International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Separable Reversible Data Hiding In Image Using a Keyless Approach

Jitbahan Samanta[1], G. Sujatha[2]

[1]M. Tech Student, Department of Information Technology, SRM University, Chennai, India

[2]Assistant Professor (Sr.G), Department of Information Technology, SRM University, Chennai, India

Abstract: ***Reversible data hiding is a process in which the original cover can be losslessly recovered after embedded data is extracted while protecting the image content's confidentiality. Earlier many research papers proposed reversible data hiding but all those techniques are dependent on keys. This paper presents an efficient approach to reversible data hiding where we use a keyless approach for image encryption. We follow standard RDH algorithms for hiding data in image before encryption and then encrypting the image with a keyless approach.***

Keywords: RDH, Visual Cryptography, SDS, Histogram shifting

1. Introduction

Reversible data hiding is useful in applications such as in law enforcement, medical images systems, where it is desired to be able to reverse the stegno media back to the original cover media for legal consideration. Reversible data hiding is an algorithm, which can recover the original image losslessly after the data have been extracted.

There are many standard rdh algorithms which are used for data hiding. We are using histogram shifting based algorithm here for our purpose. Here, reserving room before image encryption is followed. In the existing system, after reserving room for data embedding, a data hider hides additional data in image using a data hiding key. Then again for image encryption we encrypt the image using encryption key. So, this approach is dependent on keys. Figure 1shows the existing system.

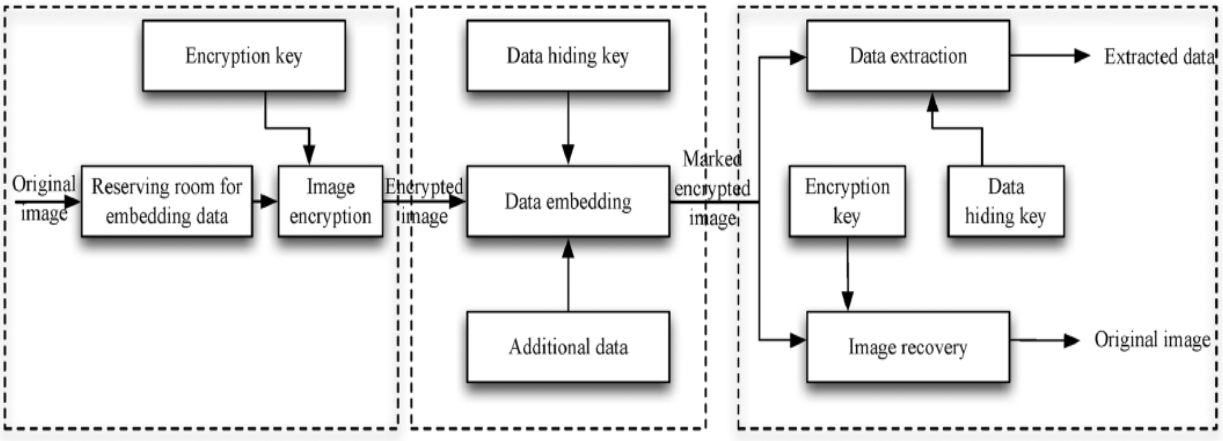

Figure 1: Existing System Framework

To avoid key management problem, in our proposed method, after applying rdh, we are using a keyless approach for image encryption using visual cryptography. We are following sds algorithm which mainly consists of three steps-sieving, division and shuffling. By this procedure, we are generating k random shares of the image. We need all those k shares to get back the original image. After recovering the original image, we can extract data back from the image. Thus we can recover original data without any loss of original cover media.

2. Proposed Method

In our proposed system are using two concepts together. We are making use of reversible data hiding and visual cryptography concept here. Figure 2 shows our proposed framework. Here, first we vacant space in image for embedding data before image encryption. Since, creating space for data after image encryption is difficult task to do and inefficient, reserving room prior to image encryption is our choice here. After vacating space using standard rdh algorithms we hide data into the image. Now the secrecy of the image is very important. There are many research papers where image is encrypted with key based encryption algorithms. In our proposed method we combine the idea of visual cryptography proposed by Naor and Shamir.

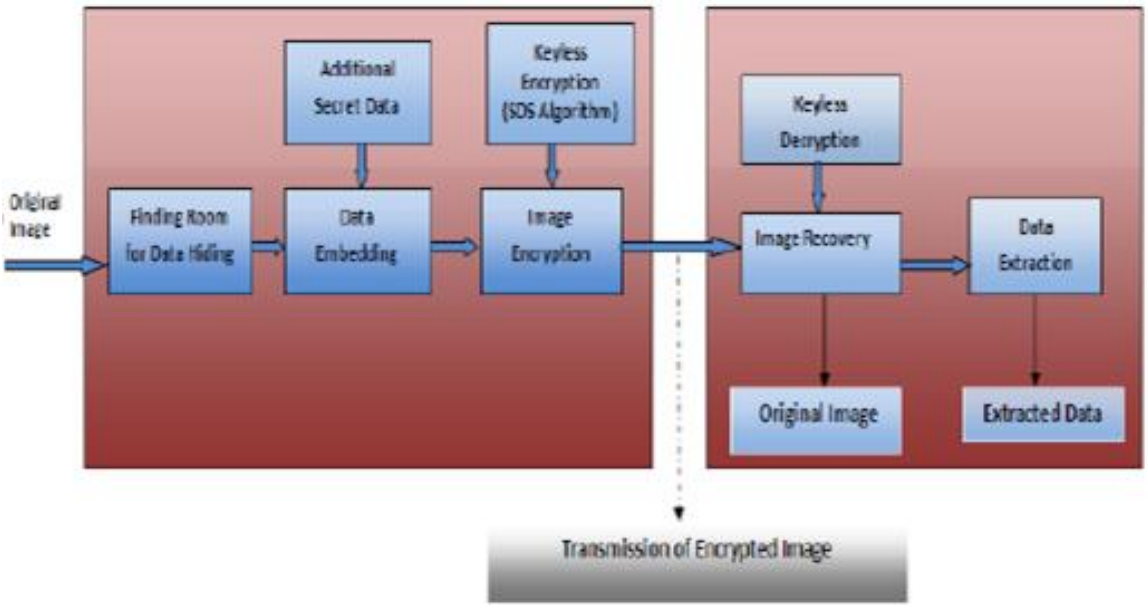

Figure 2: Our proposed framework

The proposed scheme use sieving-division-shuffling algorithm for encrypting the image in lossless fashion. The scheme is robust to withstand the brute force attack. The scheme is suitable for authentication based application or where trust cannot be reposed in any one participant for decision making and collective support is required to proceed.

A. Finding room for data embedding

The first step for embedding data is to find room in the image. In this purpose, we go for image partition. The goal of image partition is to construct a smoother area B on which standard rdh algorithms can achieve better performance. To do that without loss of generality, assume the original image C is an 8 bits gray-scale image with its size M*N pixels and

$$\mathbf{C}_{i,j} \in [0, 255], 1 \leq i \leq M, 1 \leq j \leq N.$$

First, the content owner extracts from the original image, along the rows, several overlapping blocks whose number is determined by the size of to-be-embedded messages, denoted by l. In detail, every block consists of m rows, where m=[l/N], and the number of blocks can be computed through n=M-m+1. An important point here is that each block is overlapped by pervious and/or subsequential blocks along the rows. For each block, define a function to measure its first-order smoothness

$$f = \sum_{u=2}^{m} \sum_{v=2}^{N-1} \left| \mathrm{C}_{u,v} - \frac{\mathrm{C}_{u-1,v} + \mathrm{C}_{u+1,v} + \mathrm{C}_{u,v-1} + \mathrm{C}_{u,v+1}}{4} \right|.$$

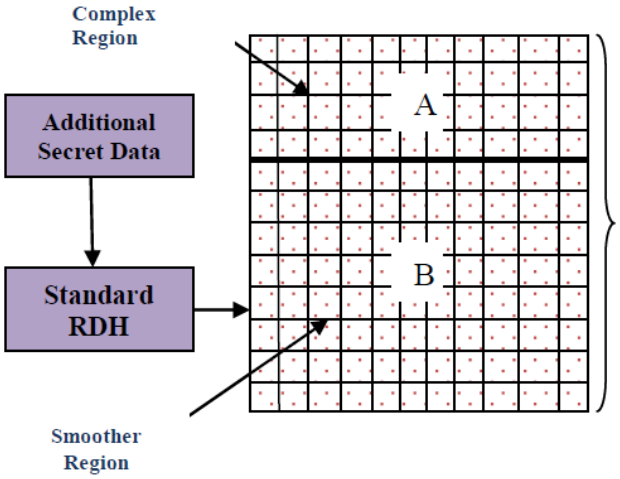

Figure 3: Image Partition

B. Data Embedding by Histogram Shifting

For data embedding we follow histogram shifting technique proposed by Zhicheng Ni, Yun-Qing Shi, Nirwan Ansari, and Wei Su in IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, VOL. 16, NO. 3, MARCH 2006. According to the algorithm we first generate histogram of a image . Then we find pairs of peak points and minimum points. We embed data in peak points. The following flowchart shows the algorithm.

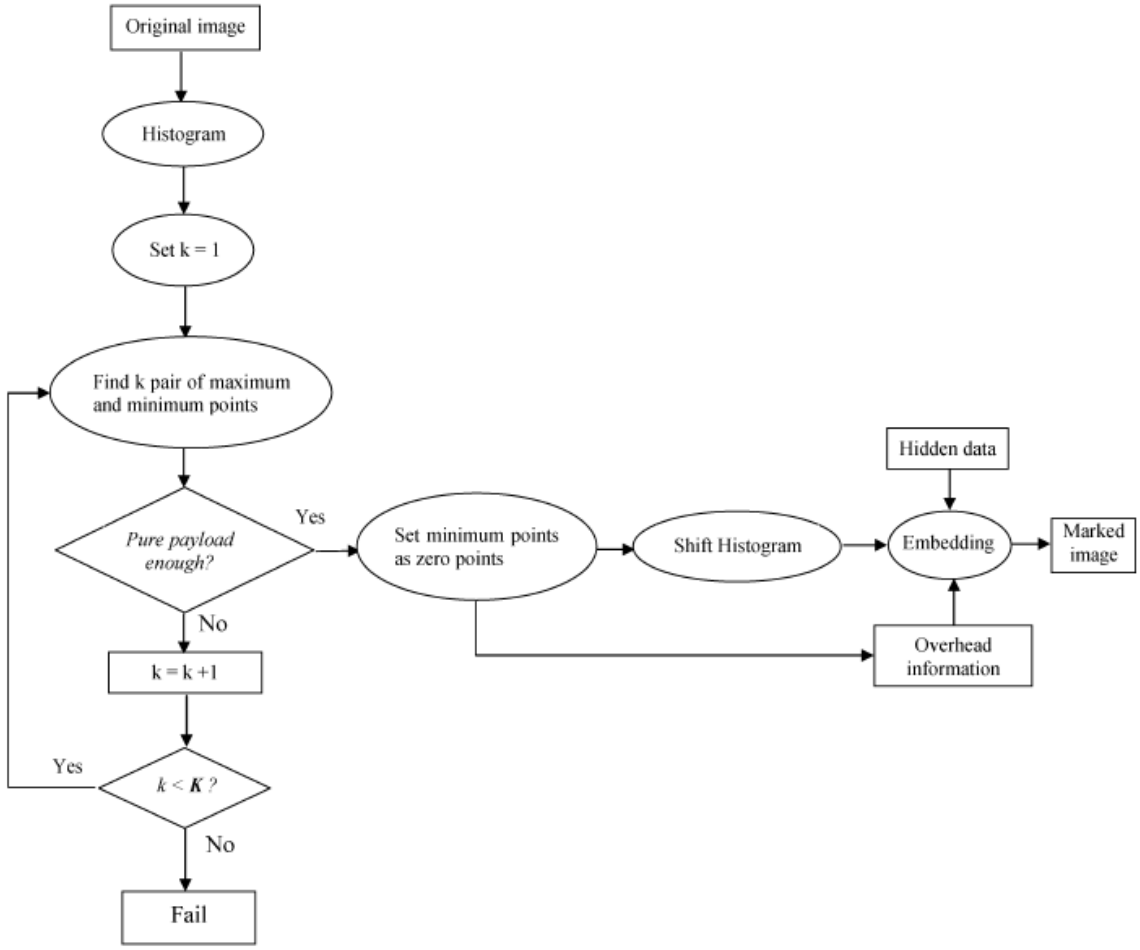

Figure 4: Algorithm for data embedding

In this case, the actual data embedding capacity C calculated as follows:

C=h(a)-O

Where O denotes the amount of data used to represent the overhead information. It is also referred to as pure payload. For data extracting we follow the reverse procedure. Fig5 shows the flowchart of the algorithm.

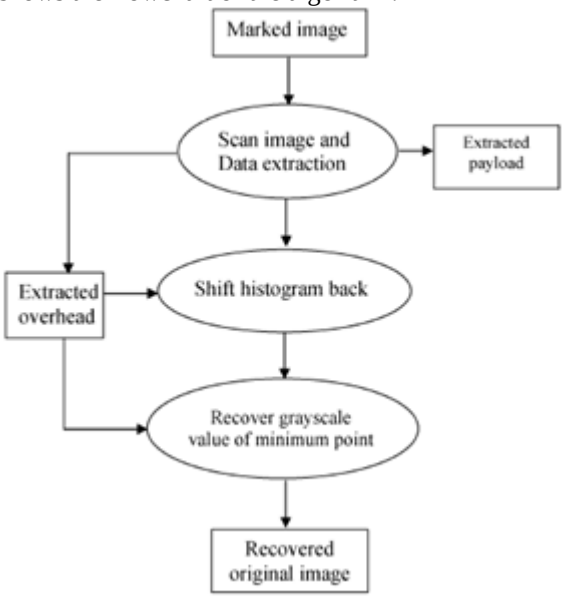

Figure 5: Data Extracting Algorithm

C. Encryption of the image with SDS algorithm

The security of the encrypted image is the most important here. In the existing system of reversible data hiding, key based algorithms are used. To enhance the security of the image we are using keyless encryption using the concept of visual cryptography. We use SDS algorithm for our image encryption. It includes three steps:

Sieving: Sieving as the name suggests involves filtering the combined RGB components into individual R, G and B components. The granularity of the sieve depends the range of values that R/G/B component may take individually. To make the process computationally inexpensive, sieving uses the XOR operator.

Division: Having filtered the original image into the R, G and B components, the next step involves dividing the R, G and B components into z parts/ shares each.
R _ (RA, RB, RC,----------, RZ)
G _ (GA, GB, GC,----------, GZ)
B _ (BA, BB, BC,----------, BZ)

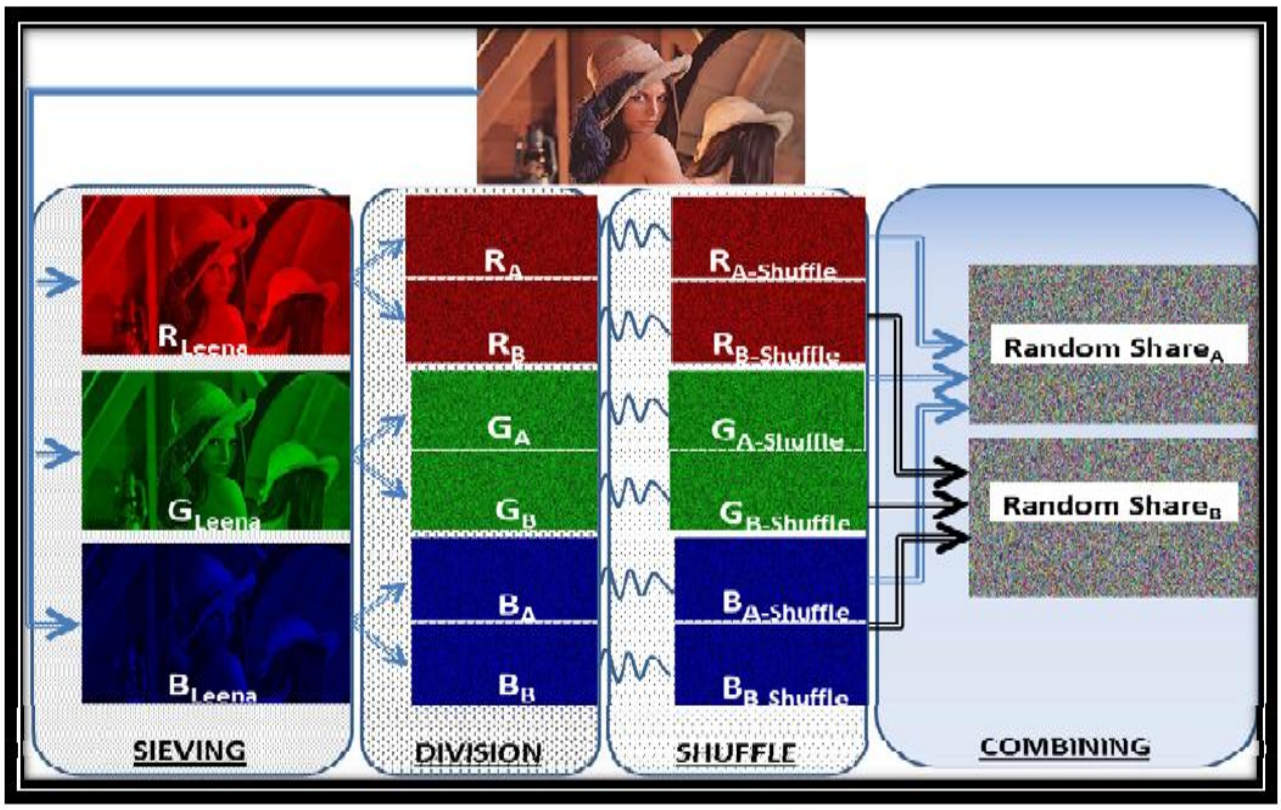

Figure 6: Steps involved generating two random shares

While dividing it is ensured that each element in RA-Z, GA-Z and BA-Z is assigned values randomly, such that the entire domain is available for randomized selection; in case x = 8,then individual elements should be randomly assigned a value varying from 0- 255. The shares so generated should be such that (RA, RB, RC,---------- RZ) should regenerate R and similarly for G/B components(Refer fig 6).

Shuffling: Though experimental results have shown that the random shares created by division in no way exhibit any resemblance to the original image, but as a second step towards randomizing the generated shares i.e. RA-Z, GA-Z and BA-Z, we perform the shuffle operation. This involves shuffling the elements in the individual shares. The sequence in which the elements within the shares are shuffled depends on the value of one of the other shares generated from the same primary color. In other words RB decides how RA is shuffled, RC decides how RB is shuffled, ---------- RZ decides RZ-1 is shuffled and RA decides how Rz is shuffled. The shuffling operation uses the comparison operator on the LSB of the determining element to decide the shuffle sequence.

Having carried out the above three operations the generated shares are combined to generate the final z random shares (RS).
RSA -(RA- shuffle, GA- shuffle and BA- shuffle)
RSB -(RB- shuffle, GB- shuffle and BB- shuffle)
- - - -
RSZ- (RZ,- shuffle GZ- shuffle and BZ- shuffle)
The random shares so generated individually convey no information about the secret image, however to recover the original image all the random shares would be required.

D. Data Extraction

1) Again separate the R-G-B component matrix of the image blocks.
2) Apply the procedure of data extraction of histogram shifting.
3) By above process the one by one bit will be extracted from the pixel pairs.
4) Then applying the proper encoding technique generates the characters for the extracted binary data bits to get the hidden text.

3. Conclusion

Proposed scheme gives a completely new framework for reversible data hiding. Partitioning the image logically into smoother and complex region improves the performance and efficiency of RDH algorithm. Reserving room from encrypted image is relatively difficult and sometimes inefficient; the proposed scheme reserves the room before encryption. In the proposed scheme the .txt file of huge size can be hidden by maintaining the quality of retrieved image.

Providing the security to the image is also a major area of concern when its storage or transmission is considered. For image encryption after hiding a data instead of using any standard cipher, a method of visual cryptography is used. For retrieving the complete image, all the random shares will be required. Image so retrieved will be same as original image. Proposed scheme guarantees the lossless retrieval of the image so as data. After retrieving the image hidden data will be extracted lossless.

References

[1] Kede Ma. Weiming Zhang, Xianfeng Zhao, Nenghai Yu,Fenghua Li, *"Reversible Data Hiding in Encrypted Images by ReservingRoom Before Encryption",* IEEE Trans on Information Forensics and security, Vol. 8, No. 3, March 2013

[2] Moni Naor, Adi Shamir," *Visual Cryptography*", in Proc. EUROCRYPT"94, Berlin, Germany, 1995, vol. 950, pp. 1- 12,Springer-Verlag, LNCS

[3] Siddharth Malik, Anjali Sardana, Jaya, "A Keyless Approach to Image Encryption",2012 international conference on Communication systems and Network Technologies ©2012 IEEE

[4] C. Vinoth Kumar, V. Nataranjan and Deepika Bhogadi, "*High capacity Reversible Data hiding based on histogram shifting for medical image*", International Conference on Communication and Signal Processing, April 3-5 2013, India © IEEE 2013

[5] V Yu, Song Wei, *"Study on Reversible Data Hiding Scheme for Digital Images",* 2nd International Asia Conference on Informatics in Control, Automation and Robotics,(CAR) 2012

[6] A. Shamir,―How to share a secret,‖ Commun.ACM, vol. 22, no. 11, pp.612–613, 1979.

[7] P. Tsai, Y. C. Hu, and H. L. Yeh, "Reversible image hiding scheme using predictive coding and histogram shifting," *Signal Process., vol.*89, pp. 1129–1143, 2009.

[8] Yi-Hui Chen, Ci-Wei lan and Chiao Chih Huang, *" A verifiable Visual Cryptography Scheme*", Fifth International Conference and Evolutionary Computing © IEEE 2011.

[9] Wen Chung Kuo, Po Yu Lai, Lih Chyau Wuu, *"Adaptive Reversible Data Hiding Based on Histogram"*, 10th International Conference on Intelligent Systems Design and Application, © IEEE 2010 (2002) The IEEE website. [Online]. Available: http://www.ieee.org/

[10] Jose, R.; Abraham, G, "*A separable reversible data hiding in encrypted image with improved performance*", Emerging Research Areas and 2013 International Conference on Microelectronics, Communications and Renewable Energy(AICERA/ICMiCR), 2013 Annual International Conference ©IEEE 2013.