Steganography using RSA Algorithm

Jatinder Kaur, Ira Gabba

Abstract— Steganography is the art and science of writing hidden messages in such a way that no one, apart from the sender and intended recipient, suspects the existence of the message, a form of security through obscurity. It is an emerging area which is used for secured data transmission over any public media. In this study a novel approach of image steganography based on LSB (Least Significant Bit) insertion and RSA encryption technique for the lossless jpeg images has been proposed. In this paper, we present a strategy of attaining maximum embedding capacity in an image in a way that maximum possible neighboring pixels are analyzed for their frequencies, to determine the amount of information to be added in each pixel. The technique provides a seamless insertion of data into the carrier image and reduces the error assessment and artifacts insertion required to a minimal. We justify our approach with the help of an experimental evaluation on a prototypic implementation of the proposed model.

Keywords— Cryptography, Steganography, RSA Algorithms.

I. INTRODUCTION

Steganography is the technique of hiding confidential information within any media. Steganography is often confused with cryptography because the two are similar in the way that they both are used to protect confidential information. The difference between the two is in the appearance in the processed output; the output of steganography operation is not apparently visible but in cryptography the output is scrambled so that it can draw attention. Steganlysis is process to detect of presence of steganography. Most of the steganography techniques use images a stego-medium. Information can be hidden in images through many different ways. The most common approaches to information hiding in images are: Least significant bit (LSB) insertion, Masking and filtering techniques, Algorithms and transformations. Masking and filtering techniques hide information by marking an image in a manner similar to paper watermarks. The least significant bit insertion (LSB) is the most widely used image steganography technique. It embeds message in the least-significant bits of each pixel. In order to increase the embedding capacity, two or more bits in each pixel can be used to embed message, which has high risk of delectability and image degradation. The LSB techniques might use a fixed least significant bit insertion scheme, in which the bits of data added in each pixel remains constant, or a variable least significant bit insertion, in which the number of bits added in each pixel vary on the surrounding pixels, to avoid degrading the image fidelity In this paper we discuss the embedding of text into image through variable size least significant bit insertion. The process of insertion of text in our proposed approach is not sequential, rather it follows a random order, based on a random algorithm.

Manuscript Received August, 2013.

Jitendra, M.Tech Scholar, Department of Computer Science & Engineering, CT Institute of Engineering and Management Technology, NawanShahar (Punjab), India.

Ira Gabba, M.Tech Scholar, Department of Computer Science & Engineering, Indo Global College of Engineering, Chandigarh, India.

The technique proposed aims at providing not only maximum insertion capacity, but also performs a maximum analysis of surrounding pixels to determine the embedding capacity of each pixel. The process results in a stego-image which is very much similar in appearance to the original image. we propose a steganography model that ensures maximum embedding of information in both gray scalable and colored images, and also ensures that maximum pixels are analyzed to determine the embedding capacity. This would lead to a reduction of the overall error induction in the image. The stego-image obtained after application of this analysis would not only have maximum amount of information, but would also have the minimum difference in appearance with the original image. Steganography is one of the most vital research subjects in the field of security communications. It differs from cryptography in the sense that where cryptography focuses on keeping the contents of a message secret, steganography focuses on keeping the existence of a message secret. Another form of information hiding is digital watermarking, which is the process that embeds data called a watermark, tag or label into a multimedia object such that watermark can be detected or extracted later to make an assertion about the object. The object may be an image, audio, video or text only.

Fig. 1 shows the different categories of file formats that can be used for steganography techniques.

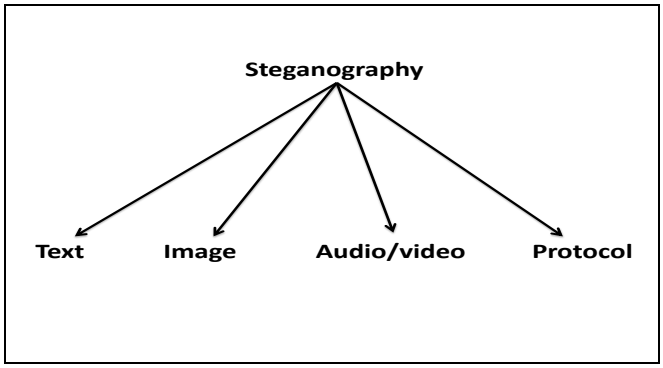

Figure 1: Different types of Steganography Techniques.

Steganography applications that hide data in images generally use a variation of least significant bit (LSB) embedding . In LSB embedding, the data is hidden in the least significant bit of each byte in the image. The size of each pixel depends on the format of the image and normally ranges from 1 byte to 3 bytes. Each unique numerical pixel value corresponds to a color; thus, an 8-bit pixel is capable of displaying 256 different colors. Given two identical images, if the least significant bits of the pixels in one image are changed, then the two images still look identical to the human eye. This is because the human eye is not sensitive enough to notice the difference in color between pixels that are different by 1 unit. Thus, steganography applications use LSB embedding because attackers do not notice anything odd or suspicious about an image if its pixel's least significant bits are modified.

Steganography is no different; attackers combat steganography using steganalysis.

Steganalysis is a process where attackers analyze an image to determine whether it has hidden data in it. A common steganalysis approach is to graph the pixel values of an image that is suspected of containing hidden data. Statistical analysis is then performed on the graphed pixel values . The attackers hope to find anomalies in the statistical analysis of these images. These anomalies may indicate that the image contains a hidden message, and the anomalies may offer some insight into how to extract the hidden message. In this paper a specific image based steganographic model has been proposed which uses a JPEG lossless image as the cover data and the secret information is embedded in the cover to form the stego image. Before embedding the secret information has been encrypted using the public key RSA algorithm. This application also provides a utility to convert JPEG images in the lossless JPEG images. We choose JPG images for this application because most images shared by people today are in the JPG format. Thus to an attacker, the fact that an image other than that of JPEG format is being transferred between two entities could hint of suspicious activity. So this can be an improvement by using JPEG Lossless images.

To combat steganalysis, this application performs an analysis on the user's library of images. This analysis allows users to hide their data in the image that is least likely to be vulnerable to steganalysis.

Steganography is the art and science of hiding communication; a steganographic system thus embeds hidden content in unremarkable cover media so as not to arouse an eavesdropper's suspicion. In the past, people used hidden tattoos or invisible ink to convey steganographic content. Today, computer and network technologies provide easy-to-use communication channels for steganography. Essentially, the information-hiding process in a steganographic system starts by identifying a cover medium's redundant bit. The embedding process creates a stego medium by replacing these redundant bits with data from the hidden message. Modern steganography's goal is to keep its mere presence undetectable, but steganographic systems— because of their invasive nature—leave behind detectable traces in the cover medium. Even if secret content is not revealed, the existence of it is: modifying the cover medium changes its statistical properties, so eavesdroppers can detect the distortions in the resulting stego medium's statistical properties. The process of finding these distortions is called statistical steganalysis. This article discusses existing steganographic systems and presents recent research in detecting them via statistical steganalysis. Other surveys focus on the general usage of information hiding and watermarking or else provide an overview of detection algorithms. Here, we present recent research and discuss the practical application of detection algorithms and the mechanisms for getting around them.

II. PROPOSED SYSTEM

The proposed system basically provides us with the system, in which we will hide the information in the form of text file behind the image, which will works as an input to the cryptography part. In the cryptography section we basically divides the produced image in the form of shares(share1 and share2). On the shares we apply the RSA Algorithm, that is we are applying the keys on it, so that we are able to provide better security.

Steganography is a special kind of cryptography that hides one piece of information inside of another. This is the art and science of conveying information in such a way that the presence of one object within the other is unnoticeable.

It comes from the Greek words steganós steganos (covered) and graptos graptos (writing), literally covered writing, in the sense of a hidden thing. In steganography the text or image in question is invisible although it resides in the interior of an apparently normal piece of other information, like a text an image or a soundtrack. Unlike cryptography where the message is clearly visible although you need the key to decipher it, in this case the information cannot be seen unless the correct procedure is applied to the text or image where it resides. In most circumstances, the security is heightened by a required key to reverse the Steganography process.

1. To develop a RSA algorithm for cryptography. Firstly we develop a algorithm for the loading the image in the database. We develop a code for steganography by using a novel score-level combination strategy.
2. Designing and implement the developed algorithm for the steganography purpose for the message and image. Develop a code for Procedures For Hide Text. This procedure hides the encrypted data into the image by searching the best position in the image. The best position defines those Least Significant Bits of the image which extremely match with the encrypted data bits. The output of this procedure is the updated image in which data is hidden.
3. Procedure for Reveal Text: This procedure reveals the secret message which is hidden in the best position of the stego image. This message is in the encrypted form so the message is decrypted by the receiver's private key. Then the original message is presented to the receiver.
4. The objective of steganography is to hide a secret message within a cover-media in such a way that others cannot discern the presence of the hidden message. Technically in simple words "steganography means hiding one piece of data within another".
5. Modern steganography uses the opportunity of hiding information into digital multimedia files and also at the network packet level.
6. The stego function operates over cover media and the message (to be hidden) along with a stego-key (optionally) to produce a stego media (*S*).
7. In this method binary equivalent of the message (to be hidden) is distributed among the LSBs of each pixel.

There are various algorithms which are been applied on the image so as to produce the outcome i.e the original message.

The Embedding Algorithm:

In the technique presented here, the colour image is decomposed into three components R, G and B. Watermark information will be embedded in the G plane using equation 1 to produce G'. Assume that *f (i, j)* represents the pixel of the component of the RGB representation of the colour host image, *w (i, j)* represents the binary pixel of the watermark.

$$F_k(u,v) = DCT\{f_k(i,j)\},$$

If w(i,j)=1 then

$$F_k(x,y) = \begin{cases} \Delta Q_e\left(\frac{F_k(x,y)}{\Delta}\right) & x,y \in H_k \quad 1 \le k \le N_{HB} \\ F_k(x,y) & x,y \notin H_k \quad 1 \le k \le N_{HB} \end{cases}$$

If w(i,j)=0 then

$$F_k(x,y) = \begin{cases} \Delta Q_o\left(\frac{F_k(x,y)}{\Delta}\right) & x,y \in H_k \quad 1 \le k \le N_{HB} \\ F_k(x,y) & x,y \notin H_k \quad 1 \le k \le N_{HB} \end{cases}$$

Figure2: The Embedding Algorithm.

Where Qe is the quantization to the nearest even number and Qo is the quantization to the nearest odd number Δ is a scaling quantity and it is also the quantization step used to quantize either to the even or odd number. The predefined coefficients in each 8x8 sub block are represented by NHB. The binary watermark digits are randomly scrambled using a secret key; this scrambling process is essential to reduce the spatial correlation between the host image and the embedded watermark. After the scrambling process, a shuffle scheme is applied for each binary watermark copy before embedding. Different shuffle schemes could be applied. Simple shuffle technique is to reshape the watermark copy as vector and shift the vector by different shifts before the binary watermark digits are randomly scrambled using a secret key; this scrambling process is essential to reduce the spatial correlation between the host image and the embedded watermark. After the scrambling process, a shuffle scheme is applied for each binary watermark copy before embedding. Different shuffle schemes could be applied. Simple shuffle technique is to reshape the watermark copy as vector and shift the vector by different shifts before the .Where WSB is the number of watermark shifted bits, This shift is necessary to reduce the spatial relation and to increase the robustness against vertical cropping attacks.

The Extraction Algorithm:

The embedded information w (i,j) can be extracted by performing 8x8 DCT transform for the watermarked host image and indicate the same coefficients of the host image that carries the 16 bits of the embedded watermarks using the same secret key in the initial scrambling operation. The scrambled watermarks are descrambled to get the original watermarks. The watermark information can be retrieved by using a reverse process to the shift scheme which has been applied in the embedding process. This will yield the original bits order. It is worth to mention that although the proposed scheme is blind since it does not require the original host image for reconstruction, but it requires information such as the sizes of both host image and watermark image. Finally, discard the totally degraded copy of the extracted watermarks and then calculate the average by summing the resultant watermark copies divided by their number, select the resultant average watermarks as the final reconstructed watermark or choose one copy of the extracted watermarks as the final watermark if it provides better result than the resultant average watermark. The bit extraction formula is shown in equation.

$$If\ Q\left(\frac{F_k(x,y)}{\Delta}\right) is\ odd\ \ then\ w(i,j)=0$$

$$If\ Q\left(\frac{F_k(x,y)}{\Delta}\right) is\ even\ then\ w(i,j)=1$$

Figure 3: The Extraction Algorithm.

For applying the Keys(Public Key), on the share we basically use the RSA Algorithm, and following are the various steps we should take care of,

Let's first see how RSA works, and then examine why it works.

Choice of Bob's public and private keys:

1. Choose two large prime numbers, p and q.

(Larger the values, the more difficult it is to break RSA, and the longer it takes to encode/decode. It is recommended that the product of p and q be on the order of 1024 bits for corporate use and 768 bits for use with "less valuable information".

2. Compute n = pq and z = (p-1)(q-1).
3. Choose a number, e, less than n, which has no common factors (other than 1) with z.

(In this case, e and z are said to be relatively prime. The letter e is used since this value will be used in encryption).

4. Find a number, d, such that ed-1 is exactly divisible (that is, with no remainder) by z.

(The letter d is used because this value will be used in decryption. Put another way, given e, we choose d such that the integer remainder when ed is divided by z is 1. The integer remainder when an integer x is divided by the integer n, is denoted as x mod n).

5. $K_B^+ = (n, e)$; $K_B^- = (n, d)$

RSA's Encryption/Decryption Algorithm:

Suppose Alice wants to send Bob a bit pattern, or number, m such that m < n.

To encrypt the message, Alice uses Bob's public key and determines the cipher text, c as:

$c = m^e \bmod n$

To decrypt the message, Bob uses Bob's private key and determines the plain text, m as:

$m = c^d \bmod n$

If p and q are prime, and n = pq, then $x^y \bmod n = x^{(y \bmod (p-1)(q-1))} \bmod n$

This rule implies that $(m^e)^d \bmod n = m$ because, as per number theory rule

$(m^e)^d \bmod n = m^{(ed \bmod (p-1)(q-1))} \bmod n$

$= m^1 \bmod n$ (as the integer remainder when ed is divided by z is 1)

$= m$

Similarly, $(m^d)^e \bmod n = m$.

By applying the above we are able to provide keys on the shares (share1 and share2) and the sender site and this process in known as encryption , where as the on the receiver site we also apply the keys and the process is known as the decryption.

The following are the various GUI outcome of the proposed system.

Figure4: The Basic layout of the GUI.

In the above figure, we just show the starting GUI of the proposed system, it consists of the Encoder and the Decoder part of the system.

The basic function of the encoder part is to hide the information or text. And provide this information to the decoder. And the decoder will decode the original message from the image.

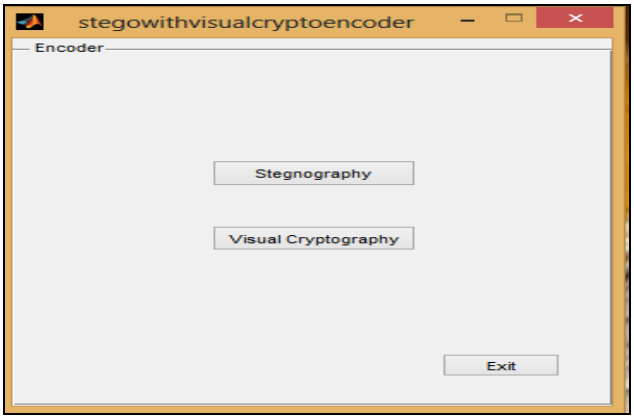

Figure 5: The layout of the Encoder GUI part.

The encoder part consist of the steganography and the cryptography part, in this the original message is hidden with the help of steganography and the cryptography and then that message is sent to the receiver. So that it will get the original message.

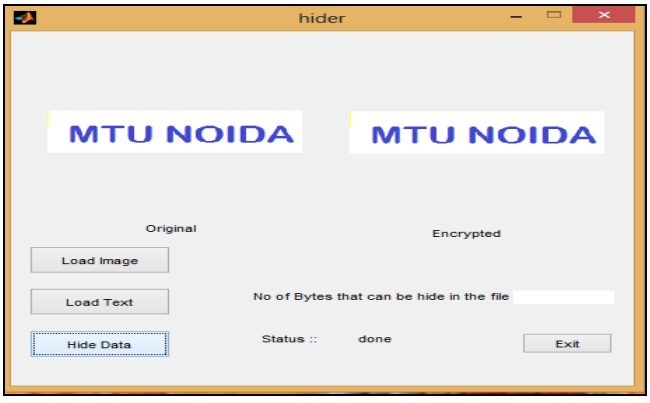

Figure 6: The GUI in which steganography part is carried out.

In the above figure we basically hide the text or message , in an image so the unauthorised user is not able to access the data easily.

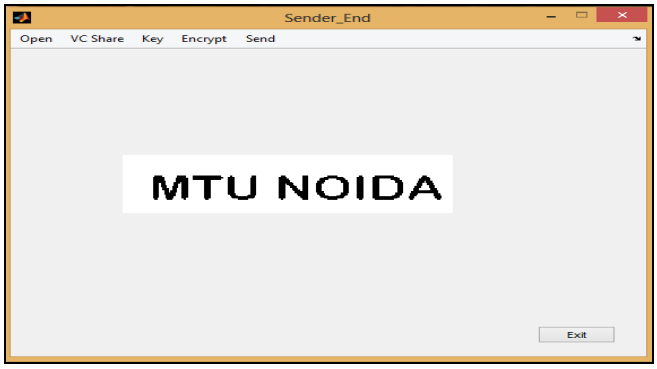

Figure 7: The GUI Used for the cryptography part.

The above figure basically shows the cryptography part of the proposed system, in this the outcome image of the steganography part will act as a input of the cryptography system and further on this image we divide it into shares and we also apply the key's on it for better security purposes.

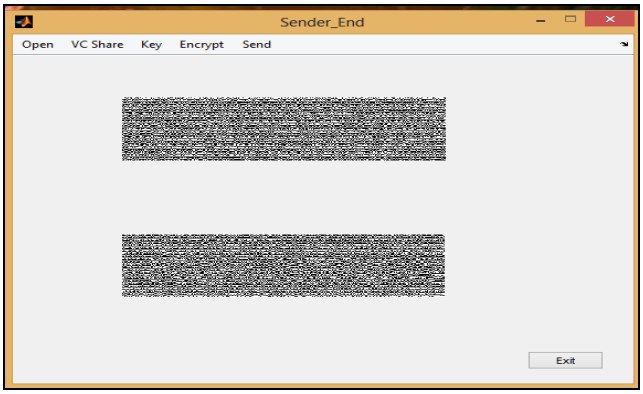

Figure 8: The input image is divided into shares.

The input image is further divided into the shares (share 1 and share2.)

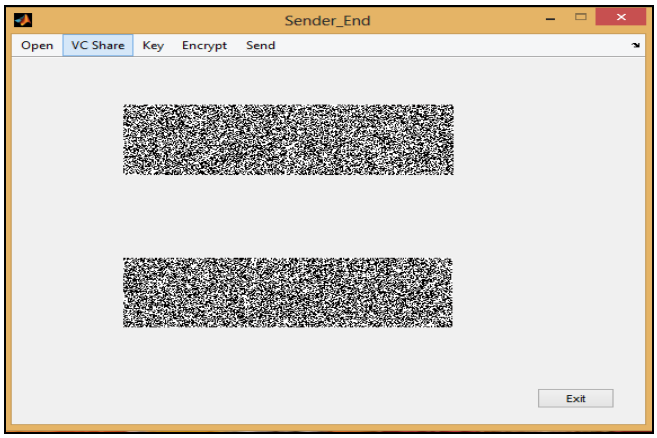

Figure 9: The public key is applied on the shares

In the above figure, we basically apply the key by using the RSA algorithm.

A utility is developed for RSA algorithm. With the help of this utility user can generate public and private keys by providing two prime numbers.

The user can enter prime numbers according to his/her choice and an automatic generated prime numbers are also provided as an aid to the user. The utility generates a pair of public and private key, which can be saved in the file or user may remember these keys.

The image utility takes the jpeg image or the folder of images (users library) and convert them into lossless jpeg images. The path is provided by the user in which these converted images are saved.

In the text box user can type the secret message and this message is encrypted with the receiver's RSA public key. This key is loaded from the file where it was saved. The stego image is obtained by user at the receiver side.

The user then gets the message in the encrypted form. The recipient then decrypts the message only with his private key which is made available to him through any media. After decryption user gets the original message. The reveal text utility is developed for this.

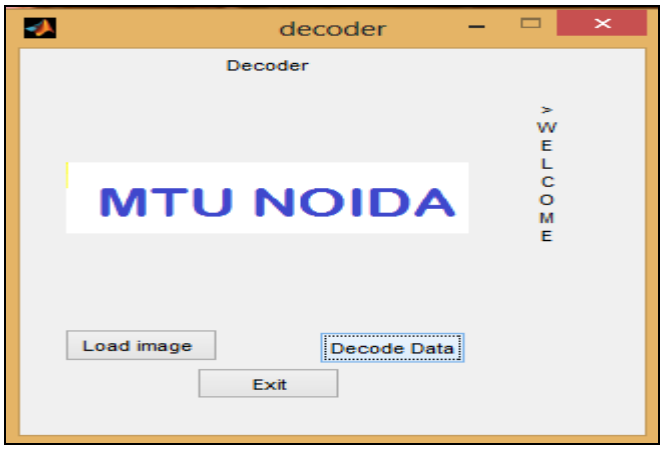

Figure 10: The layout of the Decoder part.

In the decoder part we retrieve the original message from the input image. In this proposed system, only the authorized user will be able to access the data or message properly.

III. CONCLUSION

The steganography is used in the covert communication to transport secrete information. In this paper Steganography using visual cryptography is proposed. The secret message is embedded into smaller matrix of size 8x8 and inserted into input image. In future the technique can be verified for robustness. We added the RSA algorithm process with it.

Presently, this application supports hiding data in lossless jpg images. Future improvement of this application would be extending its functionality to support hiding data in video files or in other file format.

REFERENCES

1. Raja K B, C R Chowdary, Venugopal K R, L M Patnaik. (2005) :"A Secure Steganography using LSB, DCT and Compression Techniques on Raw Images," IEEE International Conference on Intelligence Sensing and Information processing, pp.171-176.
2. Kumar V and Kumar D. (2010): "Performance Evaluation of DWT Based Image Steganography," IEEE International Conference on Advance Computing, pp. 223-228.
3. Weiqi Luo, Fangjun Huang, and Jiwu Huang. (2010): "Edge Adaptive Image Steganography Based on LSB Matching Revisited," IEEE Transactions on Information Forensics and Security, no. 2, vol. 5, pp. 201-214.
4. R O El Safy, H H Zayed and A El Dessouki (2009): "An Adaptive Steganographic Technique Based on Integer Wavelet Transform," International Conference on Networking and Media Convergence, pp.111-117.
5. Mathkour H, Al-Sadoon B and Touir A. (2008): "A New Image Steganography Technique. :" International Conference on Wireless Communications, Networking and Mobile Computing, pp.1-4.
6. V Vijaylakshmi,G Zayaraz and V Nagaraj. (2009):"A Modulo Based LSB Steganography Method," International Conference on Control,Automation,Communication and Energy Conservation, pp. 1-4.
7. Wien Hong, Tung-Shou Chen and Chih-Wei. (2008):"Lossless Steganography for AMBTC-Compressed Images," Congress on Image and Signal Processing, pp.13-17.
8. A W Naji, Teddy S Gunawan, Shihab A Hameed, B B Zaidan and A A Zaidan. (2009): "Stego-Analysis Chain, Session One," International Spring Conference on Computer science and Information Technology, pp. 405-409.
9. M Hassan Shirali-Shahreza and Mohammad Shirali-Shahreza. (2008): "A New Synonym Text Steganography," International Conference on Intelligent Information Hiding and Multimedia Signal Processing, pp. 1524-1526.
10. Vladimir Banoci, Gabriel Bugar and Dusan Levicky (2009): "Steganography Systems by using CDMA Techniques," International Conference on Radioelectronika, pp.183-186.
11. Chen Ming, Zhang Ru, Niu Xinxin and Yang Yixian (2006): "Analysis of Current Steganographic Tools: Classifications and Features," International Conference on Intelligent Hiding and Multimedia Signal Processing, pp. 384-387.
12. Mankun Xu, Tianyun Li and Xijian Ping. (2009): "Estimation of MB Steganography Based on Least Square Method," International Conference on Acoustics, Speech and Signal Processing, pp. 1509-1512.
13. Abbas Cheddad, Joan Condell, Kevin Curran and Paul Mc Kevitt. (2008): "Enhancing Steganography in Digital Images," Canadian Conference on Computer and Robot Vision, pp. 326-332.
14. Aos A Z, A W Nazi, Shihab A Hameed, Fazida Othman, B B Zaidan. (2009): "Approved Undetectable-Antivirus Steganography," International Spring Conference on Computer and Information Technology, pp. 437-441.
15. Daniela Stanescu, Valentin Stangaciu, Loana Ghergulescu and Mircea Stratulat. (2009): "Steganography on Embedded Devices," International Symposium on Applied Computational Intelligence and Informatics, pp. 313-318.
16. Jin-Suk Kang, Yonghee You and Mee Young Sung (2007): "Steganography using Block-Based Adaptive Threshold," International symposium on Computer and Information Sciences, pp. 1-7.
17. Mci-Ching Chen, Sos S Agaian and C L Philip Chen. (2008): "Generalised Collage Steganography on Images," International Conference on Systems, Man and Cybernetics, pp.1043-1047.
18. Neha Agarwal and Marios Savvides. (2009): "Biometric Data Hiding: A 3 Factor Authentication Approach to Verify Identity with the Single Image using Steganography, Encryption and Matching," International Conference on Computer vision and pattern recognition, pp.85-92.