Information Hiding using Improved LSB Steganography and Feature Detection Technique

Mamta Juneja, Parvinder Singh Sandhu

Abstract— This paper proposes *an improved* ***Least Significant bit****(LSB)* ***based Steganography technique for images imparting better information*** *hiding* ***.It presents an embedding algorithm for hiding encrypted messages in nonadjacent and random pixel locations in edges and smooth areas of images. It first encrypts the secret message, and detects edges*** *as well as smooth areas* ***in the cover-image using improved*** *feature* ***detection filter. Message bits are then, embedded in the least significant byte of randomly selected edge area pixels and 1-3-4 LSBs of red, green, blue components respectively across randomly selected pixels across smooth area of image. It ensures that the eavesdroppers 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. The Proposed approach is better in PSNR value and Capacity as shown experimentally than existing techniques.***

Index Terms—Canny filter, Feature detection, Hough transform, Image Steganography, Information hiding, LSB based Insertion.

I. INTRODUCTION

Steganography is the art of hiding information in ways that prevent the detection of hidden messages. Steganography is a process that involves hiding a message in an appropriate carrier e.g., an image, an audio or video file. The carrier can then be sent to a receiver without anyone else knowing that it contains a hidden message. Literally meaning "covered writing", it includes a wide range of secret communication methods like invisible inks, microdots, character arrangement, digital signatures, covert channels, spread spectrum etc. that conceal the very existence of message. Cryptography and steganography are related to each other. The main difference between cryptography and steganography is that cryptography scrambles the message so that it becomes difficult to understand whereas steganography hides the very existence of a message. Steganography plays the central role in secret message communication. Several message hiding techniques have been developed and implemented in the past using digital images, audio/video files and other media. These include least significant bit insertion, masking, filtering and algorithmic transformations to name a few.

In the following sections, first a brief description of concepts and available methods is presented followed by a detailed description of proposed techniques and their implementation results.

II. RELATED WORK

Neil F. Johnson and Sushil Jajodia in [1] discuss three popular methods for message concealment in digital images.

Manuscript received on April, 2013.

Mamta Juneja, Assistant Professor, University Institute of Engineering and Technology, Panjab University, Chandigarh, India.

Dr. Parvinder Singh Sandhu, Professor, Rayat and Bahara Institute of Engineering and Bio-Technology, Mohali, India.

These methods are LSB insertion, masking and filtering and algorithmic transformations.

LSB insertion is a simple approach for embedding information in a cover file. It is vulnerable to even a slight image manipulation. Image conversion from a format like GIF or BMP which reconstructs the original message exactly (i.e., lossless compression in which bits are compressed without losing any bits during compression and exactly recovered during decompression) to a JPEG which does not (i.e., lossy compression in which some bits are lost during compression resulting in some loss in fidelity) and then back could destroy the information hidden in the LSBs. LSB insertion can be performed in 24-bit, 8-bit or gray-scale images.

Masking and Filtering, usually restricted to 24-bit and gray-scale images, hide information by marking an image, in a manner similar to paper watermarks. Watermarking techniques may be applied without fear of image destruction due to lossy compression because they are more integrated into the image. Traditional steganography conceals information; watermarks extend information and become an attribute of the cover image. Digital watermarks may include such information as copyright, ownership, or license.

Algorithmic Transformation techniques like redundant pattern encoding, encrypt and scatter etc. exist which use different approaches for concealing messages. In redundant pattern encoding, a small message may be painted many times over an image so that if the stego image is cropped, there is a high probability that the watermark can still be read. In encrypt and scatter, the data are hidden throughout an image. Scattering the message makes it appear more like noise. Proponents of this approach assume that even if the message bits are extracted, they will be useless without the algorithm and stego-key to decode them.

In [2], Kevin Curran and Karen Bailey analyze seven different image steganography methods. These methods are Stego1bit, Stego2bits, Stego3bits, Stego4bits, Stego ColourCycle, StegoPRNG, and StegoFridrich.

Stego1Bit method involves utilizing a single least significant bit of one of the RGB bytes of a 24-bit image for message concealment. As the color value is not changed much, it will not considerably alter the visual appearance of color and image.

Stego2Bits method involves utilizing two least significant bits of one of the RGB bytes of a 24-bit image for message concealment. Although the capacity of data storage is 2 times improved than Stego1Bit, the resulting image is degraded than Stego1Bit.

Stego3Bits method involves utilizing three least significant bits of one of the RGB bytes of a 24-bit image for message concealment. Although the capacity of data storage is 3 times improved than Stego1Bit, the resulting image is much degraded than Stego1Bit.

Stego4Bits method involves utilizing four least significant bits of one of the RGB bytes of a 24-bit image for message concealment. Although the capacity of data storage is 4 times

improved, the resulting image is much degraded than Stego1Bit and color palette is restricted to only 16 variations.

StegoColourCycle method involves cycling through the color values in each of the pixels in which to store the data. This means that the same color is not constantly changed e.g., the first data bit could be stored in the LSB of the blue value of the pixel, the second data bit in the red value and the third data bit in the green value.

Stego1BitPRNG method involves using a pseudo random number generator to choose random pixels in which to embed the message. This will make the message bits more difficult to find and reduce the existence of patterns in the image.

StegoFridrich method involves searching for the closest color to the color of the pixel which has the correct parity for the bit to be hidden. The message is hidden in the parity bit of the RGB values of close colors. For the color of each pixel into which a message bit is to be embedded the closest colors in the palette are searched until a palette entry is found with the desired parity bit. This technique does not change the palette in any way either by ordering it or by increasing the colors in it.

Several digital data hiding techniques for images like substitution systems, hiding in two -color images, transform domain techniques, statistical steganography, distortion and cover generation are explored, analyzed, attacked and countered by Neil F. Johnson et al. in [3]. Chun-Shien Lu in [4] presents several techniques for steganography, watermarking, fingerprinting, signature based image authentication for digital image and audio files. Out of the several, only LSB insertion method is used in the implementation of the techniques proposed in this paper.

Alwan et al in [5] introduced a novel approach of image embedding on 8-bit images. The proposed method embeds three images and text in one image using edge-pixels. First, the edges of the image were obtained using Sobel mask filter. Second, the least significant bit (LSB) of each pixel is used. Finally, a gray level connectivity is applied using a fuzzy approach and the ASCII code is used for information hiding. The prior bit of the LSB represents the edged image after gray scale level connectivity, and the remaining six bits represent the original image and with very little difference in contrast. In 2003, Wu and Tsai presented an adaptive steganographic scheme based on pixel-value differencing in [6]. With their method, the hiding capacity of each pixel can be different. Edge areas or pixels hold more secret data than smooth areas because the degree of distortion tolerance of an edge areais naturally higher than that of smooth area. In addition, the features of the image blocks stay unchanged after Wu and Tsai's scheme is applied, meaning the embedding of the secret data does not change any smooth area into an edge area or any edge area into a smooth area.

Marvel and Retter in [7] introduced a new method of imagesteganography. The method embeds the hidden information within white Gaussian noise (WGN) which is subsequently added to the digital image to form the stego-image. The hidden information is encoded by an error-control code before it is embedded into the WGN signal. The WGN signalwith the embedded data, the stego-signal, is then added to the image. At the receiver, the embedded stego-signal is estimated as the difference between the stego-image and the denoised version of the stego-image. The embedded information is extracted from the estimated stego-signal and any remaining errors are corrected by the error-control decoder. Unfortunately, the estimate of the stego-signal is typically poor because the power of the signal is low compared to the image power, and thus denoising process is not optimal. Consequently, decoding errors are made. In an effort to address this shortcoming without increasing the stego-signal power (and visual detect ability), they found out that the locations of poor signal estimation, and thus decoding errors correlate to the edges within the image.Luo in [8] presented a method on embedding watermarks on cartoon images. The method converts the RGB color image to a gray-scale image. The next step is to apply edge detection on the previously converted gray-scale image using the Laplacian edge detection. After the edges are brought out, the next step is to differentiate between the background and the object which is to be segmented by applying morphological operation-dilation followed by flood-filling to fill the background with a explicit color. To embed the watermark, LSB insertion is used by ignoring the least significant bit, and let the first 7 bits do boolean plus in turn with one another, and get the result (either0 or 1). Then let the result do again boolean plus with one bit of the watermark's data which is to be encoded, and get the last result (either 0 or 1). The last result is the bit value that is to replace the least significant bit. Just as clever techniques have been devised for hiding information, an equal number of clever techniques have been designed to detect the hidden information. These techniques are collectively known as 'steganalysis'.RS Analysis in [9] makes small modifications to the least significant bit plane in an image then uses these modifications and a discrimination function to classify groups of pixels. The counts of the groups based on the modifications allow the calculation of an estimated embedding rate. Images that do not contain steganography often have a natural embedding rate of up to 3%, whereas images containing hidden information usually have estimated embedding rates which accurately reflects the amount of hidden information. RS Analysis is a special case of Sample Pairs Analysis, which also uses least significant bit modifications to help calculate an estimated embedding rate. Sample Pairs Analysis utilizes finite state machines to classify groups of pixels modified by a given pattern. Both steganalysis techniques are very accurate at predicting the embedding rate on stego-images using least significant bit embedding.

III. TECHNIQUES USED

A. Use *Feature detection*

The most important features of objects in images are edges. There are several edge detection algorithms like Laplace filter, Sobel filter, Prewitt filter, Canny filter, etc. In this paper, a new feature detection filter using Canny Filter along with Hough Transform is used which provides better results in detecting edges as cited in [10]. Canny filter is used as it provides better demarcation in edge areas and smooth areas which is need of this proposed steganography technique. The Canny method finds edges by looking for local maxima of the gradient of the image. The gradient is calculated using the derivative of the Gaussian filter. The method uses two thresholds to detect strong and weak edges, and includes the weak edges in the output only if they are connected to strong edges. This method is therefore less likely than the others to be "fooled" by noise, and more likely to detect true weak edges. It shows better results even in noisy conditions, providing actual edges by using non-maximal suppression, zero-crossing property to find the location of edges and hysteresis with thresholding.

In figure 1, it is quite clear that the gradient has a large peak centered on the edge. By comparing the gradient to a threshold value which in this case is 10 percent of the peak value, an edge can be detected if the threshold exceeds. In this case the edge has been found, but it becomes "broad" due to the threshold. However, since we know the edge occurs at the peak, we can localize it by computing the Laplacian in one dimension and the second derivative with respect to t. Hough transform as cited in [11]-[12] is used to link the edges given by canny as these are disjoint due to surrounding noise effects or other disturbances etc.

So, finding the zero crossings (shown in figure 1(c)), Figure 2 represents original image and the outcome after applying Canny and Hough edge detection operation on it and being set to bi- level according to a threshold.

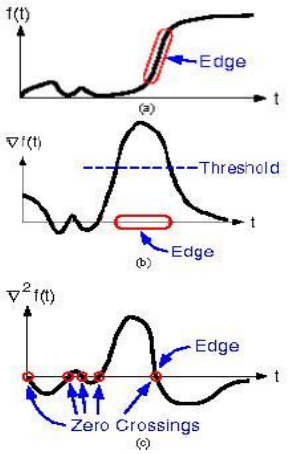

Figure1.(a)Originalimagesignal(one-dimension); (b)Gradientoftheoriginal imagesignal(firstderivativewithrespecttot); (c)Laplacianoperationonthe originalsignal (second derivative with respect to t)

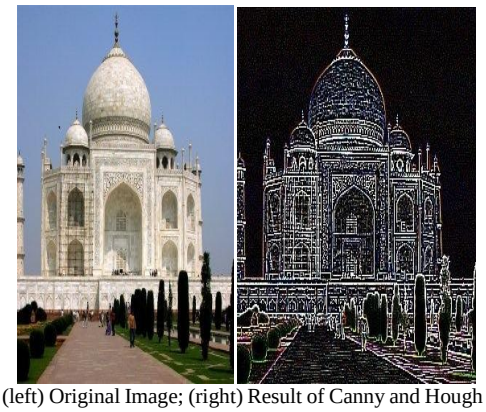

Figure 2. (left) Original Image; (right) Result of Canny and Hough transform

B. Image Based Steganography

Embedding a message into an image requires two files. The first is the innocent-looking image that will hold the hidden information, called the cover image. The second file is the message the information to be hidden. A message may be plain-text, cipher-text, other images, or anything that can be embedded in a bit stream. When combined, the cover image and the embedded message make a stego-image. A stego-key (a type of password) may also be used to hide then later decode the message. Most steganography software recommends the use of lossless 24-bit images such as BMP. The next- best alternative to 24-bit images is 256-color gray-scale images. The most common of these are BMP files

B1. Embedding data using Modified LSB (Least Significant bit) Insertion:

B1. a Embed data in Least significant byte of each pixel across Edge areas:

To embed the data, the LSB insertion as cited in [13]-[14] is used. LSB insertion is a common, simple approach in embedding information in a cover file. But in this improved LSB technique we will insert the data only in last significant byte i.e. blue component of a pixel as that having lowest contribution to the color image according to Human Visual System analysis. To hide a message in a 24-bit image, the B component of each pixel of RGB color image is modified. For example, the letter A can be hidden in a pixel with original data as:

(00100111 11101001 11001000)

The binary value for A is 01000001. Inserting the binary value for A in the given pixel would result in

(00100110 11101001 <u>01000001</u>)

The underlined bits are the only actually changed in the bytes used. On average, LSB requires that only half the bits in an image be changed. To hide more data, the cover image should have enough edge pixels to hide the data.

B1.b Embedding data using 1-3-4 LSB Insertion across Smooth areas

To embed the data in smooth areas 1-3-4 LSBs Insertion technique has been utilized which hides data in 1-bit in 1 least significant bit of Red component (Most significant byte),3-bits in 3 least significant bits of Green component and 4 bits in 4 least significant bits of Blue component(Least significant byte) of each selected pixel .This ratio 1:3:4 has been taken depending on their respective contribution of each red, green and blue component to the colors of RGB image.

C. Selecting the edge pixels randomly

To select the edge pixel randomly, a pseudorandom number generator (PRNG) will be used. Pseudorandom number generator is an algorithm that generates a sequence of numbers, the elements of which are approximately independent of each other. The outputs of pseudorandom number generators are not truly random - they only approximate some of the properties of random numbers. To use a PRNG, it first requires a seed. Seeding is the technical term for giving it an initial value, from which it can shoot out a sequence. If a PRNG is given the same seed, then it will give the same set of numbers every time and the elements of which are approximately independent of each other. The outputs of pseudorandom number generators are not truly random - they only approximate some of the properties of random numbers. To use a PRNG, it first requires a seed. Seeding is the technical term for giving it an initial value, from which it can shoot out a sequence. If a PRNG is given the same seed, then it will give the same set of numbers every time.

The linear congruential (also known as linear congruent) PRNG is probably one of the most commonly used in programs, and one of a family of pseudorandom techniques. It isn't used in any cryptographic software because it isn't cryptographically secure. The reason it is used is because it's easy to set up, and can look random. It generates numbers in the range (0...Z-1).

The reason it is known as the "linear congruential" method is because at the heart of it lies this formula (part of which is of the form ax + c):

$$X_{n+1} = (A * X_n + C) \bmod Z$$

To make random numbers, first, you need to pick a seed (X0), a multiplier (A), a modulus (Z) and a constant (C) to increment A*Xn. The modulus basically says what the range of the outputted list of numbers is; the output will be a list of numbers from 0 to Z-1. We can pick any value we like for X0, A, Z and C, as long as four conditions are satisfied:

1. The increment C must be relatively prime to our modulus, Z.
2. A-1 must be a multiple of every prime p that divides Z.
3. A-1 must be a multiple of (a number) if Z is a multiple of a number.
4. X0, A, C and Z must all be greater than 0.

Then, keeping C, A and Z the same, we iterate the formula, to get values for X1 to XZ.

IV. PROPOSED APPROACH

In this paper, we have proposed a new technique for hiding data in images with high capacity and imperceptibility. This new modified approach works in following steps:

1. Divide the image into smooth and edge areas using Canny Filter with Hough transform as mentioned above in Section III.A.
2. Apply Encryption on input using S-DES algorithm cited in [15].
3. Embed data in least significant byte of all pixels selected in random manner using PRNG across edge areas as mentioned above in Section III.B1.a and Section III.C respectively.
4. Embed data using 1-3-4 LSB technique as mentioned above across smooth areas at random locations as mentioned above in Section III.B1.b.

V. RESULTS AND DISCUSSIONS

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 Canny filter and Hough Transform. Random pixel locations were found in the cover-image by the PRNG. Then, message bits were embedded at the random-edge area pixel locations and smooth area pixel locations using modified LSB insertion algorithm. Figure 3 presents the results of applying this technique to standard image Lena and cumulative results analysis is presented in table 1.

Figure 3. A. Original image B. Stego Image

TABLE 1: Capacity And Psnr Comparison With Simple Lsb Technique Comparison On Capacity

Host Images	*LSB Technique*	*Proposed Technique*
	Capacity	*Capacity*
Lena	467004	561345
Baboon	720785	830546
Pepper	482599	588459
Jet	463758	505658
Bridge	718743	760779
Scene	593801	669880

COMPARISON ON PSNR

Host Images	*LSB Technique*	*Proposed Technique*
	PSNR	*PSNR*
Lena	41.0053	47.5897
Baboon	33.9879	36.3637
Pepper	42.3743	45.9238
Jet	39.3827	42.5050
Bridge	36.2928	40.3273
Scene	37.3948	41.2382

VI. CONCLUSION

The paper proposed a new technique for information hiding. It presents an improved steganography method for embedding secret message bit in least significant byte of nonadjacent and random pixel locations in edges of images and 1-3-4 LSBs of red, green and blue components of randomly selected pixels across smooth areas. No original cover image is required for the extraction of the secret message. The research was aimed towards the evaluation and development of a new and enhanced information hiding technique based on LSB. The primary objective of this paper is to propose a solution that is robust, effective and to make it very hard for human eye to predict and detect the existence of any secret data inside the host image. This has been achieved by using those bits for data storage that are on edges and using blue component of color image to which human eye is least perceptive. The proposed solution has not only achieved what was required but has also increased the data hiding capacity of the host image by utilizing all the pixels.

REFERENCES

[1] Neil F. Johnson, Sushil Jajodia, "Exploring Steganography, Seeing the Unseen," *IEEEComputer Magazine*, pp. 26-34, February1998.

[2] Kevin Curran, Karen Bailey, "An Evaluation of Image Based Steganography Methods," *International Journal of Digital Evidence*, Vol.2 no. 2, pp. 1-40,Fall 2003.

[3] Niel F. Johnson, Zoran Duric, Sushil Jajodia, "Information Hiding, and Watermarking - Attacks & Countermeasures," Kluwer Academic Publisher, 2000.

[4] Chun Shien Lu, "Steganography and Digital Watermarking Techniques for Protection of Intellectual Property" in Multimedia Security, Idea Group Publishing, Singapore, 2005, pp. 75-157.

[5] R.H.Alwan, F.J.Kadhim, A.T.Al-Taani. "Data Embedding Based on Better Use of Bits in Image Pixels". International Journal of Signal Processing. Vol. 2, no.1,pp:104-107,2005.

[6] N. Wu, M. Hwang,"Data hiding: Current status and key issues", International Journal of Network Security,vol.4,no.1,pp.1-9, 2007.

[7] Lisa M. Marvel, Charles T. Retter, "The Use of Side Information in Image Steganography", Proceedings of IEEE International Symposium on Information Theory and Its Applications, Honolulu, Hawaii, USA, November 5-8, 2000.

[8] W. Luo, "Object-Related Illustration Watermarks in Cartoon Images",Masters Thesis, Department of Simulation and Graphics,

Otto-vonGuericke University Magdeburg, Germany, February 2004.
[9] Jessica Fridrich , Miroslav Goljan , Rui Du, "Detecting LSB Steganography in Color and Gray-Scale Images", IEEE MultiMedia, vol.8 no.4, pp.22-28, October 2001.
[10] Mamta Juneja, Parvinder Singh Sandhu, "Performance Evaluation of Edge Detection Techniques for Images in Spatial Domain", International Journal of Computer Theory and Engineering, Vol. 1, No. 5,pp 614-621,December,2009.
[11] V.F. F. Leavers, Shape Detection in Computer Vision Using the Hough Transform, Springer-Verlag New York, Inc., Secaucus, NJ, 1992.
[12] D.Ioannou,W.Huda,F.Laine,"Circle recognition though a 2-D Hough transform and radius histogramming", Image Vision Computing ,Vol 17,no. 1,pp. 15-26,1999.
[13] Chi-Kwong Chan, l. M. Cheng," Hiding data in images by simple LSB substitution",*Pattern Recognition*,vol. 37, pp.469-474, 2004.
[14] Mamta Juneja , Parvinder Singh Sandhu, "Designing of Robust Image Steganography Technique Based on LSB Insertion and Encryption", Proceedings of the International Conference on Advances in Recent Technologies in Communication and Computing(ARTCOM-2009), Kerala,India,October 27-28, 2009.
[15] Data Encryption Standard (DES), National Bureau of Standards (US).Federal Information Processing Standards Publication National Technical information Service. Springfield VA. April 1997.

Ms. Mamta Juneja did masters in Computer science from Punjab Technical University, India and currently pursuing Doctorate in the same. She is working as Assistant Professor in University Institute of Engineering and Technology, Panjab University, Chandigarh, India. Her interest areas are Image Processing, Steganography, information hiding and Information Security.

Prof. Dr. Parvinder S. Sandhu is Doctorate in Computer Science and Engineering and working as Professor in Computer Science & Engineering department at Rayat & Bahra Institute of Engineering and Bio-Technology, Mohali, Punjab, INDIA. He is editorial committee member of various International Journals and conferences. He has published more than 150 research papers in various referred International journals and conferences. He chaired more than 100 renowned International Conferences and also acted as keynote speaker in different countries. His current research interests are Software Reusability, Software Maintenance, Machine Learning and Image Processing.