International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:1, No:7, 2007

# A Robust Method for Encrypted Data Hiding Technique Based on Neighborhood Pixels Information

Ali Shariq Imran, M. Younus Javed, and Naveed Sarfraz Khattak

***Abstract*—**This paper presents a novel method for data hiding based on neighborhood pixels information to calculate the number of bits that can be used for substitution and modified Least Significant Bits technique for data embedding. The modified solution is independent of the nature of the data to be hidden and gives correct results along with un-noticeable image degradation. The technique, to find the number of bits that can be used for data hiding, uses the green component of the image as it is less sensitive to human eye and thus it is totally impossible for human eye to predict whether the image is encrypted or not. The application further encrypts the data using a custom designed algorithm before embedding bits into image for further security. The overall process consists of three main modules namely embedding, encryption and extraction.cm.

***Keywords*—**Data Hiding, Image Processing, Information Security, Stagonography.

## I. Introduction

THOUGH the concept and idea of hiding critical information in images is quite an old one but with the advent of recent events of Iraq war and the ongoing cold war amongst the super powers, the importance and the need for information security has become a major issue. In this regard Stagonography can play an important role in hiding critical information without being noticed. Recent research in this field involves robust data hiding schemes without any image degradation.

Almost all kind of data such as plain texts, graphical images, audio and videos are represented in digital form, resulting in different types of digital media. The secret information can be embedded into the digital media by making tiny changes which cause very little notice or none at all to human perception. Such kind of secret data hiding techniques have many applications [1], [2]. Some of the most common used today's digital world are tamper proofing, copyright protection, hidden annotations, authentication, watermarking, secure and invisible communication. Thus the stagonography is the art and science of hiding messages in digital images in such a way that no one apart from the intended recipient knows of the existence of the message within a host image. This is in contrast to cryptography where the existence of the message itself is not disguised; rather the secret data content is obscured [3]-[5].

Most of the digital images contain some noise and redundant information which can be used to store critical information without being noticed [6], [7]. For all types of digital image, pre-processing is often a required step for many sophisticated exploring methods and to enhance the quality of the image both visually and theoretically. Such noisy area, channel planes and any other redundant bits can be utilized for storing and hiding important and critical information. Normally the representation of digital image is in a form of matrix of pixels where each pixel measures the brightness of the object [8]. By using these different properties of digital image pixels, secret data containing information can be hidden into images either in form of bit stream or any other suitable form. The data hiding usually consists of three stages namely: Pre-processing, data embedding and data extraction. Pre-processing is to find the suitable number of bits to be used for data representation. Data embedding is to manipulate the bits in the new space and data extraction transforms the manipulated coefficients back to the original space.

## II. Previous Work

Most of the data hiding schemes which are currently available focuses on the compressed images over the lossy channel which degrades the quality of the image [5]-[8]. On the other hand lossless data hiding schemes over un-compressed color images is the best solution for achieving robustness, accuracy, efficiency and high data capacity.

Chang and Tseng recently proposed a method for hiding secret information which is based on the side by side pixels information. The bits were estimated which were then used to store the secret data [3]. However the stored bits were not retrieved correctly. The experimental results on various images shows that 35 – 40% of the bits retrieved were incorrect. The percentage was even higher for larger secret

Manuscript received October 31st, 2007. This work was supported in part by the U.S. Department of Commerce under Grant BS123456.

Ali Shariq Imran did MS in Software Engineering from College of Signals, National University of Science and Technology, NUST, Rawalpindi, Pakistan, phone: +92-51-345 5365848; e-mail: gallian99@gmail.com.

M. Younus Javed, did his PhD. from University of Dundee UK. Currently he is the head of department (H.O.D) of Computer Engineering Dept. of Electrical and Mechanical Engineering (E&ME), National University of Science and Technology; e-mail: myjaved@ceme.edu.pk.

Naveed Sarfraz Khattak did his MS from Michigan State University (U.S.A). Currently he is associated with Computer Science Department of College of Signals as head of department (H.O.D), National University of Science and Technology, Rawalpindi, Pakistan; e-mail: naveed-khattak@mcs.edu.pk.

data to be embedded. Suk-Ling Li and Kai-Chi Leung then suggested a method to improve Chang-Tseng scheme so that the embedded secret data can be recovered properly from the stego-image [1]. Moreover they also proposed a novel method for data hiding by adaptive LSB (Least Significant Bits) substitution based on grey pixel value differencing, which was further supported by OPAP (optical pixel adjustment process) to reduce any chances of embedding error. The OPAP process was originally proposed by Chan and Cheng in 2004. Unfortunately, even after applying OPAP on LSB substituted bits the data retrieved was not always correct and thus the secret data could not be recovered properly. The quantitative analytical results can be found in [1].

This paper goes one step further and proposes a robust data hiding technique using NPI (Neighbourhood Pixels Information) along with modified LSB substitution method to achieve best performance, higher data capacity, un-noticeable image degradation and secret data embedding accuracy. The results are summarised in Section VII.

This paper is organized as follows. In the third section the System model is presented. Pre-processing module is discussed in Section IV. Data Embedding Module and Data Extraction modules are covered in Section V and VI respectively. Finally results are presented in the VII Section followed by Conclusion, Acknowledgement and References.

## III. System Model

In order to approach the solution in a modular manner, our proposed system is divided into several modules. These modules are formed in a way so that the output of every module becomes the input for the next module. However, the primary input of the system is either image or a text file to be embedded. There are three main modules which are further divided into sub-modules as:

1. Pre-processing Module
2. Data Embedding Module
3. Data Extraction Module

The modules forming the entire system include data embedding, candidate bits estimation, data conversion, bit stream encoding and hidden data retrieval. The following steps constitute the overall working of the application.

1. Pick green component of the host image
2. Calculate average value difference of 4-connected neighbourhood pixels.
3. Estimate candidate bits for each pixel
4. Convert the secret data into suitable bit stream
5. Embed the data using modified LSB mechanism.

The basic proposed system model consisting of three main modules is being shown in Fig.1.

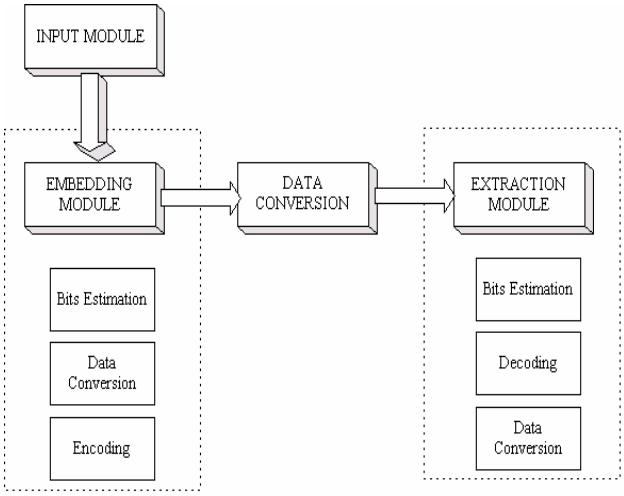


Fig. 1 Proposed System

## IV. Preprocessing Module

Pre-processing or the input module consists of acquiring image and text file and then performs the necessary calculations to estimate the number of candidate bits that will be used for substitution by the secret data. Details about sub-module are specified as follows.

### *A. Image and Text Acquisition*

Secret data to be hidden can be in any digital form. The host image can be acquired either by using a conventional camera and a scanner or using a digital one. The text file containing the data to be hidden is browsed from the directory. There are two types of image that are used. One is the host image i.e. in which the secret data is to be embedded and the other one is the logo or watermarked image that will be hidden inside the host image. Apart from this, a text file containing the secret message can also be embedded into the host image using the same embedding technique described in section V. The three types of input parameters are:

1. A host image
2. A text file containing secret data
3. A secret image or logo

Candidate bits are calculated from the host image and the secret data bits after converting into binary stream is replaced with the candidate bits.

### *B. Estimating Candidate Bits*

The bits are calculated for each pixel on the basis of color intensity of neighborhood pixels. This ensures that the image quality is also maintained and at the same time it becomes totally impossible for human eye to visually intercept existence of any secret data in image. The overall process flow diagram of the proposed system is being shown in Fig. 2.

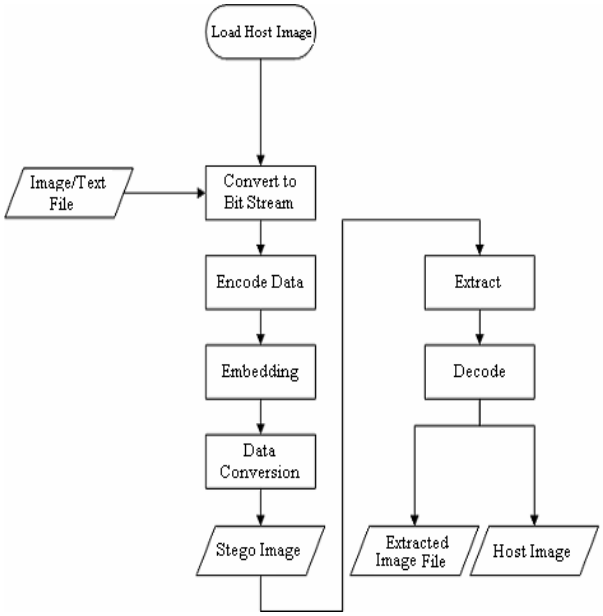


Fig. 2 Process Flow Diagram of Proposed System

First off the host image is converted into green components as it is less sensitive to human eye. Column wise raster scanning is performed on the host-image. Let $P_X$ be the input pixel and $P_L$, $P_T$, $P_R$ and $P_B$ be the left, top, right and bottom pixel of the input pixel respectively. The green component image is represented by $G_L$, $G_T$, $G_R$ and $G_B$ of the input pixel $G_X$ respectively.

The average green component value difference is calculated as

$$Gd = [(Gt - Gl) + (Gr - Gb)] /2 \quad (1)$$

By using the 4-connected neighborhood pixels, the first row and the first column are not abandoned [1]. Thus it increases the data embedding capacity. The data embedding capacity of the host image pixels depends on the green component value 'Gd'. Let Cb be the number of candidate bits that can be embedded into the input pixel of the host image $P_X$. The value of Cb is calculated as

$$Cb = \begin{cases} 1, & if\ -1 <= Gd <= 1 \\ \log 2 \mid Gd \mid & if\ .Gd > 1 or Gd < -1 \end{cases} \quad (2)$$

The maximum number of bits that can be used for substitution should not constitute more then 50% of the total bits, as the image quality gradually decrease once the number of candidate bits is greater then 4.

$$Cb = 4, \text{ if } Cb >= 5 \quad (3)$$

The results are obtained by modifying the bits of about 250 images of various sizes and then taking their average, which is shown in Table I.

TABLE I
BITS EMBEDDING CAPACITY IN HOST IMAGE

| *Image Size* | *Avg. Total # of Bits* | *Bits Capacity* |
|---|---|---|
| | ***Per 250 image*** | ***Per 250 image*** |
| 64 * 64 | 4,096 | 2,050 |
| 128 * 128 | 16,384 | 8,253 |
| 256 * 256 | 65,536 | 31,908 |
| 512 * 512 | 262,144 | 131,072 |

The results clearly indicate that we can replace half of the bits per pixel without degrading the image quality.

## V. PROPOSED EMBEDDING MODULE

The proposed embedding process makes use of the modified LSB substitution method. Rather then substituting same number of bits for each pixel, the candidate bits that were calculated in section IV are use for substituted by the secret data bit stream. This way it becomes impossible for interceptor to easily identify the bits and to extract the secret message. L.M Cheng proposed the following equation for data embedding which is also based on least significant bits substitution method but it fails to successfully embed the bits into the color host image [1]. The results are displayed in Fig. 7.

$$g'x = gx - gx \bmod 2^n + b \quad (4)$$

Instead of using the equation (4), the proposed system implements a modified least significant bits scheme by passing the number of candidate bits that can be substituted with the secret data in host image. The system caters two types of secret data.

### C. Text File Containing Secret Data

A text file containing the secret data to be embedded is first converted into ASCII pattern code. The generated code is then given a shape of bits which constitute the bit stream for the input text. Before embedding, based on the number of candidate bits different chunks from the bit stream is substituted by binary addition with the host input pixel. The stego-image after embedding secret data bits looks exactly like the original image.

### D. Image to be Embedded

If the data to be embedded is in the form of image, then the image is first reshaped from 2D (two dimensional) array to a single vector array. After each pixel element in an image is converted to a suitable character form, the same procedure is applied to generate the bit stream as for the text. Based on the number of candidate bits, different bit chunks of size specified by Nb are substituted with the host-image bits. The resultant image contains the secret image in the form of bits.

The original host-image and the stego-image after embedding random generated bit stream are shown in Fig. 3. The results are verified both theoretically and visually and

there is hardly any image degradation as far as quality of the stego-image is concerned.

(a) (b)

Fig. 3 Lena (a) Host-Image (b) Stego-Image

## VI. Extraction Module

The data extraction process is the reverse of the embedding process with the inclusion of finding embedded bits that were substituted per pixel during the data embedding process. Fig. 4 shows the flow chart for data extraction module.

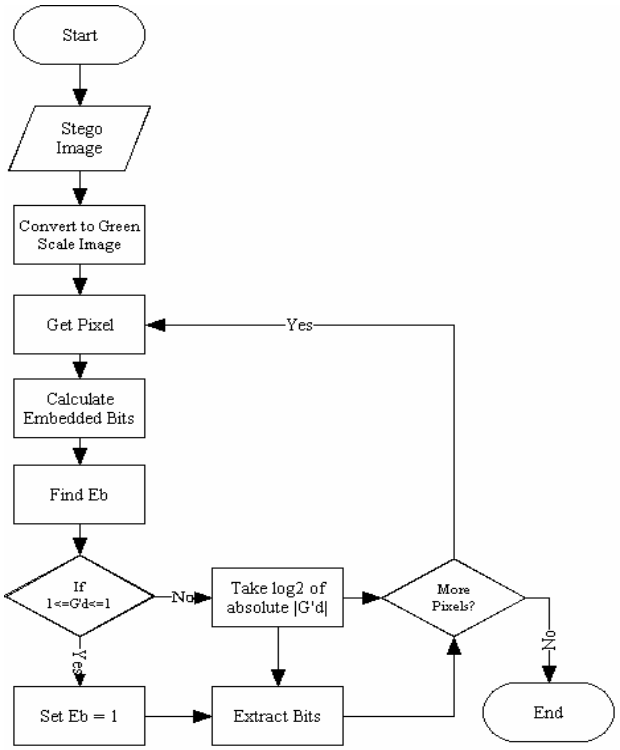


Fig. 4 Proposed Flow Chart for Extraction Process

The input of the extraction module is the stego-image. Column wise raster scanning is performed on the stego-image to find the number of bits per pixel. Let $P'_X$ be the input pixel and $P'_L$, $P'_T$, $P'_R$ and $P'_B$ be the left, top, right and bottom pixel of the input pixel respectively of the stego-image. The new green component of the modified image is represented by $G'_L$, $G'_T$, $G'_R$ and $G'_B$ of the input pixel $G'_X$ respectively.

The average green component value difference is calculated as in (2)

$$G'd = [(G't - G'l) + (G'r - G'b)]/2 \quad (5)$$

Let Eb be the embedded bits that were used during the substitution process. The Eb can be calculated as

$$Eb = \begin{cases} 1, & if\ -1 <= G'd <= 1 \\ \log 2 \mid G'd \mid, & if G'd > 1 or G'd < -1 \end{cases} \quad (6)$$

We know that the maximum bits that could be substituted per pixel can not be greater then 4. So we set the value of Eb to 4 such as

$$Eb = 4,\ if\ Eb >= 5 \quad (7)$$

This ensures to maintain the quality of the image.

## VII. Experimental Results

Experiments were initially conducted on Maple Leaves and after successful execution of the proposed method the time and performance comparisons between the LSB technique, Tseng Cheng scheme and proposed NPI technique was made. Images and text generated bit stream was embedded into different sizes of image for the performance analysis. The Time Comparison analysis between the three techniques is shown in Fig. 5.

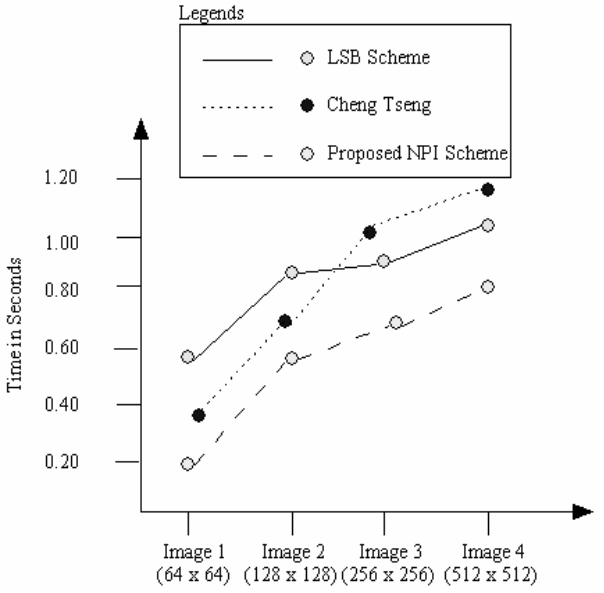


Fig. 5 Time Comparison Graph

Fig. 5 shows the comparisons of three tested schemes on various images. The results obtained are the average of the 16 images grouped by 4 different image sizes.

Moreover results from the six standard host-images namely Lena, Jet, Baboon, Bridge, Pepper and Scene using LSB

technique, Tseng Cheng scheme and proposed NPI based technique were obtained. The results clearly indicate that the proposed NPI based solution not only increases the data hiding capacity of the host-image by utilizing all the bits in the image but also reduces the overall processing time. Apart from this the stego-image obtained is less conspicuous to human eye which ensures that the confidential and secret data can be carried by the host-image without attracting the attention of the interceptors. The data capacity graph is shown in Fig. 6 which clearly indicates that the proposed NPI technique has the maximum data capacity as compare to other two.

Similarly the results for data embedding capacity were obtained by applying the three mentioned techniques on six classical host-images. The host images used were all of different sizes and thus the data capacity varies per image as well.

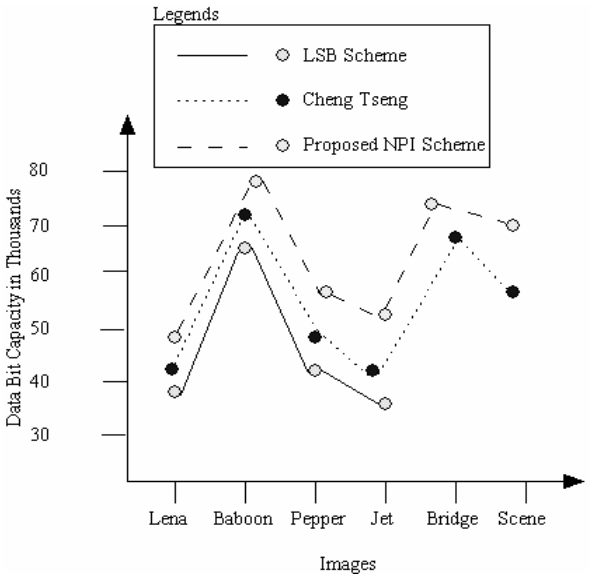


Fig. 6 Data Hiding Capacity per Image

The results are also summarised in Table II.

TABLE II
COMPARISON OF DATA EMBEDDED BITS PER IMAGE

| *Host Images* | *LSB Technique* | *Tseng Cheng Technique* | *Proposed NPI technique* |
|---|---|---|---|
| | *Capacity* | *Capacity* | *Capacity* |
| Lena | 467004 | 494806 | 501341 |
| Baboon | 720785 | 767058 | 790547 |
| Pepper | 482599 | 503286 | 568839 |
| Jet | 463758 | 474473 | 485443 |
| Bridge | 718743 | 720455 | 740323 |
| Scene | 593801 | 601191 | 639883 |

A series of further tests by embedding random generated bit streams were conducted on maple leave image along with other classical host images. The results obtained shows that by gradually increasing the bits capacity the quality of the image degrade drastically when applying LSB and Tseng Cheng scheme. However the results obtained from NPI based technique proves to be the best.

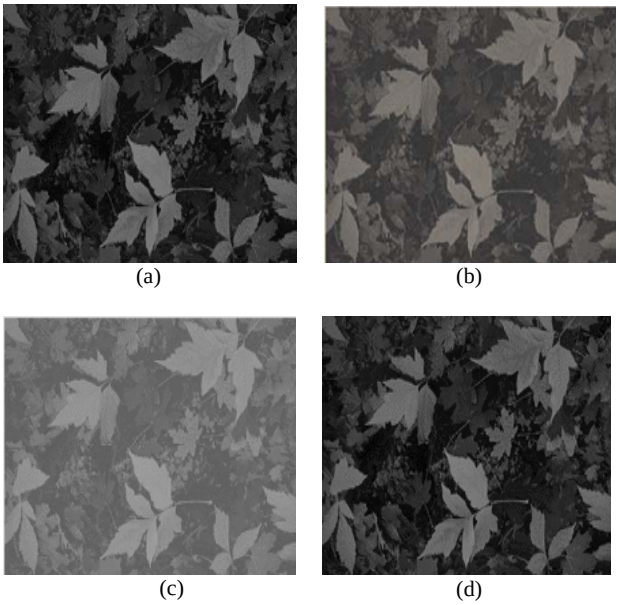


Fig. 7 Visual Comparison of: (a) original image, (b) LSB Stego-Image, (c) Tseng Cheng Stego-Image and (d) proposed NPI Scheme

Theoretical comparison was conducted on the basis of PSNR (peak signal to noise ratio) values. The proposed NPI scheme proves to be the best both on visual as well as on theoretical analysis. Table III summarizes the results.

TABLE III
COMPARISON ON PSNR VALUES

| *Host Images* | *LSB Technique* | *Tseng Cheng Technique* | *Proposed NPI technique* |
|---|---|---|---|
| | *PSNR* | *PSNR* | *PSNR* |
| Lena | 41.0053 | 40.0553 | 42.5897 |
| Baboon | 33.9879 | 31.3836 | 38.3637 |
| Pepper | 42.3743 | 39.8973 | 43.9238 |
| Jet | 39.3827 | 38.2873 | 42.5050 |
| Bridge | 36.2928 | 34.3822 | 39.3273 |
| Scene | 37.3948 | 36.6984 | 41.2382 |

## VIII. CONCLUSION

The research was aimed towards the evaluation and development of a new and enhanced data hiding technique based on NPI. The primary objective of this paper is to propose a solution that is robust, effective and to make it totally impossible 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 either on edges or uses green 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.

## ACKNOWLEDGMENT

The authors would like to thank Mr. L.M. Cheng for providing continuous support in delivering and transferring valuable knowledge of his previous research papers in this domain.

## REFERENCES

[1] Suk-Ling Li, Kai-Chi Leung, L.M. Cheng, Chi-Kwong Chan. "*Data hiding in Images by Adaptive LSB Substitution Based on The Pixel-Value Differencing*", 2006.

[2] K.M. Whelan, G.C.M. Silvestre, N.J. Hurley, "*Iterative Decoding of Scale Invariant Image Data-Hiding*", 2005.

[3] Chang, C.C., Tseng, H.-W., "*A steganographic method for digital images using side match*" Pattern Recognition Lett. 25, 1431–1437, 2004.

[4] Kaushal Solanki, Noah Jacobsen, "*Robust Image-Adaptive Data Hiding Using Erasure and Error Correction*", Dec 2004.

[5] Chan, C.K., Cheng, L.M., "*Hiding data in images by simple LSB substitution*", Pattern Recognition 37, 469–474, 2004.

[6] K. Solanki, O. Dabeer, B. S. Manjunath, U. Madhow, and S. Chandrasekaran, "*A joint source-channel coding scheme for image-in-image data hiding*," in *Proc. Int. Conf. Image Processing*, Barcelona, Spain, Sept. 2003.

[7] Yang Zhao, Patrizio, "*Dual Domain Watermarking for Authentication and Compression of Cultural Heritage Image*". March 2004.

[8] Chang, C.C., Lin, M.H., Hu, Y.-C., "*A fast and secure image hiding scheme based on LSB substitution*", Int. Journal of Pattern Recognit. And Artif. Intell. 16 (4), 399-416, 2002.