Efficient and Robust Reversible Image Data Hiding using RSA Encryption

Pallavi Dixit[1], Prof. Bhupendra Sen[2]

[1]M. Tech. Scholar, [2]Guide

Department of Electronics and Communication Engineering, BIST, Bhopal

Abstract - Image data hiding the domain deal with the information security transmit via images. The images are being processed in such a way that the information is being hidden behind image without compromising with the quality of image, if image lose its visibility then anyone can predict there some information hidden with the image or some processing has been done with images. For increasing the security of the information lots of algorithms are there to encode the data of information being hidden. Some of the encryption techniques need private keys to encrypt as well decrypt the extracted information with image. This work focused to implement the algorithm which is being reversible in nature with optimum security and accuracy and has the capability to keep as much data as needed with the domain limitations. The proposed methodology uses RSA encryption algorithm to secure the secret information/data and LSB to make it 100% reversible. The robustness is evaluated with the capacity in bits and accuracy in percentage.

Keywords - Private Keys, RSA Encryption, Reversible Data Hiding, Image Processing, Information Security.

I. INTRODUCTION

Cryptography, defined as the science and study of secret writing concerns the ways in which communications and data can be encoded to prevent disclosure of their contents through eavesdropping or message interception, using codes, ciphers and other methods, so that only certain people can see the real message.

Security often requires that data be kept safe from unauthorized access. And the best line of defence is physical security (placing the machine to be protected behind physical walls). However, physical security is not always an option, due to cost and/or efficiency considerations. Instead, most computers are interconnected with each other openly, thereby exposing them and the communication channels that they use. With regards to confidentiality, cryptography is used to encrypt data residing on storage devices or travelling through communication channels to ensure that any illegal access is not successful. Also, cryptography is used to secure the process of authenticating different parties attempting any function on the system. Since a party wishing be granted a certain functionality on the system must present something that proves that they indeed who they say they are. That something is sometimes known as credentials and additional measures must be taken to ensure that these credentials are only used by their rightful owner. The most classic and obvious credential are passwords. Passwords are encrypted to protect against illegal usage.

Authorization is a layer built on top of authentication in the sense that the party is authenticated by presenting the credentials required (passwords, smart cards ... etc.). After the credentials are accepted the authorization process is started to ensure that the requesting party has the permissions to perform the functions needed.

Data integrity and Non-Repudiation are achieved by means of digital signature, a method that includes performing cryptography among other things.

With a public key (PKA) or asymmetric key algorithm, a pair of keys is used. One of the keys, the private key, is kept secret and not shared with anyone. The other key, the public key, is not secret and can be shared with anyone. When data is encrypted by one of the keys, it can only be decrypted and recovered by using the other key. The two keys are mathematically related, but it is virtually impossible to derive the private key from the public key. The RSA algorithm is an example of a public key algorithm used in proposed work. Figure 1.1 shows the RSA public encryption structure.

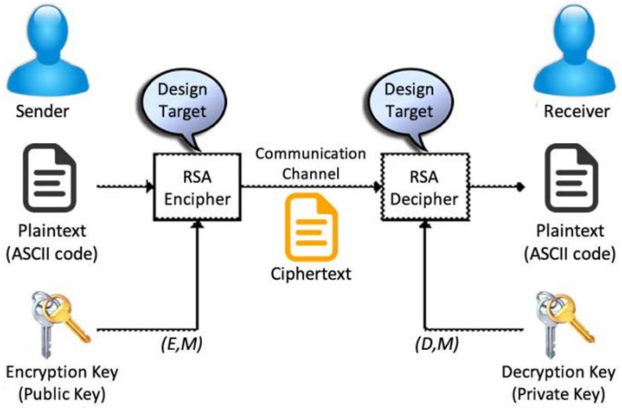

Figure 1.1 RSA public key encryption algorithm.

II. RSA ENCRYPTION ALGORITHM

The rising growth of data communication technique and electronic transactions over the web has made system security to become the most important issue over the network. To provide modern security features, public-private key cryptosystems are used. One of such cryptosystem is RSA algorithm. Though computation in RSA takes more time by if the message to be encrypted is generated randomly then RSA will prove to be good cryptography algorithm for system security [7].

For the better working of RSA based cryptosystem the system has the public key for decryption and the user will have the device containing the private key assigned to the user. And instead of entering the password the user will just have to insert the device to the system and the system will do the cross checking of the password for that particular user and allow access accordingly [8].

1. Public-key encryption

In RSA, encryption keys are made public while the decryption keys are kept private, so only the person with the correct decryption key can decipher an encrypted message. Everyone has their own encryption and corresponding decryption keys. The keys are made in such a way that the decryption key cannot be easily deduced from the public encryption key.

2. Digital signatures

The receiver may need to verify that a transmitted message is actually originated from the sender, and didn't just come from authentication. This is done with the help of the sender's decryption key, and later the signature can be verified by anyone, using the corresponding public encryption key. Signatures therefore cannot be copied. Also, no signer can later deny having signed the message.

The various steps involved in RSA algorithm are :

A. Finding large prime numbers

Finding ‘n’ is the first step of the algorithm, where ‘n’ is the product of two prime numbers ‘p’ & ‘q’. The number ‘n’ will be revealed in the encryption and decryption keys, but the numbers ‘p’ and ‘q’ will not be explicitly shown. The prime numbers ‘p’ and ‘q’ should be large such that it will be very difficult to derive from ‘n’.

$$n = p \times q \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots Eq. 1$$

B. Finding the public key (e)

Choose a number ‘e’ such that ‘e’ is co-prime to $\varphi(n)$, where $\varphi(n)$ is the Euler’s totient function that counts the number of positive integers less than or equal to ‘n’ that are relatively prime to ‘n’ i.e.

$$\varphi(n) = (p-1)(q-1) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots Eq. 2$$

$$Gcd\big(e, \varphi(n)\big) = 1 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots Eq. 3$$

where, $1 < e < \varphi(n)$

C. Determine the private key (d)

Determine the private key ‘d such that‘d’ is the multiplicative inverse of the public

key ‘e’ i.e.

$$d^{-1} = e\Big(mod\big(\varphi(n)\big)\Big) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots Eq. 4$$

C. Encryption

Let ‘m’ be the message (integer type) that is to be encrypted using public key ‘e’ to give the encrypted message as ‘c’ where ‘c’ is calculated as

$$c = m^{e}\big(mod(n)\big) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots Eq. 5$$

D. Decryption

The decrypted message ‘m’ is found out using the private key‘d’ and is calculated

as:

$$m = c^{d}\big(mod(n)\big) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots Eq. 6$$

III. PROPOSED METHODOLOGY

The implementation of proposed is based on Least Significant Bit (LSB) as one of the steganography techniques along with RSA encryption algorithm to enhance the security level of image stenography. The least significant bit (in other words, the 8th bit) of some or all of the bytes inside an image is changed to a bit of the secret message. The implementation and simulation of proposed algorithm is completed in Matlab. Figure 3.1 shows the taxonomy of Reversible Image Data Hiding with Encryption.

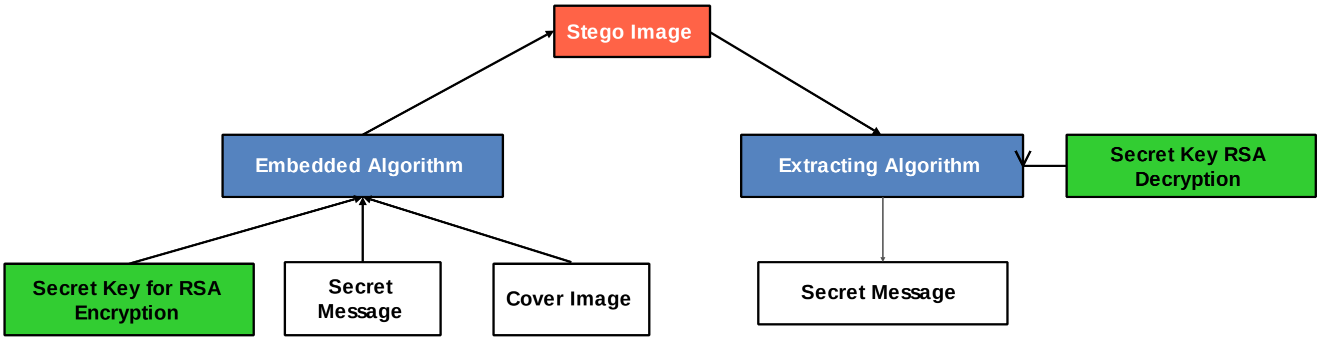

Figure 3.1 Reversible Image Data Hiding with Encryption

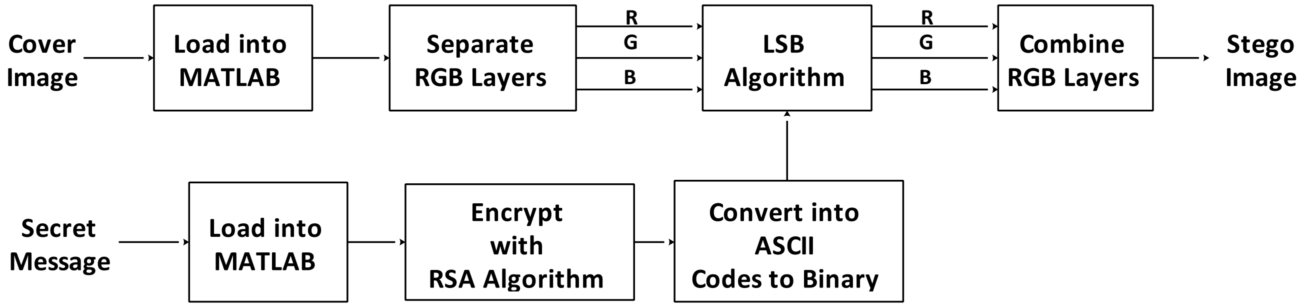

Figure 3.2 Block diagram of Embedding process.

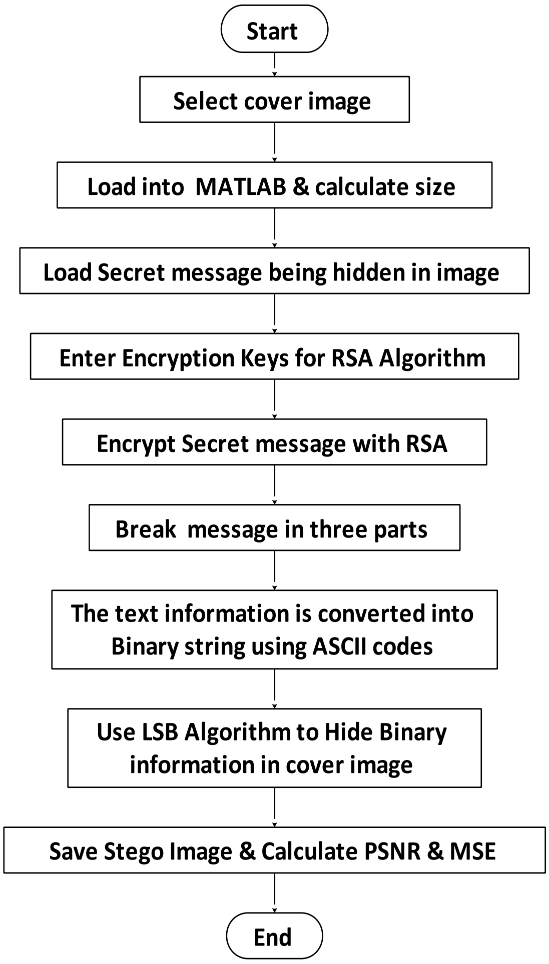
Start
Select cover image
Load into MATLAB & calculate size
Load Secret message being hidden in image
Enter Encryption Keys for RSA Algorithm
Encrypt Secret message with RSA
Break message in three parts
The text information is converted into Binary string using ASCII codes
Use LSB Algorithm to Hide Binary information in cover image
Save Stego Image & Calculate PSNR & MSE
End

Figure 3.3 Embedding process flow chart

Increasing or decreasing the value by changing the LSB does not change the appearance of the image; much so the resultant stego image looks almost same as the cover image. Figure 3.1 block diagram of water marking process has are segmented in two parts.

Two inputs are Cover image and secret message to be embedded in it and a stego image output.

First cover image is loaded into Matlab environment. Then RGB layers are separated and LSB algorithm is applied on cover image. Now secret message is loaded in it to Matlab. First the message is encrypt with RSA encryption algorithm then convert encrypted secret message in to ASCII to binary code. Now using LSB algorithm embed text with cover image. Combine layers of cover image. Finally highly encrypted stego image is obtained. The process flow of proposed work has given in figure 3.3.

Figure 3.5 demonstrated the block diagram of image retrieval process to extract the secret information from stego image.

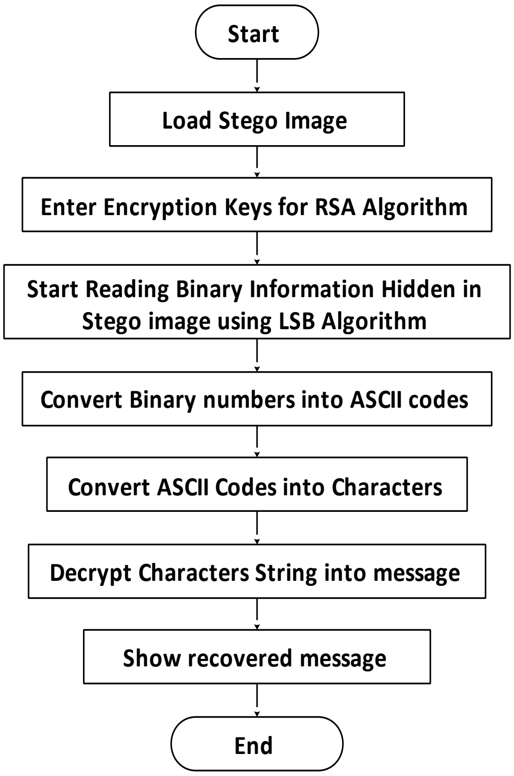
Start
Load Stego Image
Enter Encryption Keys for RSA Algorithm
Start Reading Binary Information Hidden in Stego image using LSB Algorithm
Convert Binary numbers into ASCII codes
Convert ASCII Codes into Characters
Decrypt Characters String into message
Show recovered message
End

Figure 3.4 Retrieval Process flow Chart.

To extract secret data embedded in image first stego image has to be loaded into Matlab environment. The separate layers of stego image read hidden information using LSB algorithm. Decrypt retrieved information with RSA decryption algorithm. The process flow of information retrieval has given in figure 3.4. Block diagram shows the internal mechanism of proposed algorithm where there are 4 separate functional block are used to retrieve secret data from stego image. The fundamental blocks are separate layers read hidden information using LSB. Decrypt with RSA figure 3.4 shows the process flow of proposed decryption algorithm.

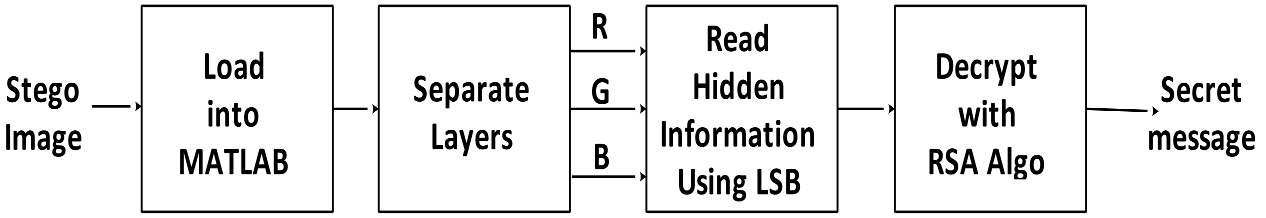

Figure 3.5 Block Diagram of Retrieval Process.

IV. EXPERIMENTAL RESULTS

Implementation and synthesis of proposed work has done on MATLAB. Some of the test images are shown below with their respective stego images.

Simulation is performed for several different images, each of which is shown in figure 4.1. The results of three test images are shown here for illustrative purposes in figure 4.2. The performances were observed to be approximately the same for both algorithms proposed algorithm with corresponding base algorithm, and on average a better performance was observed for all test images using proposed algorithm.

It is believe that for various applications, including using hashing within watermarking, large rotation and cropping attacks can be handled by proposed algorithm. Table 1 gives the comparison of performance of proposed algorithm with base algorithm in terms of capacity and accuracy.

Figure 4.1 Original Cover Images (a) Lena, (b) Baboon, (c) Airplane.

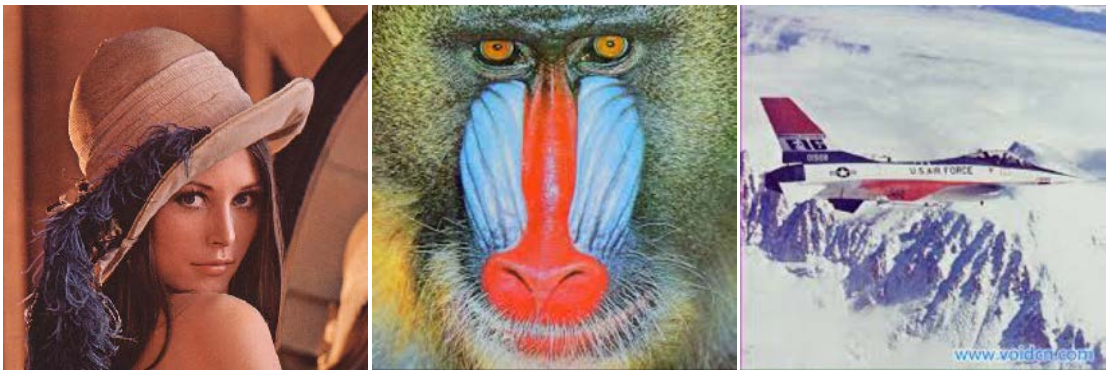

Figure 4.2 Stego Images (a) Lena, (b) Baboon, (c) Airplane.

Table: 1 Comparison of Capacity and Accuracy with Previous Methodology

Capacity		*Accuracy*	
Proposed Methodology	*Previous Methodology*	*Proposed Methodology*	*Previous Methodology*
12288 bits	12288 bits	100%	100%
14016 bits	14016 bits	100%	100%
16000 bits	15987 bits	100%	100%

V. CONCLUSION AND FUTURE SCOPES

In this investigation explored a new approach of Image data hiding using RSA encryption. This approach is gone for protection of the marginal statistics of a cover image. The security of marginal statistics helps in defeating the targeted attacks designed for intended for particular steganographic algorithms.

It is found that under a specified constraint the proposed algorithm is optimal in terms of the capacity and accuracy during restoration procedure. It was likewise observed that despite the fact that the encryption and restoration of the image can protect image and improve the security of an embedding algorithm.

In future there is a scope to analyze convergence and stability properties of the proposed algorithm and apply it to other multimedia objects, such as audio. Moreover, in future the proposed algorithm provides direction to design multimedia watermarking algorithms.

REFERENCES

[1]. J. Zhou, W. Sun, L. Dong, X. Liu, O. C. Au and Y. Y. Tang, "Secure Reversible Image Data Hiding Over Encrypted Domain via Key Modulation," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 26, no. 3, pp. 441-452, March 2016.

[2]. H. Nyeem, "Reversible data hiding with image bit-plane slicing," 2017 20th International Conference of Computer and Information Technology (ICCIT), Dhaka, 2017, pp. 1-6.

[3]. L. Tomy and Namitha T N, "Secure data transmission through reversible data hiding," 2016 Online International Conference on Green Engineering and Technologies (IC-GET), Coimbatore, 2016, pp. 1-4.

[4]. N. N. Chendulkar and P. S. Mahajani, "Reversible Data Hiding in Cloud Based Applications," 2015 International Conference on Computational Intelligence and Communication Networks (CICN), Jabalpur, 2015, pp. 1141-1146.

[5]. Y. Y. Satpute and B. A. Tidke, "Data Compression and Hiding Using Advanced SMVQ and Image Inpainting," 2015 International Conference on Computational Intelligence and Communication Networks (CICN), Jabalpur, 2015, pp. 1074-1077.

[6]. Agham and T. Pattewar, "Data hiding technique by using RGB-LSB mechanism," International Conference on Information Communication and Embedded Systems (ICICES2014), Chennai, 2014, pp. 1-5.

[7]. T. S. K. Shripriyadharshini, S. yohalakshmi and S. Deepa, "Reserve Room based Reversible Data Hiding in digital images," 2014 International Conference on Communication and Signal Processing, Melmaruvathur, 2014, pp. 1452-1456.

[8]. M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, “Lossless generalized-LSB data embedding,” IEEE Trans. Image Process., vol. 14, no. 2, pp. 253–266, Feb. 2005.

[9]. M. U. Celik, G. Sharma, and A. M. Tekalp, “Lossless watermarking for image authentication: A new framework and an implementation,” IEEE Trans. Image Process., vol. 15, no. 4, pp. 1042–1049, Apr. 2006.

[10]. Z. Ni, Y.-Q. Shi, N. Ansari, and W. Su, “Reversible data hiding,” IEEE Trans. Circuits Syst. Video Technol., vol. 16, no. 3, pp. 354–362, Mar. 2006.

[11]. X. Li, W. Zhang, X. Gui, and B. Yang, “A novel reversible data hiding scheme based on two-dimensional difference-histogram modification,” IEEE Trans. Inf. Forensics Security, vol. 8, no. 7, pp. 1091–1100, Jul. 2013.

[12]. C. Qin, C.-C. Chang, Y.-H. Huang, and L.-T. Liao, “An inpainting- assisted reversible steganographic scheme using a histogram shifting mechanism,” IEEE Trans. Circuits Syst. Video Technol., vol. 23, no. 7, pp. 1109–1118, Jul. 2013.

[13]. W.-L. Tai, C.-M. Yeh, and C.-C. Chang, “Reversible data hiding based on histogram modification of pixel differences,” IEEE Trans. Circuits Syst. Video Technol., vol. 19, no. 6, pp. 906–910, Jun. 2009.