HIDDEN ENCRYPTED TEXT BASED ON SECRETE MAP EQUATION AND BIOINFORMATICS TECHNIQUES

1 ALAA KADHIM F., 2 RASHA SUBHI ALI

1 Computer Sciences Department, University of technology/ Baghdad, Iraq
dralaa_cs@yahoo.com

2 Department of Computer Techniques Engineering, AL Nisour University College/ Baghdad, Iraq
danafush@Gmail.com

ABSTRACT

The speedy development in information technology desires the secure transmission of confidential information that gets an excellent deal of attention. Therefore, it's necessary to use effective methods to reinforce information security. Steganography is one of the leading technologies getting utilized around the world for a long time. Biotechnological methods can be used for cryptography to improve security of data. Steganography is the act of hiding messages inside an image. Combining these two methods is a topic of high relevance since secure communication is inevitable for mankind. This research presents an analysis of steganography, by using Least Significant Bit (LSB), DNA computing, and creating a secret map for hiding data. The DNA computing was used to encrypt secret data, LSB was utilized to add the encrypted data into least significant bits of the cover and the secret map was utilized to specify the location of hiding data. The same equation must be used by the sender and the receiver to create the secret map and the creation for this map depends on the shared key.

Keywords- Least Significant Bit (LSB), DNA Computing, Secret Map, Steganography.

1. INTRODUCTION

Data security has become a big resource nowadays for the effective operations of the various demands of any organization. One among the most necessary demands of those networks is to supply secure transmission of knowledge from one place to another. Cryptography is one among the mechanisms that give most secure way to transfer the sensitive info from sender to supposed receiver. Its major aim is to create sensitive info unclear to all or any totally different, except the supposed receiver [1]. Data hiding is the method of embedding info within a data source without ever-changing its perceptual quality. Information hiding is that the art and science of writing hidden messages in a way that no-one except the sender and supposed recipient even realizes there is a hidden message. The goal of steganography is to hide a message m in any audio or video (cover) data d, to induce new data d, practically indistinguishable from d, by people, in such how that associate snooper cannot discover the presence of m in d [2]. Image Steganography for data Security 1st we tend to write data with the assistance of desoxyribonucleic acid algorithmic rule then we tend to choose secrete image that we would like to transfer with data and that secrete image is additionally used to hide encrypted information [3]. Shortly, the cryptography used for protecting the content of messages; steganography utilized for concealing its terribly existence. With the inclusion of newer cover media and stronger algorithms we are able to deal with the most recent attacks [4]. The advantage of hiding data over cryptography alone is that the intended secret message doesnot attract attention to itself as an object of security. Desoxyribonucleic acid secret writing is one in every of the foremost secure cryptography and decryption technique. Desoxyribonucleic acid (Deoxyribonucleic Acid). DNA is considered as the genetic pattern of living or existing creatures. All individual body cells have a complete set of DNA. DNA is a polymer madeout of monomers called DeoxyriboseNucleotides. A DNA sequence consists of four nucleic acid bases A (adenine), C (cytosine), G (guanine), T (thymine), where A and T are complementary, and G and C are complementary within the secretwriting algorithmic the desoxyribonucleic acid writing rule
performed as a part of secret key and DNA addition and subtraction operations is employed to confuse the Desoxyribonucleic Acid sequences. Eight deoxyribonucleic acid map rules are used to satisfy the Watson Crick complement rule the detailed of deoxyribonucleic acid conversion was shown in table 1 [5]. The number of possible coding patterns is illustrated in table 1. Steganography incorporates a range of drawbacks compared to encryption. It needs to have overhead to hide a comparatively few bits of information, though utilizing some scheme like that proposed within the preceding paragraph could build it more effective or else a message can be 1st encrypted and then hidden by utilizing steganography[4]. The main purpose of this study is hiding secret data by using steganography, LSB and DNA cryptosystem, which is a new science in information security. In this research a new the cryptography and steganography methods was used and implemented to encrypt the data that was required to be hiding in the cover image. The objective of stenography is to avoid drawing suspicion to the transportation of the secret message between sender and receiver.

A secure data transmission is made using cryptography and steganography. A combination of both these two techniques results in appearing a highly secured method for data communication [6]. Steganography involves four steps [7]:

1. Choice of the cover media within which the info are going to be hidden.

2. The secret message or data that's required to be masked within the cover image.

3. A function which will be wont to hide information within the cover media and its inverse to retrieve the hidden data.

4. An optional key or the password to certify or to hide and unhide the info.

Table 1: Eight Map Rules [5]

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>A</td>
<td>A</td>
<td>C</td>
<td>C</td>
<td>G</td>
<td>G</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>1</td>
<td>C</td>
<td>G</td>
<td>A</td>
<td>T</td>
<td>A</td>
<td>T</td>
<td>C</td>
<td>G</td>
</tr>
<tr>
<td>2</td>
<td>G</td>
<td>C</td>
<td>T</td>
<td>A</td>
<td>T</td>
<td>A</td>
<td>G</td>
<td>C</td>
</tr>
<tr>
<td>3</td>
<td>T</td>
<td>T</td>
<td>G</td>
<td>G</td>
<td>C</td>
<td>C</td>
<td>A</td>
<td>A</td>
</tr>
</tbody>
</table>

Table 2: addition and subtraction operations on DNA nucleotides

<table>
<thead>
<tr>
<th></th>
<th>+</th>
<th>A</th>
<th>T</th>
<th>C</th>
<th>G</th>
<th>-</th>
<th>A</th>
<th>T</th>
<th>C</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>T</td>
<td>G</td>
<td>A</td>
<td>C</td>
<td>A</td>
<td>C</td>
<td>G</td>
<td>A</td>
<td>T</td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>G</td>
<td>C</td>
<td>T</td>
<td>A</td>
<td>T</td>
<td>A</td>
<td>C</td>
<td>T</td>
<td>G</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>A</td>
<td>T</td>
<td>C</td>
<td>G</td>
<td>C</td>
<td>G</td>
<td>T</td>
<td>C</td>
<td>A</td>
<td></td>
</tr>
<tr>
<td>G</td>
<td>C</td>
<td>A</td>
<td>G</td>
<td>T</td>
<td>G</td>
<td>T</td>
<td>A</td>
<td>G</td>
<td>C</td>
<td></td>
</tr>
</tbody>
</table>

Table 3: XOR operation on DNA nucleotides [8]

<table>
<thead>
<tr>
<th></th>
<th>+</th>
<th>A</th>
<th>T</th>
<th>C</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>A</td>
<td>T</td>
<td>C</td>
<td>G</td>
<td></td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>C</td>
<td>G</td>
<td>A</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>C</td>
<td>G</td>
<td>A</td>
<td>T</td>
<td></td>
</tr>
<tr>
<td>G</td>
<td>G</td>
<td>C</td>
<td>T</td>
<td>A</td>
<td></td>
</tr>
</tbody>
</table>

Steganography hides the existence of a secret message and within the best case no one will see that eachparty is communicating in secret. This makes steganography applicable for some tasks that secret writing aren’t, comparable to copyright marking. Table four shows a comparison of varied techniques for communicating in secret [9].

TABLE 4: COMPARISON OF SECRET COMMUNICATION TECHNIQUES

<table>
<thead>
<tr>
<th>SECRET COMMUNICATION TECHNIQUE</th>
<th>CONFIDENTIALITY</th>
<th>INTEGRITY</th>
<th>UMREMOVABILITY</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENCRYPTION</td>
<td>YES</td>
<td>NO</td>
<td>YES</td>
</tr>
<tr>
<td>DIGITAL SIGNATURES</td>
<td>NO</td>
<td>YES</td>
<td>NO</td>
</tr>
<tr>
<td>STEGANOGRAPHY</td>
<td>YES/NO</td>
<td>YES/NO</td>
<td>YES</td>
</tr>
</tbody>
</table>

2. IMAGE STEGANOGRAPHY

The most wide used technique these days is hiding secret messages into a digital image. This steganography method exploits the failure of the human visual system (HVS). The HVS cannot detect the difference in luminance of color vectors at set of pixels color. Foreexample: a24-bit picture can have 8bits, representing every 3 color values (red, green, and blue) at every pixel. If we have a
tendency to take into account just the blue there'll be two totally different values of blue. The distinction between 11111111 and 11111110 within the value of blue intensity is probably going to be undetectable by the human eye. Hence, if the terminal recipient of data is nothing however human the least (HVS) then the least significant Bit (LSB) are often used for one thing else excluding color information. The only approach to hiding information within a picture file is termed least significant bit (LSB) insertion [10].

The LSB is that the lowest bit in an exceedingly series of numbers in binary. E.g. within the binary number: 10110001, the least significant bit is much right one. The LSB primarily based Steganography is one in all the steganographic strategies, utilized to embed the key information into the least significant bits of the pixel values in an exceedingly cover image. E.g. 240 can be hidden within the 1st eight bytes of 3 pixels in an exceedingly 24 bit image [11].

**PIXELS:**

00100111 11101001 11001000  
00100111 11001000 11101001  
11001000 00100111 11101001  
240 : 011110000

**RESULT:**

00100110 11101001 11001001  
00100111 11001001 11101000  
11001000 00100110 11101000

Here the number 240 is embedded into first eight bytes of the grid and only 6 bits are changed.

3. **SOME METRICS FOR PICTURE QUALITY EVALUATION (PQE) [12] [10]**

For the sake of measuring the quality of the encrypted image must be using the Picture Quality Evaluation (PQE), which is depicted in below points [13] [14]:

1. **Mean Square Error (MSE):** It is the measurement of the square of error, the error is the amount by which the original image's pixel value is different to the encrypted image's pixel value.

\[
MSE = \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{[f(i,j) - f'(i,j)]^2}{MN}
\]

Where, \(M\) and \(N\) represent the image's height and width respectively. \(f(i,j)\) is the \((i,j)\)th pixel value of the original image and \(f'(i,j)\) is the \((i,j)\)th pixel value of the decrypted image.

2. **Peak Signal to Noise Ratio (PSNR):** Represents the ratio between the maximum probable signal power and the power of corrupting noise which influences the fidelity of its representation. PeakSignal to Noise Ratio is typically represented in terms of the logarithmic decibel. PeakSignal to Noise Ratio is calculated by:

\[
PSNR = \frac{10}{\frac{1}{2^n - 1}}
\]

3. **Average Difference (AD):** It is the average error measurement between both images the original and the decrypted one and is calculated as:

\[
AD = \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{[f(i,j) - f'(i,j)]^2}{MN}
\]

Where: \([f(i,j) - f'(i,j)]\) correspond to the error between the pixel value of the original image and the pixel value of the decrypted image at height \(i\) and width \(j\).

4. **Maximum Difference (MD):** It is the measurement of the maximum error value between both images the original and the decrypted one.

\[
MD = Max([f(i,j) - f'(i,j)])
\]

The Maximum error for gray level image is up to 255; while the original image pixel value at \((I,j)\) is 255 and the decrypted image pixel value at \((i,j)\) is equal to 0. The maximum difference between both images the original and the decrypted is supposed to be minimum, for the sake of preserving the quality of the image.

5. **Normalized Cross-correlation (NC):** It is a measure of how the decrypted image pixel value at \((I,j)\) is connected with the original image pixel value at \((I,j)\) when there is no distortion in the decrypted image, then Normalized Cross-correlation will be equivalent to 1.
6. Mean Absolute Error (MAE): It is a number utilized for measuring how close predictions or forecasts are to the genuine outcomes.

\[
NC = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} |f(i, j) - f'(i, j)|^2}{\sum_{i=1}^{M} \sum_{j=1}^{N} f(i, j)^2}
\]

7. Normalized Absolute Error (NAE): It is the sum of differences among both the original image \(f(i, j)\) and the decrypted image \(f'(i, j)\) divided by the sum of the pixel square value for the original image.

\[
NAE = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} |f(i, j) - f'(i, j)|^2}{\sum_{i=1}^{M} \sum_{j=1}^{N} f(i, j)^2}
\]

8. Structural Content (SC): It is the sum of pixels' square values for the original image divided by the pixelsquare values for the decrypted image, it is also a measure for the correlation between the two images, the SC will be equivalent to 1 if there is no distortion in the decrypted images.

\[
SC = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} f(i, j)^2}{\sum_{i=1}^{M} \sum_{j=1}^{N} f'(i, j)^2}
\]

9. Signal-To-Noise Ratio (SNR): It is a measurement utilized in science and engineering that compares the level of the desired signal to the level of background noise. It is defined as the ratio of signal power to the noise power, often expressed in decibels. A ratio higher than 1 (greater than 0 dB) indicates there is more signal than noise. While SNR is commonly quoted for electrical signals, it can be applied to any form of signal.

\[
SNR = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} f'(i, j)^2}{\sqrt{\sum_{i=1}^{M} \sum_{j=1}^{N} \left(f(i, j) - f'(i, j)\right)^2}}
\]

10. Similarity Measure (SIM): It is utilized for measuring the similarity between two images.

\[
SIM = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} f(i, j) \cdot f'(i, j)}{\sum_{i=1}^{M} \sum_{j=1}^{N} f(i, j) \cdot \sqrt{f(i, j)} \cdot \sqrt{f'(i, j)}}
\]

11. Unified average changing intensity Measure (UACI): The UACI measures the average intensity of differences between the plain image and the decrypted image.

\[
UACI = \frac{1}{W \cdot H} \left[ \sum_{i=1}^{M} \sum_{j=1}^{N} \left| f(i, j) - f'(i, j) \right| \right]_{256} \times 100\%
\]

12. Delay: It is the time taken by the process of steganography or cryptography with steganography.

4. DESIGN AND IMPLEMENTATION

Secrets can be hidden in all forms of cover info. The following formula provides a very generic description of the steganographic process:

\[
cover\_medium + hidden\_data + secret\_map = stego\_medium
\]

In this context, the cover medium means that the file in which the hidden data are going to be hidden. The resultant file is that the stego medium (which can, of course, be identical kind of file as the cover medium). There are four ways that to implement steganography using (text; images; audio files or video files). For security, most effective encryption might not be enough; as a result the proposal consists of Steganography wherein encrypted data is hidden in the image after which the stego image is transmitted inside the network.
Figure 1: The Proposed Method

1. Key generation process: Read the secret key and converting it to DNA codon. There are many approaches for converting the binary data to DNA sequence and these are known as DNA coding technology. In this study the conversion of DNA sequence depends on the DNA coding rules that are shown in tables (I and II) sequentially. The key generation process explained in figure 2 with an example for each step.

Example about key generation let

**Secret Key=asd**

Binary=011000010111001101100100

DNAString=CCCTACTACCAC

Addition=CTAGCA

Subtraction=TACTCC

Combine Subtraction with first half of DNA String=TACTCCCCCTAC

Combine Addition with second half of DNA String=CTAGCATACCAC

Combine Pervious Results=CTAGCATACCACTACTCCCCCTAC

XOR Operation=GTCCACGCAGAA

Output=GTCCACGCAGAA

Length=3*4=12 Characters
The following steps in figure 3 used to convert user strings to the DNA coding scheme:

After conversion strings to DNA codon apply 2 DNA operations on this string (addition and subtraction operations). In the next step combine results of addition operation with the second part of the DNA string let it be (R) and combine the results of the subtraction operation with the first part of the DNA string let it be (W). Finally combine R and W, and finally the DNA XOR operation is implemented for the combination to produce the output key (the output key length=key length*4).

2. Encryption process: Read the plain text (secret message) and encrypting it by using the XOR operation with the generated, then check the size of the secret message with that of the cover image such that the size of the secret message should be less than the size cover image. After that the secret map will be generated depending on the length of the secret message. The secret map should be same on both sides (sender and receiver sides). The steps of encryption process are explained in figure 4 with an example.

Example about Encryption Process let

Plain Message=Computer
Character (i)=C
Cipher=Chr (Asci(C)XOR Asci(G)= _

This process ending until last message character is reached
3. Steganography process: this concept includes reading the cover image data, secret map locations and secret message. After that each character of secret message is converted to binary data, if the length of binary data less than 8 bits the padding it left by 0's. Each character of the secret message will be hiding in 3 pixels. In this work the LSB method will be used for hiding data. The steps for hiding and retrieving data are illustrated in the algorithm (1 and 2):

Algorithm1 to hide text message:-

**Input:** Cover image, Secret message, Secure lookup table  
**Output:** stego_image.

**Begin:**

Step 1: Read the cover image and secret message which is to be hidden in the cover image.

Step 2: Read secret message and write the length of secret message into the secret location shared between the sender and receiver.

Step 3: Generate hiding locations by using 2 secret maps.

Hiding locations= message length*3 e.g. if message length= 3 characters, then hiding locations=3*3=9

Secret maps: first one for x location and second one for y location.

\[ X=(i + \text{messageLength}) \mod (\text{picture.Width - 1}) + 1 \]

\[ Y=((\text{asc1(s((i \mod \text{messageLength}) + 1)) + X}) \mod (\text{picture.Height - 1}) \]

Where i from 1 to messageLength*3, s(i) is an array for message characters and asc1 extract ascii code value for the message characters depending on secure lock up table.

Step 4: For each character in the secret message do

Step 5: Convert the character to binary form.

Step 6: For each character read three locations from secret map, these locations include the pixel position in the cover image.

Step 7: Calculate LSB of each pixel in secret map of cover image.

Step 8: Replace LSB of the cover image with each bit of secret message one by one.

Step 9: Write stego image.

**End.**

Example for hiding text=Computer and key=asd

First step generates key as shown in figure 2 and after that encrypt message as shown in figure4.

1- The encrypted message=secret message=_(9365&5

2- Key Length =3

Write Message Length in pixel (3,3) in Red value

3- Select locations for hiding secret message hiding locations=MessageLength*3=8*3=24 pixels

The first location for this example X=9 and Y=106

Second location X=10 and Y=125

This process continued until generate 24 locations

4- Convert characters of the Secret Message to binary

5- Hide each character in 3 pixels in LSB of RGB values as shown in example1

6- Write the result to the Stego_image

7- This process continued until all secret message characters are hidden

Algorithm2 to retrieve text message:-

**Input:** Stego_image image, Secret Key, Secure lookup table

**Output:** Plain message.

**Begin:**

Step 1: Read the stego image.

Step 2: Extract the length of message from the shared secret location.

Step 3: Generate hiding locations by using 2 secret maps.

Hiding locations= message length*3 e.g if message length= 3 characters, then hiding locations=3*3=9

Secret maps: first one for x location and second one for y location.

\[ X=(i + \text{messageLength}) \mod (\text{picture.Width - 1}) + 1 \]
Y = ((asc1(s((i Mod messageLength) + 1)) + X) Mod (picture.Height - 1))

Where i from 1 to messageLength*3, s(i) is an array for message characters and asc1 extract ascii code value for the message characters depending on secure lock up table.

Step 4: From each location in the secret map calculates LSB of each pixel of stego image.

Step 5: For getting one byte repeat step 4 for 8 locations.

Step 6: Retrieve bits and convert each 8 bits into a character.

Step 7: Repeat step 4 to 6 until achieving maximum length of the message.

End.

The main steps for hiding and extracting data are summarized in the following steps:

Firstly an image is read from the computer, generate 3 locations for each character and then convert the character and RGB values of the extracted location to binary form. After that, the message characters are embedded using the LSB method for each character 3 pixels were needed. Next, writing the results of substitution into the stego_image, then hide the message length into image cover in pixel (key length, key length) in Red Value. Finally the extraction process, this process includes extracting the length of the secret message and secret message which was embedded during the embedding process in the first step. At first declare message bytes by reading a pixel location from the secret map starting from the first location of the secret map (for extracting one byte we need to read 3 locations from secret map). Extract the LSB bits and put it in k, when k = 8, a byte is extracted. Repeat for extracting next byte.

In order to demonstrate the online transmission of hidden data by using the proposed system. At the sender side, it is required to provide original image, creating of secret map, secret key, DNA table coding and secret message to be hiding in the stego_image. At the receiver side, it is required to provide stego_image, creating secret map, secret key, DNA table coding, retrieving the encrypted message and finally decrypting the retrieved message to get the plain text.

The analysis and results of this work illustrated in five measures:
1- Time space
2- Complexity
3- Key space
4- Brute force attack, and
5- Statistical tests: these tests include several parameters like MSE, PSNR, AD, MDR, MDG, MDB, NC, MAE, NAE, SC, SNR, UACI and SIM. The values for these parameters are shown in table 5.

5.1 Time space

This measurement involved the consumed time for encryption, decryption, hiding and retrieving messages. The consumed time was shown in table 4.
Table 4: Time consuming for encryption hiding data and decryption retrieving data

<table>
<thead>
<tr>
<th>File name</th>
<th>File type</th>
<th>Length of secret key</th>
<th>Length of encrypted message</th>
<th>E+S time (ms)</th>
<th>D+S time (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>computer</td>
<td>jpg</td>
<td>12 k</td>
<td>7 k</td>
<td>0.40</td>
<td>0.62</td>
</tr>
<tr>
<td>download</td>
<td>jpg</td>
<td>20 k</td>
<td>10 k</td>
<td>0.35</td>
<td>0.51</td>
</tr>
<tr>
<td>download</td>
<td>bmp</td>
<td>100 k</td>
<td>50 k</td>
<td>0.38</td>
<td>0.48</td>
</tr>
<tr>
<td>images/I</td>
<td>jpg</td>
<td>5 k</td>
<td>2 k</td>
<td>0.41</td>
<td>0.63</td>
</tr>
<tr>
<td>images/I</td>
<td>jpg</td>
<td>100 k</td>
<td>50 k</td>
<td>0.33</td>
<td>0.50</td>
</tr>
<tr>
<td>Images/I</td>
<td>jpg</td>
<td>5 k</td>
<td>2 k</td>
<td>0.30</td>
<td>0.46</td>
</tr>
<tr>
<td>Images/I</td>
<td>jpg</td>
<td>100 k</td>
<td>50 k</td>
<td>0.33</td>
<td>0.50</td>
</tr>
<tr>
<td>lena_orig</td>
<td>jpg</td>
<td>5 k</td>
<td>2 k</td>
<td>0.30</td>
<td>0.46</td>
</tr>
<tr>
<td>lena_orig</td>
<td>jpg</td>
<td>100 k</td>
<td>50 k</td>
<td>0.33</td>
<td>0.50</td>
</tr>
<tr>
<td>Peppers</td>
<td>jpg</td>
<td>5 k</td>
<td>2 k</td>
<td>0.30</td>
<td>0.46</td>
</tr>
</tbody>
</table>

E+S time = consumed time for Encryption and Steganography process
D+S time = consumed time for Decryption and Retrieval message process

The time was measured in milliseconds, it is noted that the proposed algorithm consumed too small amount of time for encryption, hiding data, decryption, and retrieving messages. The time taken does not exceed a second.

5.2 Complexity

For the proposed method to be broken the attacker needs to know the utilized algorithm, secret key, length of key, length of encrypted message, secret maps for generating hiding locations, location of hiding message length, which color value the length of message was embedded, key generation algorithm and finally there is important thing needs to know the secure lock up table which is shared between the sender and receiver, these seven objects are increased the complexity of the proposed method. So, if the attacker gains one of these requirements, then it was remaining needs another eight things to gain the plain message. The strength of the proposed method depended on this nine points.

5.3 Key space

In this work the key space is varying from user to another. The key length was not fixed and it was depended on utilized secret key of the receiver, but the length of generating key equal to the three time utilized key length.

5.4 Brute force attack

In cryptography, a brute force attack consists of an attacker trying many keys generated from password with a hope of guessing the correct key. The attacker checks all possible keys until the correct one is guessed. The following reports show that the generate key using the proposed method cannot be broken by using brute force attack.
5.5 Statistical tests

The test images differ in their sizes, and hidden data also differ in their sizes. The hidden data composed Arabic characters, English (capital and small letters), symbols and numerical data. From table 5 it is noted the proposed method produced a good result, there is slightly different even when hiding large amounts of data; the similarity between the original image and stego_image a proximity more than 99 %, it is clear in table 5 while for ciphering image the SIM measurement should be decreased a proximity equal to zero. The comparison between original images and stego_images was presented in figure 5, and this is shown that there are universal changes (nobody can see the hidden data) between the original and stego_images. The UACI measures the average intensity of differences between the plain image and stego_image or ciphered image, the result of UACI for ciphered image should be between 32 and 33 but for stego_image should be nearing to zeroes and this is presented in table 5; it was very near for zeroes. PSNR for steganography should be increased, the best results when the PSNR value is high. While the PSNR for ciphered image should be decreased. Also, another measurement the MSE, for steganography this
measurement should be decreased and this was shown in Table 5; but the MSE for ciphered image should be increased.

Table 5: Result of Statistical Tests for the Proposed Method

<table>
<thead>
<tr>
<th>File name</th>
<th>File size</th>
<th>Secret Type</th>
<th>Length of msg</th>
<th>MSE</th>
<th>PSNR AD</th>
<th>dB</th>
<th>ADr</th>
<th>MDs</th>
<th>LACI</th>
<th>SNR</th>
<th>SC</th>
<th>MAE</th>
<th>NAE</th>
<th>SC</th>
<th>E-SC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer</td>
<td>8 k jpeg</td>
<td>jpg</td>
<td>1</td>
<td>0.008</td>
<td>99.999</td>
<td>0.08</td>
<td>99.8</td>
<td>99.8</td>
<td>99.8</td>
<td>99.8</td>
<td>99.9</td>
<td>99.9</td>
<td>99.9</td>
<td>99.9</td>
<td>99.99</td>
</tr>
<tr>
<td>Download</td>
<td>12 k bmp</td>
<td>jpg</td>
<td>1</td>
<td>0.004</td>
<td>95.777</td>
<td>0.004</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
</tr>
<tr>
<td>Download</td>
<td>14 k jpg</td>
<td>jpeg</td>
<td>1</td>
<td>0.004</td>
<td>95.777</td>
<td>0.004</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
</tr>
<tr>
<td>Download</td>
<td>14 k jpg</td>
<td>jpeg</td>
<td>1</td>
<td>0.004</td>
<td>95.777</td>
<td>0.004</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
</tr>
<tr>
<td>Download images()</td>
<td>7 k jpg</td>
<td>jpeg</td>
<td>1</td>
<td>0.005</td>
<td>92.8</td>
<td>0.005</td>
<td>99.1</td>
<td>99.1</td>
<td>99.1</td>
<td>99.1</td>
<td>99.1</td>
<td>99.1</td>
<td>99.1</td>
<td>99.1</td>
<td>99.1</td>
</tr>
<tr>
<td>Download images()</td>
<td>7 k jpg</td>
<td>jpeg</td>
<td>1</td>
<td>0.003</td>
<td>99.19</td>
<td>0.002</td>
<td>99.1</td>
<td>99.1</td>
<td>99.1</td>
<td>99.1</td>
<td>99.1</td>
<td>99.1</td>
<td>99.1</td>
<td>99.1</td>
<td>99.1</td>
</tr>
<tr>
<td>Images images()</td>
<td>14 k jpg</td>
<td>jpeg</td>
<td>1</td>
<td>0.003</td>
<td>96.78</td>
<td>0.003</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
</tr>
<tr>
<td>Images images()</td>
<td>14 k jpg</td>
<td>jpeg</td>
<td>1</td>
<td>0.003</td>
<td>96.78</td>
<td>0.003</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
</tr>
<tr>
<td>Images images()</td>
<td>23 k jpg</td>
<td>jpeg</td>
<td>1</td>
<td>0.003</td>
<td>97.8</td>
<td>0.003</td>
<td>99.8</td>
<td>99.8</td>
<td>99.8</td>
<td>99.8</td>
<td>99.8</td>
<td>99.8</td>
<td>99.8</td>
<td>99.8</td>
<td>99.8</td>
</tr>
<tr>
<td>Images images()</td>
<td>137 k jpg</td>
<td>jpeg</td>
<td>1</td>
<td>0.003</td>
<td>96.78</td>
<td>0.003</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
</tr>
<tr>
<td>Images images()</td>
<td>21 k jpg</td>
<td>jpeg</td>
<td>1</td>
<td>0.004</td>
<td>94.14</td>
<td>0.004</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
</tr>
<tr>
<td>Images images()</td>
<td>79 k jpg</td>
<td>jpeg</td>
<td>1</td>
<td>0.005</td>
<td>94.1</td>
<td>0.005</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
<td>99.4</td>
</tr>
<tr>
<td>Images images()</td>
<td>115 k jpg</td>
<td>jpeg</td>
<td>1</td>
<td>0.003</td>
<td>96.78</td>
<td>0.003</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
<td>99.7</td>
</tr>
<tr>
<td>Images images()</td>
<td>9 k jpg</td>
<td>jpeg</td>
<td>1</td>
<td>0.003</td>
<td>97.8</td>
<td>0.003</td>
<td>99.8</td>
<td>99.8</td>
<td>99.8</td>
<td>99.8</td>
<td>99.8</td>
<td>99.8</td>
<td>99.8</td>
<td>99.8</td>
<td>99.8</td>
</tr>
<tr>
<td>Original Image</td>
<td>Stego. Image</td>
<td>Hiding Message</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------------</td>
<td>-------------</td>
<td>----------------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><img src="image1.png" alt="Original Image" /></td>
<td><img src="image2.png" alt="Stego. Image" /></td>
<td><strong>High level encryption algorithms and their applications on a real-world scenario.</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><img src="image3.png" alt="Original Image" /></td>
<td><img src="image4.png" alt="Stego. Image" /></td>
<td><strong>International Data Encryption Algorithm (IDEA)</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><img src="image5.png" alt="Original Image" /></td>
<td><img src="image6.png" alt="Stego. Image" /></td>
<td><strong>Hashing techniques are an important aspect of any block symmetric systems.</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><img src="image7.png" alt="Original Image" /></td>
<td><img src="image8.png" alt="Stego. Image" /></td>
<td><strong>Proposal for A New Robust Image Steganography Algorithm Using Watermarking Operations.</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><img src="image9.png" alt="Original Image" /></td>
<td><img src="image10.png" alt="Stego. Image" /></td>
<td><strong>A Block Cipher Based on IDEA and XOR.</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><img src="image11.png" alt="Original Image" /></td>
<td><img src="image12.png" alt="Stego. Image" /></td>
<td><strong>A collection of IDEA variants.</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><img src="image13.png" alt="Original Image" /></td>
<td><img src="image14.png" alt="Stego. Image" /></td>
<td><strong>Differential Cryptanalysis.</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><img src="image15.png" alt="Original Image" /></td>
<td><img src="image16.png" alt="Stego. Image" /></td>
<td><strong>DNA ADDITION AND SUBTRACTION OPERATIONS.</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><img src="image17.png" alt="Original Image" /></td>
<td><img src="image18.png" alt="Stego. Image" /></td>
<td><strong>DNA ADDITION AND SUBTRACTION OPERATIONS.</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><img src="image19.png" alt="Original Image" /></td>
<td><img src="image20.png" alt="Stego. Image" /></td>
<td><strong>DNA ADDITION AND SUBTRACTION OPERATIONS.</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><img src="image21.png" alt="Original Image" /></td>
<td><img src="image22.png" alt="Stego. Image" /></td>
<td><strong>DNA ADDITION AND SUBTRACTION OPERATIONS.</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><img src="image23.png" alt="Original Image" /></td>
<td><img src="image24.png" alt="Stego. Image" /></td>
<td><strong>Message Transmission Based on DNA Cryptography.</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><img src="image25.png" alt="Original Image" /></td>
<td><img src="image26.png" alt="Stego. Image" /></td>
<td><strong>12/12/2014 - 12:12:12.</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><img src="image27.png" alt="Original Image" /></td>
<td><img src="image28.png" alt="Stego. Image" /></td>
<td><strong>The operations used for DNA operations are different and are based on the specific DNA algorithms.</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><img src="image29.png" alt="Original Image" /></td>
<td><img src="image30.png" alt="Stego. Image" /></td>
<td><strong>A Brief History of Information Hiding.</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
6. CONCLUSION

This paper proposes to design an effective scheme for efficient Secure Image Steganography design using DNA sequence based on DNA Cryptography, LSB technique and secret map for creation hiding locations. LSB primarily based steganography imbed information within the least significant bits of digital pictures. The LSB insertion may be a common and straightforward approach for embedding info into cover file. DNA sequencing is any method used to design the sequence of nucleotides that comprise a strand of DNA. The data was encrypted with the help of DNA sequence (for creating keys), XOR operation for encrypting data and secure asci code lookup table. After that the locations for hiding secret data was generated by using two secret maps. The secret data was hidden by cover image with the help of Image Steganography. This paper provides effective steganography technique, so that the person can detect the variation of choosing the method to protect the information. In Image Domain, we tend to mentioned the foremost powerful technique referred to as LSB to hide information specially within images in any format (BMP, JPG, PNG and .. etc). The proposed method takes few milliseconds for encryption and steganography processes. Also, the statistical tests show the proposed method achieves good results. The results of similarity measure proximity 99.8 %, so the hiding data cannot be detected without knowing (the utilized algorithm, secret key, length of key, length of encrypted message, secret maps for generating hiding locations, location of hiding message length, which color value the length of message was embedded, key generation algorithm and finally there is important thing needs to know the secure lock up table which is shared between the sender and receiver). Also, the other statistical test give a good results such as (MSE, PSNR, UACI, .. etc). Finally this paper ends with Application of a novel new steganography method.

REFERENCES


