the length of the secret message in the cover image. Westfeld [14] proposed the blind steganalysis based on statistical analysis of PoVs (pairs of values). This method, so-called χ^2 - statistical test, gives a successful result to a sequential LSB steganography only. Fridrich *et al*. [13] proposed the RS steganalysis. This method makes small alternations to the least significance bit plane in an image. It uses these alternations and a discrimination function to classify three types of pixels groups: R, S and U. The counts of the groups reflect the embedding length accurately. This method works very well for the random LSB steganography. Dumitrescu *et al*. [15] proposed the sample pair analysis, which utilizes finite state machine to classify groups of pixels modified by a pattern. Zhang *et al.* [16] proposed difference image histogram that gives the correlation between the LSBs of pixels and remained bit planes in the image. Farid proposed [17] a detection algorithm based on higher-order statistics for separating original images from stego-images. Zhi *et al*. [18] proposed a blind detection (Gradient Energy) algorithm that estimates the accurate of embedded message through the analysis of the variation of the gradient energy resulted from the spatial LSB embedding.

5. Experimental Results

The message to be hidden in the image was first encrypted using the S-DES algorithm. Features (edges, corners, thin straight lines, end of lines etc.) were detected from the cover-images using Roberts' edge detection algorithm. Random pixel locations were found in the cover-image by the PRNG. Then, message bits were embedded at the random-edge pixel locations using LSB insertion algorithm. The proposed algorithm is named as "Random Edge LSB" (RELSB) technique. Mandrill, Woodland Hill and Miramar grayscale images of 512×512 size were used for comparison the performance of the proposed method with three different LSB embedding techniques written below in estimating the correct length of the message bits using the gradient energy detection technique:

- Sequential LSB (SLSB) embedding
- Random LSB (RLSB) embedding
- Edge LSB embedding (ELSB) and

The gradient energy detection technique gives -6180, -2120 and 52 as message bit lengths from Mandrill, Woodland Hill and Miramar respectively when no secret message was embedded to them.

Table 1: Truth table of *D*, *CB*, *MB* and output for x_1 and x_3

Rule	D	G_x	x_1	MB	Output x_1
1	$D \geq \theta$	$G_x \geq 0$	even	1	$x_1 = x_1 + 1$
2	$D \geq \theta$	$G_x < 0$	even	1	$x_1 = x_1 - 1$
3	$D \geq \theta$	$G_x \geq 0$	odd	0	$x_1 = x_1 + 1$
4	$D \geq \theta$	$G_x < 0$	odd	0	$x_1 = x_1 - 1$

Table 2: Truth table of *D*, *CB*, *MB* and output for x_2 and x_4

Rule	D	G_y	x_2	MB	Output x_2
1	$D \geq \theta$	$G_y \geq 0$	even	1	$x_2 = x_2 - 1$
2	$D \geq \theta$	$G_y < 0$	even	1	$x_2 = x_2 + 1$
3	$D \geq \theta$	$G_y \geq 0$	odd	0	$x_2 = x_2 - 1$
4	$D \geq \theta$	$G_y < 0$	odd	0	$x_2 = x_2 + 1$

Table 3 shows the result of estimating the length of the secret message bits by the gradient energy technique from three images embedded with 6%, 12% and 18% LSBs of the cover-image with the secret message bit.

From Table 3, it has been seen that the gradient energy technique could not estimate the length of the secret message bit accurately for RELSB embedding technique. However, the detection technique succeeded to estimate the length for other three LSB embedding methods.

6. Conclusions

The paper described a novel method for embedding secret message bit in least significant bit of nonadjacent and random pixel locations in edges of images. No original cover image is required for the extraction of the secret message. It has been shown experimentally that the blind LSB detection technique like the gradient energy method could not estimate the length of the secret message bits accurately for the proposed algorithm.

Table 3: Comparison of different LSB embedding techniques in term of secret message length

Sequential LSB			
LSB %	Mandrill	Woodland Hill	Miramar
6	11304	10288	20208
12	27820	23228	33052
18	38424	30336	49372
Random LSB			
6	4380	17848	15920
12	16344	29848	31448
18	29588	44620	44960
Edge LSB			
6	15248	1656	10120
12	11864	880	5104
18	27968	-6212	4940
Random Edge LSB			
6	2884	-8724	6060
12	-2824	-6860	9656
18	-2428	-5264	6260

References

[1] R. Chandramouli and N. Memon, "Analysis of LSB based Image Steganography", IEEE ICIP, pp. 1022-1022, Oct. 2001.

[2] D. Artz, "Digital Steganography: Hiding Data within Data", IEEE Internet Computing, pp. 75-80, May-June 2001.

[3] H. Hastur, Mandelsteg, ftp://idea. sec.dsi.unimi.it/pub/security/crypt/code/

[4] K. Rabah, "Steganography- the Art of Hiding Data", Information Technology of Journal, 3(3), pp. 245-269, 2004.

[5] http://fbi.edgesuite.net/libref/historic/famcases/dickinson/dickinson.htm

[6] N.F. Johnson and S. Jajodia, "Exploring Steganography: Seeing the Unseen", Computer 31, pp. 26-34, 1998.

[7] http://en.wikipedia.org/wiki/null_cipher

[8] F.A.P. Petitcolas, R.J. Anderson and M.G. Kuhn, "Information Hiding – A Survey", IEEE Proc., Special Issue on Protection of Multimedia Content, 87(7), pp.1062-1078, July 1999.

[9] http:/wetstonetech.com/f/stego/kessler.pdf

[10] D. Verton, "Expert Debate Biggest Network Security Threats", USA Today, 12 April, 2002.

[11] K. Maney, "Bin Laden's Messages could be Hiding in Plain Sight", USA Today 19 December, 2001.

[12] R. C. Gonzalez and R. E. Woods, Digital Image Processing, Prentice Hall of India, 2000.

[13] J. Fridrich, M. Goljan, and R.Du, " Reliable detection of LSB steganography in color and gray-scale images," Proc. ACM Workshop Multimedia Security, Oct.5, pp.27-30, 2001.

[14] A. Westfeld, "Detecting low embedding rates", Proc.Information Hiding Workshop. vol. 2578 of Springer LNCS, pp. 324–339, 2002.

[15] S. Dumitrescu, X. Wu, and Z. Wang, “Detection of LSB Steganography via Sample Pair Analysis”, IEEE Tran.s on Signal Proc., vol.51, no.7, Jul. 2003.
[16] T. Zhang and X. Ping, “Reliable detection of LSB steganography based on the difference histogram”, Proc. IEEE ISCASS, vol. 3, pp.545-548, 2003.
[17] H. Farid, “Detecting hidden messages using higher-order statistical models," Proc ICIP, Sept. 2002.
[18] L. Zhi, S. A. Fen and Y. Y. Xian, “ A LSB steganography detection algorithm”, Proc. IEEE ISPIMRC, pp. 2780-2783, 2003.

Kh. Manglem Singh received his BSc Engg. from DEI, Agra in the year 1986, his M.E. (Control & Instrumentation in the year 1992, M.S. (Software systems) from BITS, Pilani in the year 1994, and his Ph.D.(Digital Image Processing) from Indian Institute of Technology, Guwahati in the year 2006. He has published thirty papers in National and International journals and conferences including four IEEE conference papers. He is currently working as a principal design engineer at DOEACC Centre, Imphal. His research interests are in Digital Signal & Image Processing, Information Security and Fuzzy Logic & Applications.

S. Birendra Singh received B.E., M.Tech, Ph.D. from Delhi University, Indian Institute of Technology, Delhi & Novocherkassk Polytechnic Institute, Russia in the years 1976, 1978 and 1993 respectively. He has published ten papers in international journals and conferences. His research interests are in Operating Systems and Computer Networks etc. He is currently working as a director at DOECC Centre, Imphal.

L. Shyam Sundar Singh did B.E. and M.E. from Delhi University and Bangalore University in 1992 and 1999 respectively. He has published three papers in international journal and conferences. His research interests are in Operating Systems, Informational Security and Computer Networks etc. He is currently working as a design engineer at DOEACC Centre, Imphal.