An Enhanced Image Security Using Improved RSA Cryptography and Spatial Orientation Tree Compression Method

Vandana Rajput
Research Scholar, CSE Department
VITM, Gwalior, India
Vandanarajput2010@gmail.com

Sandeep Kumar Tiwari
Asst. Professor, CSE Department
VITM, Gwalior, India
sandeep72128@gmail.com

Rohit Gupta
Asst. Professor, CSE Department
VITM, Gwalior, India
cs_rohit@vitm.edu.in

Abstract- In this research, presented an enhanced image security using improved RSA cryptography and Spatial Orientation Tree Wavelet compression (STW) method for color or grayscale image. Steganography is the science of hiding secret messages so that you can be sent to target without mistrust. It is the most famous type of carrier to grip data. Firstly, input the secret message, then encrypt the message using an improved RSA algorithm. In this method, take four prime numbers for enhanced security. Using STW compression method, compress the bits of an image for utilizing the memory space. For color pictures an changed parent offspring connection and an extra stage of wavelet decomposition on RGB planes have been carried out. For embedding, encrypted message into compressed image for obtaining stego image. PSNR improvement of 83.541% was achieved at 0.177 bpp for RGB images and 2.223% for color images.

Keywords: RSA, LSB, STW Compression, PSNR, Entropy, CR.

I. INTRODUCTION

Internet has become a convenient way for data transmission in today’s era, therefore the information security is considered as one of the immediate matter. A form of approaches was developed to encrypt and decrypt the key knowledge to preserve the message secret. Cryptography is a technique which is used for maintaining the confidentiality of communication [1]. Compression of color images is done by extending gray level image compression algorithm using SPIHT to three planes of color image. For color picture compression, the picture is changed to luminance-chrominance color planes (YCbCr). Each color plane is coded independently and resultant bit streams are transmitted one after other [2]. In the LSB system, the LSB of the pixels are changed by means of the message bits which are permuted before embedding [3]. RSA algorithm works slowly and provides much less safety over the community. To develop the pace of computation of RSA algorithm and to expand the safety we need to modify the RSA algorithm which will also be done by way of third prime quantity and offline storage procedure [4].

The overall work is summarized as follows: section 2 gives literature survey, section 3 gives proposed work, section 4 gives simulation results and finally section 5 concludes work.

II. LITERATURE SURVEY

Message is first encrypted using Vernam cipher algorithm and then the message (encrypted) is embedded inside an image using the new image steganography method i.e. LSB with Shifting (LSB-S) [5]. Embed EPR in the cover image by discrete cosine transform watermarking and RSA algorithm of cryptography [6]. Image encryption is done using Blowfish Algorithm as it is faster and has good performance and for image hiding LSB technique is used as it is faster, simple and Bit manipulated [7]. Analyze the available picture based steganography together with cryptography method utilizing LSB method to achieve protection [8]. Development of LSB approach for hiding understanding in pictures
uses the concept of fame checking for insertion and retrieval of message [9].

III. PROPOSED WORK

In this algorithm, proposed work, grayscale image or color picture is taken as the input image. After that, RSA is performed for decryption and encryption with the private and public key. In this process, take four prime numbers for enhances security issues. Using STW compression technique, form the roots of the modified tree and LIP and LIS are initialized with these nodes. These nodes have three offspring in the same band as specified by the modified parent children relationship. Hide the message in picture to make stego image using LSB because later than embedding the size of cover media increases, which may direct to cause suspicions.

Encryption Algorithm

Input: Text Message, Random Numbers
Output: Encrypted Message

1. Enter the text message (M) input for encryption.
2. Select A, B, C and D are random numbers, \( p \neq q, a \neq b \).
3. Determine the value of modulus of random numbers:
   \[ n = A \times B \times C \times D \] (1)
4. Find the Euler totient function using formula:
   \[ \phi(n) = (A - 1) \times (B - 1) \times (C - 1) \times (D - 1) \] (2)
5. Select integer \( e \) whose \( gcd = (\phi(n), e = 1) \); \( 1 < e \)
6. Find the modular multiplicative inverse and it is represented as a d
   \[ d = e^{-1} \left( mod(\phi(n)) \right) \] (3)
7. Public key PU = \{e, n\}.
8. Private Key PR = \{d, n\}.
9. Encrypt message using this equation:
   \[ Cipher = M^e \ mod(n) \] (4)

STW Algorithm

In STW, this algorithm is identical for encoder and decoder. Each encoder and decoder consist three lists and update always. These lists are: i) list of insignificant pixels (LIP) ii) List of significant pixels (LSP) iii) List of insignificant sets (LIS)

Set Partitioning Rules

Some rules in spatial orientation tree:

ON (n1, n2): In this set, contain coordinates of all pixels i.e. offspring of the node (n1, n2). It namely as offspring set.

DN (n1, n2): In this set, contain coordinates of all pixels, i.e. descendant of the node (n1, n2). It namely as descendant’s set.

L (n1, n2): This set is used for determining the difference between ON and DN.

H: This on this set, maintain every spatial orientation tree root’s coordinates which belongs to the LL sub-band.

Rule 1: Initially the Partition consist DN (n1, n2) for all \( (n1, n2) \in H \)

Rule 2: If DN (n1, n2) searched significant then split into L (n1, n2) with four single element set \((i, j) \in ON(n1, n2)\)

Rule 3: If L (n1, n2) is search significant then break it into four sets of DN (my, j) where \((my, j) \in ON(n1, n2)\)

A. Initial Stage: For output n, LSP is empty; afterward add the commencing root coordinates to LIP or LIS.

B. Sorting Pass: Each entry in LIP, it will decide that it is significant and give the output decision result. If the decision result is significant then moved into the LSP and the coordinate present with their sign. If the rest coordinates all are going to be significant then this process is stopped.

C. Refinement Pass: In this pass all the pixel values in LSPare now \( 2n \leq |Ci,j| \), and output will be the nth significant bit.

D. Quantization-step replace: in this step, decrement n by 1 and go to step 2.

Hiding Algorithm

Input: Compressed Image and Secret Message
Output: Stego Image

10. Select a color image or grayscale image as a cover image and resize the image with 512*512 size.
12. Calculate LSB of every pixel of compressed image.
13. Perform LSB on the compressed image to embed text into compressed image.
   
   \[ LSB = \{(I(i,j),2)\} \]  
   
   Where I is compressed image and i, j is size of the image
14. Set back LSB of the compressed image with every bit of hidden message individually.

**Extraction Algorithm**

**Input:** Stego Image

**Output:** Plaintext Message

15. For the message decryption, Bob uses Bob’s personal key and resolves the simple text, M as:
   
   \[ M = C^d \mod(n) \]  
   
   Where d is decrypt, C is cipher text, M is plain text
16. Decrypt text from the stego image using RSA decryption algorithm.

---

**IV. SIMULATION RESULTS**

This algorithm is implemented and executed using MATLAB R12. The PSNR of Stego images are calculated in figure shown below. The entropy stands for the collective random error between cover and Stego images shown in table2. Lower entropy indicates lower error between Stego and cover picture. PSNR is generally utilized in steganography to calculate the PSNR in cover image and Stego image after embedding the concealed information shown in table1. Entropy and PSNR are the generally used metrics for calculating the superiority of Stego image.

**Fig1. Proposed Flow Chart**

**Fig2. Image Dataset with Color and Grayscale Images**

**Fig3. Image Dataset with Color and Grayscale Images**

**Fig4. Proposed Result on Image ‘gantrycrane’**

The proposed scheme is functional on many cover images with unusual values of keys and plain text.
The consequential cover and Stego image are displayed in Fig4 above.

The tentative outcomes convey that this technique provides sufficiently good PSNR value and low entropy value. Along with this a huge amount of data may be accumulated due to enhanced image of cover image. For testing purpose following eight images were taken.

1. Calculate PSNR and MSE value of stego and cover image.
   
   \[
   MSE(x) = \frac{1}{N} \sum_{i=1}^{N} (x - x^\wedge)^2
   \]

   Where \( x \) is cover image, \( x^\wedge \) is stego image, \( N \) is the size of cover image.

   \[
   PSNR(x) = \frac{20 \times \log_{10}(\text{double}(m) \times 2))}{MSE(x)}
   \]

   Where \( m \) is the maximum value of the cover image.

2. Calculate entropy of an image:
   
   \[
   E = -\sum_{i=1}^{N} p(x_i) \log p(x_i)
   \]

   Where \( E \): Entropy

   \( N \): highest value of gray level

   \( p(x_i) \): prospect of rate of \( x_i \)

3. Calculate compression ratio (CR) of an image:

   \[
   CR = \frac{\text{no_of_coded_bits}}{N}
   \]

Table 1. PSNR Comparison between Base and Proposed System

<table>
<thead>
<tr>
<th>Image</th>
<th>Base PSNR</th>
<th>Proposed PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fabric</td>
<td>55.927</td>
<td>77.397</td>
</tr>
<tr>
<td>football</td>
<td>56.418</td>
<td>83.981</td>
</tr>
<tr>
<td>gantrycrane</td>
<td>59.487</td>
<td>81.461</td>
</tr>
<tr>
<td>greens</td>
<td>55.819</td>
<td>82.061</td>
</tr>
<tr>
<td>Mountain</td>
<td>50.058</td>
<td>77.701</td>
</tr>
<tr>
<td>Photo101</td>
<td>59.477</td>
<td>82.298</td>
</tr>
<tr>
<td>Turkevich</td>
<td>59.725</td>
<td>83.541</td>
</tr>
<tr>
<td>wom2</td>
<td>53.520</td>
<td>80.046</td>
</tr>
</tbody>
</table>

Table 2. Proposed Result on CR, BPP and Entropy

<table>
<thead>
<tr>
<th>Image</th>
<th>CR</th>
<th>BPP</th>
<th>Entropy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fabric</td>
<td>9.532</td>
<td>2.287</td>
<td>0.749</td>
</tr>
<tr>
<td>football</td>
<td>5.735</td>
<td>1.376</td>
<td>0.4586</td>
</tr>
<tr>
<td>gantrycrane</td>
<td>4.522</td>
<td>1.085</td>
<td>0.752</td>
</tr>
<tr>
<td>greens</td>
<td>21.657</td>
<td>5.197</td>
<td>0.541</td>
</tr>
<tr>
<td>Mountain</td>
<td>9.061</td>
<td>0.724</td>
<td>0.999</td>
</tr>
<tr>
<td>Photo101</td>
<td>3.078</td>
<td>0.738</td>
<td>0.966</td>
</tr>
<tr>
<td>Turkevich</td>
<td>2.223</td>
<td>0.177</td>
<td>0.997</td>
</tr>
<tr>
<td>wom2</td>
<td>26.459</td>
<td>2.116</td>
<td>0.970</td>
</tr>
</tbody>
</table>

Fig5. PSNR comparison between Base and Proposed system.
In this figure, blue line shows that the base system [1] and red line shows that the proposed system result. You can

V. CONCLUSION

It can be concluded that when normal image security using steganographic and enhancement technique is applied, it creates the job of the researchers impracticable to decrypt the encoded secret message. It is tough to decipher the key and to decrypt the ciphered text with knowing the proper key. It gives efficiency for secret message and the usability of key makes the system simple and yet confident. A reordering in spatial orientation tree of SPIHT algorithm ensures that it codes more significant information in the initial bits of the encoded bit stream leading to raised performance at very low bit premiums. Thus it can be used in small mobile devices having limited memory and processing capability when they are compressing an image for transmission over any network. For color pictures a different DWT decomposition stage and an altered composite STW outcome in an increased PSNR performance at curb bit charges. This algorithm enhances the security using improved RSA method. Further we can extend this work to use this technique with 3D images for making the allocation that have half secret and disclose that secret by stacking to everyone. We will be focussed to worked with Doc file, Text File and secure those files of data using Image Steganography technique.

REFERENCES


