International Journal of Electrical and Computer Engineering (IJECE)
Vol. 7, No. 2, April 2017, pp. 799~809
ISSN: 2088-8708, DOI: 10.11591/ijece.v7i2.pp799-809

Hungarian-Puzzled Text with Dynamic Quadratic Embedding Steganography

Ebrahim Alrashed, Suood Suood Alroomi
Computer Engineering Department, Kuwait University, Jamal Abdul Nasser St, Kuwait

Article Info

Article history:

Received Sep 2, 2016
Revised Nov 7, 2016
Accepted Nov 21, 2016

Keyword:

Assignment problem
Dynamic embedding
Hungarian algorithm
Image quality
LSB
Security
Steganography

ABSTRACT

Least-Significant-Bit (LSB) is one of the popular and frequently used steganography techniques to hide a secret message in a digital medium. Its popularity is due to its simplicity in implementation and ease of use. However, such simplicity comes with vulnerabilities. An embedded secret message using the traditional LSB insertion is easily decodable when the stego image is suspected to be hiding a secret message. In this paper, we propose a novel secure and high quality LSB embedding technique. The security of the embedded payload is employed through introducing a novel quadratic embedding sequence. The embedding technique is also text dependent and has non-bounded inputs, making the possibilities of decoding infinite. Due to the exponential growth of and quadratic embedding, a novel cyclic technique is also introduced for the sequence that goes beyond the limits of the cover medium. The proposed method also aims to reduce the noise arising from embedding the secret message by reducing bits changed. This is done by partitioning the cover medium and the secret message into N partitions and artificially creating an assignment problem based on bit change criteria. The assignment problem will be solved using the Hungarian algorithm that will puzzle the secret message partition for an overall least bit change.

Corresponding Author:

Ebrahim Alrashed,
Computer Engineering Department,
Kuwait University,
Jamal Abdul Nasser St, Kuwait.
Email: dr_ebrahim@mac.com

1. INTRODUCTION

The The *Internet* has become the primary medium of communication and data exchange. The increasing dependency on this medium for communication, lead to data privacy, protection, and security becoming of primary concern [1]. Cryptography and encryption have always been synonymous with achieving security in the digital world. However, an alternative way of achieving digital security is through Information Hiding. While cryptography provides means to achieve secrecy of the communicated data through encryption, Information Hiding hides the actual existence of communication through deceiving covers.

Although Cryptography can provide for secure communication, there are many drawbacks to sending encrypted data stream to the network. The appearance of encrypted data would give grabbers an impulse to recover them [2]. When an encrypted data is transmitted from point A to point B, it could possibly create an incentive for intruders to explore options to either disclose the data or simply to block the transmission. Moreover, the secure communication between two points might be enough of an indication to draw conclusions from. Both Cryptography and Information Hiding achieve the intended security in data exchange but in two different manners, each with its own strengths and weaknesses.

Information hiding is the concept of hiding data in plain sight such that the actual existence of any communication between two parties is hidden. In cryptography, the secrecy of the communicated data is measured by its robustness, while in information hiding the security is measured through the stealthiness of the communicated data. Although the two methods can be easily distinguished, the security of systems may rely on a combination of both techniques with different degrees of adoption to each [3-5].

The field of information hiding encapsulates two sub-disciplines: Digital Watermarking and Steganography. Digital watermarking is concerned with intellectual property and asset protection. This is done by embedding ownership signature, called watermark, in the digital asset. Digital watermarking has been a great tool to monitor the unauthorized and the misuse of digital assets on the Internet [6]. Steganography is "the art and the science of invisible communication" [7]. This sub-discipline is concerned with hiding the communicated data. The term steganography is composed of two Greek words Stegans and Graptos which translate to covered and writing, respectively defining the term steganography as "covered writing".

In the digital world, there are many different file formats that can be used as a deceiving cover to hide the communication in steganography, and the more suitable ones are those with a high degree of redundancy [8]. Redundancy can be defined as the bits that provide far greater accuracy than necessary for the Human Visual System (HVS). Thus, Image, video and audio file formats are three widely used file formats for steganography due to their high degree of redundancy. Effectiveness of image steganography is based on three essential attributes: security, imperceptibility, and message capacity. Security either refers to how stealthy the cover image is from being exposed containing a secret message or how robust the embedded secret message is being revealed. Imperceptibility is the degree of quality of the cover image after it is embedded with the secret message, also called the stego image. Message capacity is the amount of secret message an image can contain.

Secret message embedding methods in steganography can be generally classified into two embedding classes: frequency domain and spatial domain. Least-Significant-Bit (LSB) is one of the widely used embedding technique in spatial domain, however, it is prone to attacks as illustrated in [9], [10]. When a secret message is suspected, the payload can be easily extracted from the image, as the LSB embedding is straightforward.

In this paper, we will propose a novel technique that improves both the security and the imperceptibility of the traditional LSB insertion. The security is improved through the use of an un-patterned embedding based on quadratic equations. The improvement in the imperceptibility is achieved through artificially creating an assignment problem which will facilitate the selection of the least noisy (bit change) combination of the partitioned cover image and secret message parts. The selection of the least bit change is done through the use of an optimization algorithm called Hungarian algorithm.

In this paper, we propose a novel method which, when compared to LSB insertion, provides more security for the embedded secret while maintaining the quality of the cover image. The contribution of this paper is through the use of dynamic irregular and un-patterned embedding as well as the partitioning of both the cover image and the secret message and finding the best assignment by employing the Hungarian algorithm.

The organization of the paper is as follows. A survey of related work in the literature is given in Section 2. In Section 3, we will provide a brief introductory to the assignment problem and how to apply the Hungarian algorithm. Section 4 will showcase the proposed method. Following that, Section 5 will analyze the security the proposed embedding. In Section 6, we will present our experimental results. Finally, we will provide our conclusion in Section 7.

2. RELATED WORK

There have been several different approaches to secure the traditional LSB embedding proposed in the literature. Bailey and Curran in [11] proposed an LSB method called Stego Color Cycle (SCC) which alternates bit embedding in different channels. While the traditional LSB embedding scheme embeds on all three Red, Green, and Blue (RGB) channels, the SCC propose to hide the secret message by embedding in one channel in each cycle. One of the major limitation of this approach is the systematic pattern it creates when embedding which makes it easier to decode once the cyclic embedding pattern is found. Jamil et al. in [12] suggested using randomization which provides a security layer over the cyclic embedding approach to overcome the flaws of SCC. In [13], Bhattacharyya et al. proposed which embeds secret messageby modulating adjacent DCT coefficient differences.

Parvez et al. in [14] proposed a Pixel Indicator Technique (PIT), which divides the channels of the RGB image into data channels and indicator channel. For example, if the indicator channel is the 2-LSB bits

(the 7th and 8th bit). The permutation of 0 and 1 decides whether to embed or not bad on a set of applied rules. PIT security was further improved in [15] by using stego key for channel selection.

In [16], Bandyopadhyay et al. proposed a secure spatial domain method based on encrypting the secret message bits using chaos theory. The cover image is also divided into parts for added security. Their results also showed a slight improvement of both image quality and image fidelity.

Liu and Koenig in [17] developed a video encryption algorithm inspired by the children's puzzle game. A picture is split into pieces and are placed in disorder, which makes the original picture unrecognizable at first glance. Liu and Koenig's work inspired our puzzling approach of disordering the payload when embedding into the cover image. Akhtar et al. in [18] enhanced the security of the traditional LSB by relying on the popular RC4 algorithm to securely embed the payload bits into the image with a random sequence produced from the algorithm. In addition, the noise from the embedded bit is reduced by a novel technique called bit inversion. The technique divides the stego image into 4 classes based on the 6th and 7th bit permutation. In each class, a check will be done based on flipping the LSB (8th bit) or not, whichever has less bit changes (flipped or not) is used.

Sun et al. in [19] showcased how the security of LSB embedding can be improved based on Fisher Information. Their experiments adopted ROC curves to evaluate the security improvement of their embedding against the traditional Bashardoost et al. [20] proposed a secure LSB embedding through the use of Vigenere cipher method. The advantage of Vigenere cipher is that it does not produce a cipher text that is longer than the original text.

How would you assign 1 job when you have 1 worker available? Immediately the answer would be to assign the single job to that single worker. Now imagine having 2 jobs and 2 workers where each worker has a unique cost to perform each job. There will be 4 combinations to consider. The idea is to find the combination the results in the least possible cost. Such a problem grows exponentially with the increase of workers and job which makes it unfeasible to consider each combination every time. Such problems are known as the Assignment problem and without optimizing the problem, *n*! trials need to be conducted to arrive at the solution.

An expeditious way to solve the assignment problem, without going through all possible combinations, is to use Kuhn's Hungarian algorithm [21] to arrive at a solution in polynomial time. The steps to perform the Hungarian algorithm is as follows which is also illustrated in Figure 1 and Figure 2.

(1) Step 1: For each row, subtract the minimum row value from all entries in that row, and as a result, all rows have at least one zero entry, and all entries of the matrix are nonnegative.
(2) Step 2: For each column, subtract the minimum row value from all entries on that column, and as a result, all the rows and columns of the matrix have at least one zero entry, and all the matrix entries are still nonnegative.
(3) Step 3: Draw a line across the rows and columns in a way that all the zeros in the matrix are covered with minimum lines used.
(4) Step 4: If the number of lines drawn is equal to n then the optimality test is complete, however if it is lower we continue to step 5.
(5) Step 5: Find the smallest uncovered entry, subtract it from all uncovered entries and add it to the entries that are covered twice. Then go back to step 3.

Note that there are times when two different assignments can provide the same total minimum value.

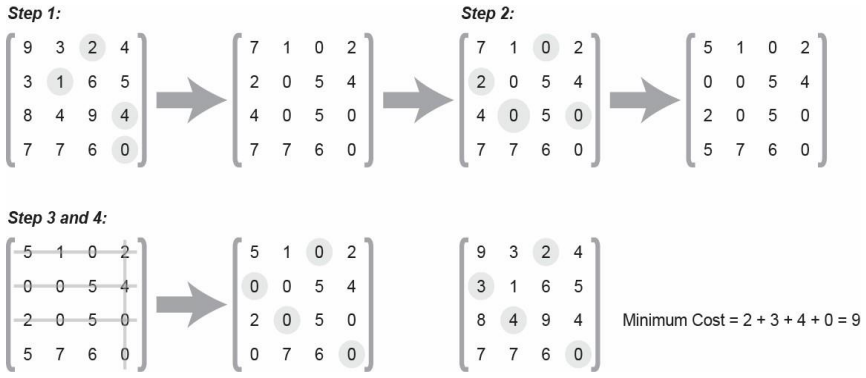

Figure 1. Hungarian Algorithm

3. PROPOSED SCHEME

In this section, the Hungarian-Puzzled Text with Dynamic Quadratic Embedding (HPT-DQE) scheme is described, the main components of which are shown in Figure 2. The inputs of HPT-DQE are the secret message that needs to be communicated, and the medium used for embedding this secret message, which is called the cover image. The outputs of the proposed scheme are the stego image and a key that contains information to extract the secret message.

3.1. Partitioning Cover Image and Secret Message

HPT-DQE will first divide both the cover image and the secret message into N partitions, where N is a positive number with a maximum value equal to the number of pixels in the cover image. Let us denote the secret message partitions and the cover image partitions as smp and cip respectively, which are interchangeably called puzzle pieces. Following the partitioning step, the proposed scheme creates a reference list for all the smp-cip combinations.

3.2. Quadratic Equation

In traditional LSB embedding, bits are usually embedded in all three RGB channels of each pixel in the color image. However, in our proposed method several embedding techniques can be used for each image puzzle or partition. The following are the types of embedding supported by our proposed LSB Embedding Methods component:

Note that some of the selected embedding techniques skip a channel like RG and GB, while others skip two channels, e.g. R only embed in the red channel. Also some embedding technique differ in the order of embedding, i.e. RGB and RBG. These different techniques are tested against each partition to find the one with the least bit change.

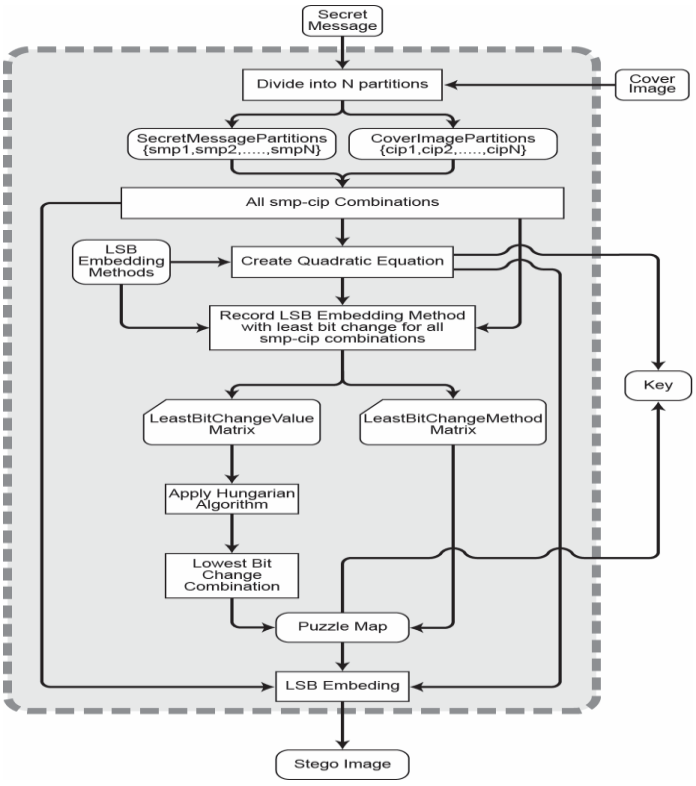

Figure 2. Proposed Method: Hungarian-Puzzled Text with Dynamic Quadratic Embedding

3.3. LSB Embedding Techniques

The proposed embedding scheme has two main elements, a secret message of length *m* bits and a cover image partition of length *c* bits. A unique random quadratic equation is generated to determine the secret message bit embedding locations within the cover image partition. Let *A*, *B*, *R* and *Z* be randomly

chosen integer constants of zero or integer values. Alternatively, these constants can be previously agreed between the sender the receiver. The embedding locations are given by the following equation:

$$BitEmbedLoc(i) = |A \cdot X_1^2(i) + B \cdot X_2(i) + D(i-1) \cdot S| \quad (1)$$

At first glance, Equation 1 models the structure of a quadratic equation, however, to sway away from the deterministic quadratic pattern, we included D as a message dependency factor, where D is given by:

$$D(i) = Dec(E_{i-1}) + H \quad (2)$$

where E_i is the i^{th} embedded character, *Dec*(.) is a function that returns the decimal value of character *E*, and *H* is a pre-shared random constant.

This definition of *D* introduces a message dependency in Equation 1 that enables the embedding of different secret messages with the same constants in different locations. This is true since different messages will result in different values of *D* in Equation 2, which will subsequently result in different values of embedding bit locations from Equation 1, even if the values of *A*, *B*, *R* and *Z* are the same in all cases. Note that in Equation 1 the values of the constants are permitted to be positive or negative, hence an absolute value in Equation 1 is employed, as we cannot have negative bit locations. This flexibility increases the robustness of the proposed scheme as shall be discussed in section IV.

3.4. Cyclic Embedding

Since Equation 1 emulates a quadratic equation, it is easy to see that the bit embedding location values generated will quickly exceed the limits of the cover image partition. To resolve this problem, we introduce an enhancement to the embedding equation we call, *cyclic embedding*. If the quadratic equation produces a location that exceeds the number of pixels in an image partition, we repeatedly subtract the maximum number of pixels in the partition from the generated location value until we get a number that is within the limits of the partition.

Figure 3 illustrates cyclic embedding on an image of 16 partitions as shown in Part *a*. For each partition, the generated embedding location may eventually exceed the maximum capacity as indicated in Part *b* by the red border dots. Parts *c* and *d* show the cycled positions of these locations in comparison to their original location. Finally, the partition will accommodate all out of range bits in their new positions after cycling them, as shown in Part *e*.

A location value generated by cyclic embedding, will always be checked for its uniqueness. In the case the location generated at iteration *i* has already been used, the scheme will run another iteration, *i* + 1, to generate a new embedding location.

3.5. Puzzling the Message with the Hungarian Algorithm

Partitioning both the secret message and the cover image into *N* partitions will result in *N* ! smp-cip combinations, each combination can use any one of the 15 embedding techniques listed in Table 1. Finding the tuples <*smp, cip, embeddingtechnique* > that would result in the minimum noise (bit change) is a complex assignment problem.

To reduce the complexity of this assignment problem, our proposed scheme tries all embedding techniques on every smp-cip combination and records the one technique which generates the lowest bit change. We define two *N*x*N* matrices; *LeastBitChangeMethod* and *LeastBitChangeV* alue for which the rows represent the cover image partition number and the column represent the secret message partition number. Matrix *LeastBitChangeMethod* holds the number of the technique, as shown in Table 1, with the lowest bit change value for every smp-cip combination, and matrix *LeastBitChangeValue* holds the corresponding bit change value generated by that technique. For example, *LeastBitChangeValue* (3,4) represents the least bit change value when embedding the fourth secret message partition (smp4) in the third cover image partition (cip3).

LeastBitChangeValue matrix in this case represents an assignment problem that we artificially created. The optimal assignment (or puzzling) of message partitions to image partitions would produce the least overall bit change solution. We use the Hungarian algorithm to arrive at such optimal assignment. The results generated by the Hungarian algorithm are cross-referenced with the *LeastBitChangeM ethod* matrix to produce the Puzzle Map component shown in Figure 2 which contains the embedding technique used and the secret message partition for each image partition. Using this puzzle map and the produced sequence of bit

embedding locations from the quadratic equation, the secret message is embedded in the cover image, and the stego image is produced and is sent to the receiver.

Table 1. Embedding Techniques Employed

Technique Number	Embedding Technique
1	R
2	G
3	B
4	RG
5	GR
6	RB
7	BR
8	GB
9	BG
10	RGB
11	RBG
12	GRB
13	GBR
14	BRG
15	BGR

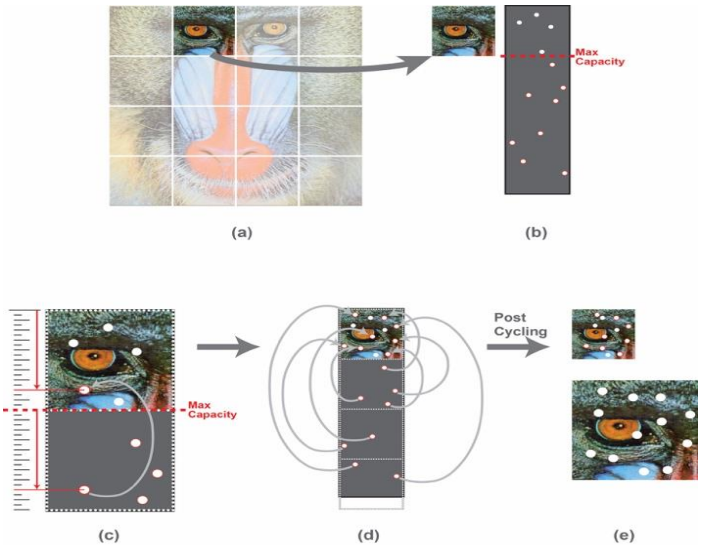

Figure 3. Cyclic Embedding

A key, which contain the puzzle map and the generated quadratic equation inputs, is sent to the receiver, using which the receiver can decode and rearrange the secret message partitions to reveal the embedded secret message. As previously discussed and shown in Figure 2. The first step in our proposed scheme involves partitioning both the cover image and the secret message into N partitions. Figure 4 illustrates the partitioning step, where in this example, both the cover image and the secret message are partitioned into 4 parts, upon which a 4x4 matrix of all possible smp-cip combinations is created. Recall that the rows in the matrix represent the cover image partition number and the columns is the secret message partition number.

After partitioning, the inputs of the quadratic equation are randomly chosen. Using the quadratic equation along with smp-cip combination matrix, every possible LSB embedding method (see Table1) is tested for each combination. The method yielding the least bit change value is recorded in the *LeastBitChangeValue* matrix and the method number is recorded in the *LeastBitChangeMethod* matrix. This step is illustrated in the first part of Figure 5.

The Hungarian algorithm is then applied to *LeastBitChangeValue* matrix, resulting in the following combination that generates the least bit change as shown in matrix *HungarianBestMatch* in the second part of Figure 5):

a. 1*st* cip → 4*th* smp → Using embedding method 12
b. 2*nd* cip → 2*nd* smp → Using embedding method 7
c. 3*rd* cip → 1*st* smp → Using embedding method 5
d. 4*th* cip → 3*rd* smp → Using embedding method 9

Finally, the optimal assignment is cross-referenced with *LeastBitChangeMethod* to generate the puzzle map. This is done through performing an element-wise product of both *HungarianBestMatch* and *LeastBitChangeMethod* matrices (also called Hadamard Product) as shown in the last part of Figure 5. The puzzle map will be used by the sender to embed the secret message onto the cover image, and subsequently will be sent to the receiver along with the selected quadratic equation inputs to guide the receiver into decoding the secret message from the stego image.

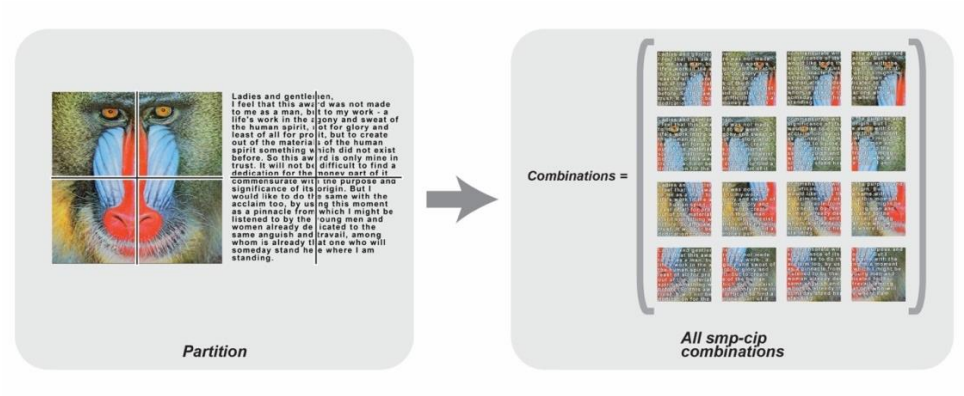

Figure 4. Hungarian Algorithm

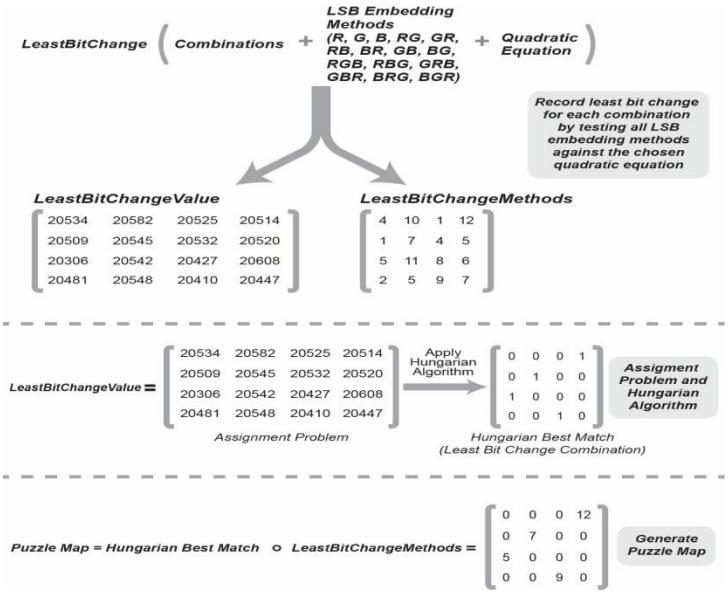

Figure 5. Generating Puzzle map after partitioning

4. SECURITY ANALYSIS

In this section, we analyze the security of the proposed quadratic embedding. We first investigate the robustness of the generated embedding location sequences by looking at the general parameters that make up the embedding location formula. We also examine the time complexity of decoding the secret message from the cover image.

The quadratic embedding equation described in section C. contains three main factors that ensure the randomness and irregularity of bit embedding locations, increasing the robustness of the proposed scheme against unauthorized efforts to reveal the secret message. The three factors are cyclic embedding, unbounded input length, and message dependency. First recall that, when at iteration *i* the bit embedding location formula generates a location value greater than the maximum available capacity of the cover image c, c is subtracted from the generated location value iteratively until the resulting value is within the range [1, c], in a looping fashion we called cyclic embedding. If the resulting cyclic embedding value lies on a previously embedded location, the proposed scheme will skip to the following *i*+1 embedding location until an unused location is found. This cyclic process yields an irregular and patternless embedding locations which increases the difficulty of guessing the message bit locations.

The second factor is the unbounded inputs of the equations. The absolute value on the right-hand side of the embedding formula allows for the inside calculations to result in either negative or positive values since they are ultimately converted to positive values. The infinite possible values of the embedding location formula parameters also increase the degree of randomness of formula itself which renders it infeasible to guess.

Finally, the message dependency adds another layer of randomizing the produced bit embedding location sequence such that two identically selected quadratic equation inputs will produce two different embedding location sequences when the secret messages are different. This further increases the adversary's difficulty in guessing the bit embedding locations of the secret message.

The number of all possible bit-embedding locations of *m* message bits in a pool of *c* image bits is finite. However, there are an infinite number of parameters that can be selected for the embedding formula and these parameters also cover all possible locations. Therefore, multiple formula parameter selections can result in the same observed locations of bits.

Since as per our proposed scheme, an adversary would need to have the embedding location formula and the secret message itself to obtain the bit embedding locations, it is then more efficient for an adversary to recover the secret message by brute forcing all possible bit locations. The number of possible bit locations of a secret message of size *m* bits into a cover image partition of length c bits can be determined by combination theory as follow:

$$Numberoftrials = \frac{c!}{(c-m)!} \quad (3)$$

Since the size of the secret message is unknown to an adversary, recovery of the secret message required a total number of decoding trials that is the sum of all trials for m for sizes ranging from 1 to *c* bits.

$$TotalNumberofTrials = \sum_{m=1}^{c} \frac{c!}{(c-m)!} \quad (4)$$

Thus, robustness of the proposed embedding security scheme significantly increases as the length of the cover image partition *c* increases.

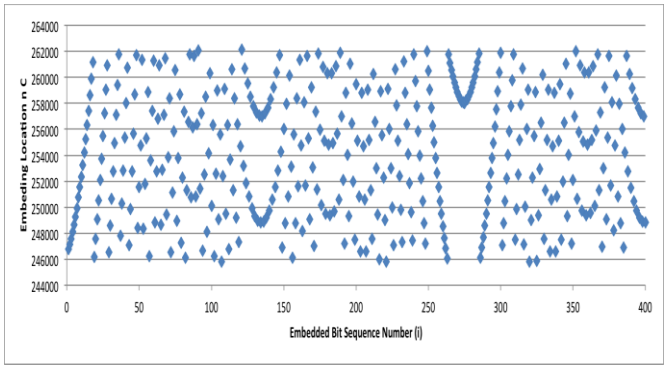

Figure 6. Irregularity of generated bit embedding locations

To illustrate the randomness and irregularity of the bit locations generated by the proposed method, we generated 400 consecutive bit embedding locations using Equation 1 an plot them in Figure 6, which clearly shows the irregularity in shape and position of the generated bit embedding locations. This produced

irregularity and randomness increase the degree of difficulty for an adversary to recover the embedded secret message. With unknown message length and an image of size 34 bits, it would take, according to equation 4, 8.27x1038 trials to decrypt the secret message. This is equivalent to 2.4 times the number of decryption trials needed for a 128-bit key used in many applications today. The number of decryption trials needed to decrypt the original message increases exponentially as the size of the cover image increases as shown in Figure 7.

Unlike password guessing, where a computer can typically perform more than 100 trials per second, extracting bit locations from a colored image is a more complex task and takes a longer time on average. Extracting 50% of the LSB bits of the popular Mandrill image requires 0.35 second in MATLAB when using a machine equipped with Intel Core i7-4770k and 16 GB DDR3 RAM operating a 64-bit version of Windows 7. This makes it possible to do 2.86 extractions per second. Converting theses values to years and using the number of trials from Figure 6, we can estimate the number of years required for each value of c as shown in Figure 8. Note that the values are halved as on average the adversary will only have to try half of the trails to reveal the secret message.

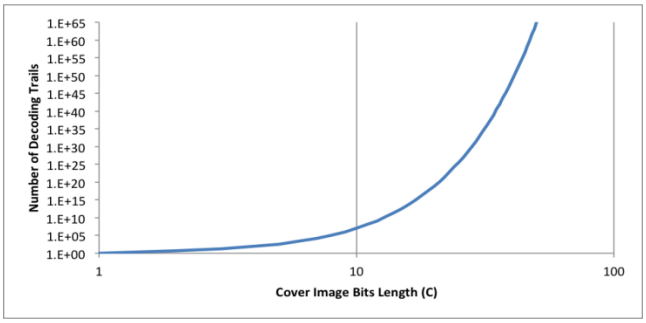

Figure 7. Number of possible placements as a function of cover image partition size

Analysis of HPT-DQE has shown that the proposed embedding scheme produces an un-patterned embedding location sequence from an infinite set of inputs that is infeasible to guess by an adversary. Brute forcing all possible values quickly becomes a matter of years (or more) in time to reveal the embedded secret message. The robustness of the proposed embedding method is increased exponentially as the cover image size is increased.

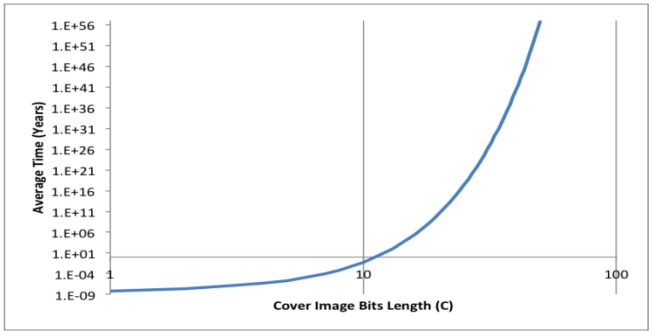

Figure 8. Estimated brute forcing time as a function of cover image partition size

5. EXPERIMENTAL RESULTS

In this section, we will evaluate the amount of noise (bit changes) saved by the use of Hungarian algorithm alongside selecting the least bit change embedding method for each section (see Table 1). In addition, the Signal to Noise Ratio (SNR) and the Peak Signal-to-Noise Ratio (PSNR) of the produced stego image is measured. The results of HPT-DQE are compared with the bit-inversion technique proposed in [16] and the traditional LSB embedding.

The conducted experiments in this section will use Act I of Macbeth script play by William Shakespeare as a secret message. The script contains 21,808 characters (including space and new line) with a

total size of 23KB. For the cover image, the popular mandrill image that is featured in our previous illustrations is used. The dimension of the image is 256x384 with an LSB capacity of 256*384*3 = 294,912 bits. In total, the embedded play script amounts for 52% of the cover image LSB capacity.

Figure 9 compares the amount of bit changed after embedding the script play into the cover image. The comparison is between the proposed method (using different numbers of partitions), bit-inversion technique [16] and the traditional LSB method.

The proposed scheme outperforms both the traditional LSB and the Bit-Inversion technique. Notice that the noise introduced by the embedded secret message is further reduced as the number of partition is increased. An increased number of partitions facilitates more noise reduction because the Hungarian algorithm will have more options to choose from to select the least noisy smp-cip combination. Looking closely at the linear relation between partition number and the cover image, Figure 10 computes the correlation coefficient and the relation for the six parts tested. The six measured parts have a linear relation of y=-58.939x + 81676 with a high measured R2 value of 0.9899. As such, using the equation provided, one can estimate the number of bit change as function of partition number with good reliability. Both PSNR and SNR values are also from the produced stego image. Table 2 lists these values alongside the bit change count for the three methods.

As expected, the three values (Bit change, PSNR and SNR) are steadily improving as the number of partition increases. This is mainly due to the decrease of the noise produced from the secret message.

Table 2. PSNR Comparisons

Method		Bit Change	SNR	PSNR
Traditional LSB		82419	52.6177	57.9271
Bit-inversion (No Flip)		82685	52.6037	57.9131
Bit-inversion (With Flip)		82309	52.6235	57.9329
HPT- DQE	5 Partitions	81537	52.6645	57.9739
	10 Partitions	81028	52.6916	58.0011
	20 Partitions	80389	52.726	58.0354
	30 Partitions	79814	52.7572	58.0666
	40 Partitions	79361	52.7819	58.0913
	50 Partitions	78790	52.8132	58.1227

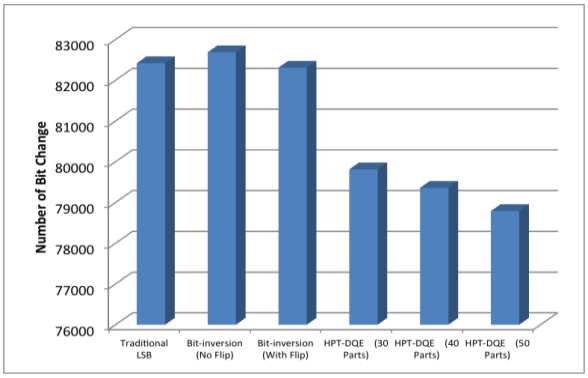

Figure 9. Bit change of proposed method in comparison with other methods

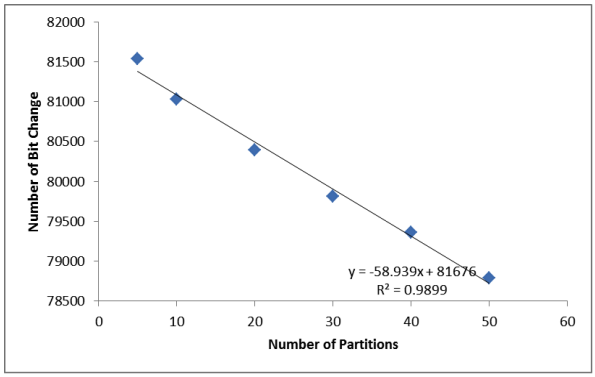

Figure 10. Linear relation and correlation coefficient values between partition number and bit change.

6. CONCLUSION

In this paper, we have developed a novel technique to elevate the security and reduce the noise of the LSB embedding. A new approach of secure embedding is presented through relying on an un-patterned quadratic embedding sequence with unbounded input parameters. The approach is analyzed to have a

security that exceeds the standard 129-bit encryptions used in many applications today. Solving an artificially created assignment problem with an optimized solution of the Hungarian algorithm found to be an effective technique. The results show a reduction of the noise produced from the secret message and an improvement on both the PSNR and SNR. The noise reduction and the image quality metrics are found to have a reliable relation with the number of partitions used.

REFERENCES

[1] Yan L, Zhang Y, Yang LT, Ning H, "The Internet of things: from RFID to the next-generation pervasive networked systems", CRC Press, 2008.

[2] Chang CC, Hsiao JY, Chan CS, "Finding optimal least-significant-bit substitution in image hiding by dynamic programming strategy", Pattern Recognition, 2003 Jul 31; 36(7): 1583-95.

[3] Saraireh S, "A Secure Data Communication system using cryptography and steganography", *International Journal of Computer Networks & Communications*. 2013 May 1; 5(3): 125.

[4] Gupta S, Goyal A, Bhushan B, "Information hiding using least significant bit steganography and cryptography", *International Journal of Modern Education and Computer Science*. 2012 Jun 1; 4(6): 27.

[5] Sarmah DK, Bajpai N, "Proposed System for data hiding using Cryptography and Steganography", *International Journal of Computer Applications*. 2010 Oct; 8(9): 7-10.

[6] Minghui D, Lanying Z, Zhancheng L, "An Information Hiding Algorithm Based on Improved S-Hough Transformation*", Indonesian Journal of Electrical Engineering and Computer Science*. 2014 Feb 1; 12(2): 1109-15.

[7] Morkel T, Eloff JH, Olivier MS, "An overview of image steganography", *Proceedings of the 5th Annual Information Security South Africa Conference (ISSA2005)*, Sandton, South Africa, Jun 2005; 1-11.

[8] Batra S, Khattra HK, "An Improved Data Transfer Technique Using Steganography with Watermarking and Visual Cryptography", *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*. 2013 Dec; 3(7): 144-147.

[9] Westfeld A, Pfitzmann A, "Attacks on steganographic systems", *International workshop on information hiding*, 1999, Springer Berlin Heidelberg. Sep 29; 61-76.

[10] Fridrich J, Long M, "Steganalysis of LSB encoding in Color Images", *IEEE International Conference on Multimedia and Expo (ICME)*. 2000; 3: 1279-1282.

[11] Bailey K, Curran K, "An Evaluation of Image based Steganography Methods", *Multimedia Tools and Applications*. 2006 Jul 1; 30(1): 55-88.

[12] Bhattacharyya S, Khan A, Sanyal G, "DCT Difference Modulation (DCTDM) Image Steganography", *International Journal of Information and Network Security*. 2014 Jan 1; 3(1): 40.

[13] Muhammad K, Ahmad J, Rehman NU, Jan Z, Qureshi RJ, "A Secure Cyclic Steganographic Technique for Color Images using Randomization", *Technical Journal*, 2015 Feb 27; 19(3): 57-64.

[14] Parvez MT, Gutub AA, "RGB Intensity based Variable-Bits Image Steganography", *IEEE Asia-Pacific Services Computing Conference APSCC'08*, 2008 Dec 9; 1322-1327.

[15] Gutub AA, "Pixel Indicator Technique for RGB Image Steganography", *Journal of Emerging Technologies In Web Intelligence*. 2010 Jan 2; 2(1): 56-64.

[16] Bandyopadhyay D, Dasgupta K, Mandal JK, Dutta P, "A Novel Secure Image Steganography Method based on Chaos Theory in Spatial Domain", *International Journal of Security, Privacy and Trust Management (IJSPTM)*. 2014; 3(1): 11-22.

[17] Liu F, Koenig H, "A Novel Encryption Algorithm for High Resolution Video", *ACM Proceedings of the International Workshop on Network and Operating Systems Support for Digital Audio and Video* 2005 Jun 13; 69-74.

[18] Akhtar N, Johri P, Khan S, "Enhancing the Security and Quality of LSB based Image Steganography", *5th IEEE International Conference on Computational Intelligence and Communication Networks (CICN)*, 2013 Sep 27; 385-390.

[19] Sun Y, Niu D, Tang G. Gao Z, "Optimized LSB Matching Steganography based on Fisher Information", *Journal of Multimedia*, 2012 Jan 8; 7(4): 295-302.

[20] Bashardoost M, Sulong G, Gerami P, "Enhanced LSB Image Steganography Method using Knight Tour Algorithm, Vigenere, Encryption and LZW Compression", *International Journal of Computer Science Issues (IJCSI)*. 2013; 10(2): 221-7.

[21] Kuhn HW, "The Hungarian Method for the Assignment Problem", Naval research logistics quarterly. 1955 Mar 1; 2(1-2): 83-97.