NOVATEUR PUBLICATIONS
INTERNATIONAL JOURNAL OF INNOVATIONS IN ENGINEERING RESEARCH AND TECHNOLOGY [IJIERT]
ISSN: 2394-3696
VOLUME 3, ISSUE4, APR.-2016

COMPRESSED DOMAIN DATA HIDING APPROACH ON ENCRYPTED IMAGES USING AUXILIARY INFORMATION

Amruta Nehete
Department of Electronics and Telecommunication,
TSSM's BSCOER narhe, Pune, India

Deepak Pawar
Department of Electronics and Telecommunication,
TSSM's BSCOER narhe, Pune, India

ABSTRACT

This paper presents secret data concealment and compressing encrypted images based on auxiliary information generation for efficient secure transmission under insufficient bandwidth. The process starts by encryption of uncompressed original images and generation of some auxiliary information, which will be used for data compression and image reconstruction. Then, the channel provider compress the encrypted data by a quantization method with optimal parameters that are derived from a part of auxiliary information and transmit the compressed data, which includes an encrypted sub-image, the quantized data, the quantization parameters and another part of auxiliary information. In this project discrete cosine transform is used under the quantization method for encrypted image compression. The secret text message is first converted to bit streams then it will be concealed into encoded bits of generated binary map. This binary map is obtained from identifying difference between the original and interpolated image. With the help of compressed encrypted data and the secret key, the principal image content can be reconstructed at receiver side. Before image reconstruction process, hidden bits are extracted from encrypted encoded version of binary map then restoring the original bits on it. An approximated encryption image will be determined from decoded binary map and estimated key. An estimated encrypted image will be modified using quantized data under the discrete cosine transformation. Then image will be reconstructed from estimated key and modified encrypted image. The simulated result shows that used methodologies provides better performance in terms of compression ratio and reconstructed image quality.

INTRODUCTION

Larger images of greater bit depth become too large to transmit over a standard Internet connection. In order to display an image in a reasonable amount of time, techniques must be integrated to reduce the image's file size. These techniques make use of mathematical formulas to analyze and condense image data, resulting in smaller file sizes. This process is called compression [2].

Many methods are used for compression; in general these methods can be divided into two broad categories: lossless and lossy methods. In images there are two types of compression: lossy and lossless. Both methods save storage space, but the procedures that they implement are different. Lossless compression represents data in

mathematical formulas without removing any information from the original image. The original image's integrity is maintained and the decompressed image output is exactly identical to the original image input. The most popular image formats that use lossless compression is GIF (Graphical Interchange Format) and 8-bit BMP (bitmap file). Lossless compression keeps the original digital image intact without the chance of lost, but it does not compress the image to such a small file size. Lossless compression reduces bits by identifying and eliminating statistical redundancy. No information is lost in lossless compression [3].

Encryption is the conversion of electronic data into another form, called cipher text, which cannot be easily understood by anyone except authorized user. The primary purpose of encryption is to protect the confidentiality of digital data stored on computer systems or transmitted via the Internet or other computer networks. Data, often referred to as plaintext, is encrypted using an encryption algorithm and an encryption key. This process generates cipher text that can only be viewed in its original form if decrypted with the correct key. Decryption is simply the inverse process of encryption, following the same steps but reversing the order in which the keys are applied [7].

Compressing encrypted multimedia is a promising technology intended at reducing the data amount of cipher-text signals without revealing the plaintext content. Content owner encrypts the uncompressed plain signals for privacy protection; the task of compression may be left to a channel or storage-device provider who has limited available resources but not the encryption key. After receiving the compressed encrypted data, an authorized user who knows the secret key can recover the plaintext content as shown in figure 1[1].

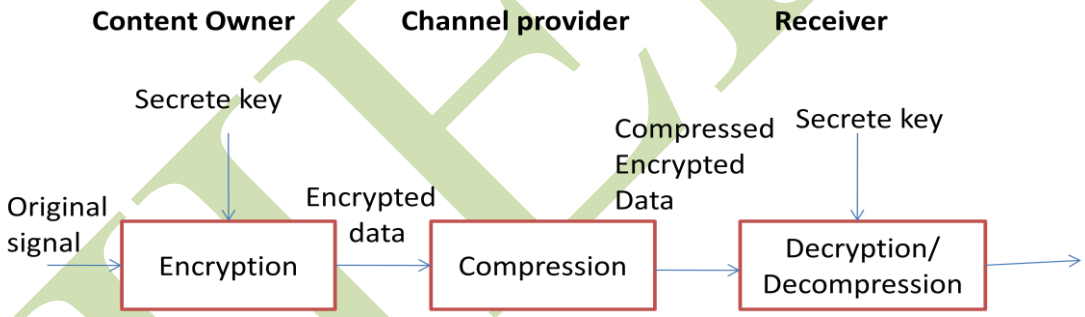

Figure 1: System of Encryption and Compression

This paper proposes a novel scheme of compressing encrypted images with the help of auxiliary information. In encryption phase, the original uncompressed images are encrypted by the content owner, and some auxiliary information is also formed when the channel bandwidth is not enough. In compression phase, the encrypted data in various DCT sub-bands are effectively compressed by using a quantization mechanism without enlightening the original content, and an optimization method with ratio-distortion criteria is employed to select the quantization parameters according to the auxiliary information. At a receiver side the principal plaintext content can be reconstructed with secret key. The experimental result shows the ratio-distortion performance of the proposed scheme is significantly better than that of existing techniques [1].

In the proposed system, the content owner firstly masks all pixel values in original uncompressed image to get an encrypted image and provides the encrypted data to the channel provider. If the bandwidth is enough, the channel provider transmits the encrypted

data. Otherwise, the channel provider sends a "bandwidth-insufficiency" message to the content owner. Then the content owner generates the auxiliary information according to the original and encrypted content and provides it to the channel provider. The auxiliary information (AI) is made up of two parts that will be used for data compression and image reconstruction. Then, the channel provider may compress the coefficients in encrypted domain by a quantization method with the aid of the first part of auxiliary information (AI 1), and transmits the compressed data, which include an encrypted sub-image, the quantized data, the quantization parameters and the second part of auxiliary information (AI 2), through a channel. At receiver side, only authorized user can reconstruct the principal content of original image by retrieving the coefficient values. By involving the auxiliary information into encrypted image compression, the ratio-distortion performance is improved and the computational complexity is reduced. Figure 2 presents the proposed scheme [1].

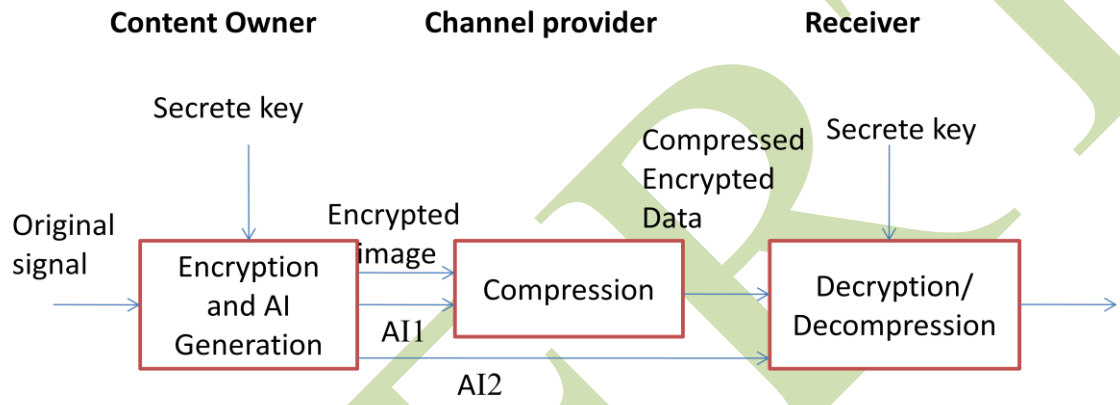

Figure 2: Block diagram of proposed method

PROPOSED METHOD

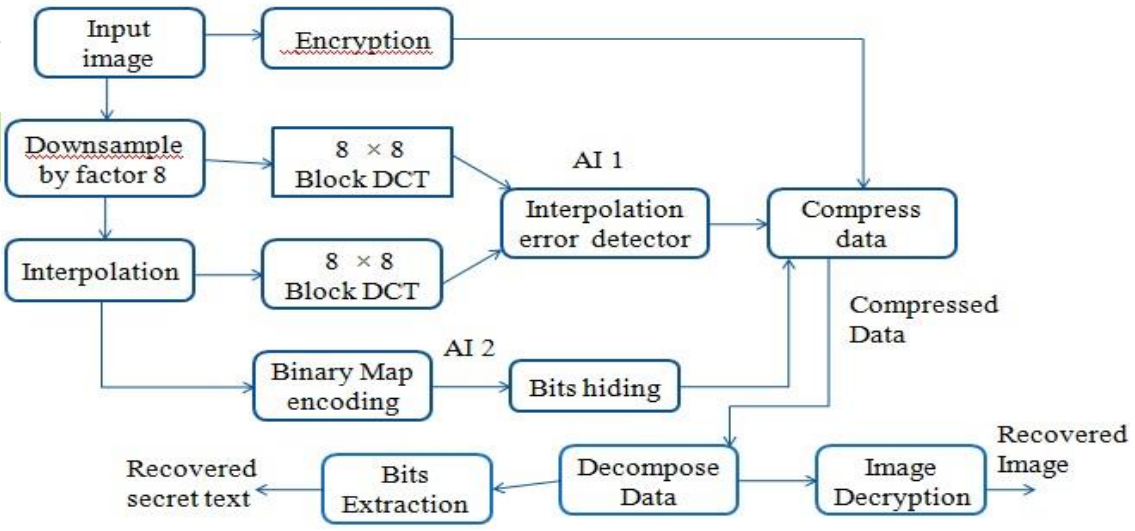

Figure 3: Block diagram of proposed method

ENCRYPTION

The content owner encrypts the original image by adding pseudo-random numbers into the pixels. Assume the original image is in uncompressed format and the pixel values are within [0, 255]. Numbers of the rows and the columns in the original image are denoted as $N1$ and $N2$, and the number of all pixels as N ($N = N1 \times N2$), implying that the bit amount of original image is $8 \times N$. The content owner pseudo-randomly generates N integers uniformly distributed within [0, 255], and employs one-by-one addition modulo 256 and produce an encrypted image [1].

$$c(i,j) = mod[p(i,j) + k(i,j), 256] \qquad 1 \le i \le N1,\ 1 \le j \le N2 \qquad (1)$$

Here, p (i, j) are the gray values of pixels at positions (i, j), k (i, j) are the pseudo-random numbers derived from a secret key, and c (i, j) are the data of encrypted image. The values of c (i, j) obey a uniform distribution within [0, 255]

GENERATION OF AUXILIARY INFORMATION A1

If the bandwidth is enough, no other operation is needed. Otherwise, after receiving a "bandwidth-insufficiency" message from the channel provider, the content owner should generate the auxiliary information and provides it for the data compression by channel provider and image reconstruction at receiver side [1]. We assume both $N1$ and $N2$ are multiples of 8. The content owner generates a down sampling sub-image with a size of $N1/8 \times N2/8$,

$$pD(i,j) = p(8i, 8j) \qquad (2)$$
$$1 \le i \le N1/8, \quad 1 \le j \le N2/8$$

DCT

In the image compression algorithm, the input image is divided into 8×8 or 16×16 blocks, and the two-dimensional DCT is computed for each block. The DCT coefficients are then quantized, coded, and transmitted. The receiver decodes the quantized DCT coefficients, computes the inverse two-dimensional DCT of each block, and then puts the blocks back together into a single image. For typical images, many of the DCT coefficients have values close to zero. These coefficients can be discarded without seriously affecting the quality of the reconstructed image

DCT DEFINITION

The discrete cosine transform (DCT) represents an image as a sum of sinusoids of varying magnitudes and frequencies. The DCT function computes the two-dimensional discrete cosine transform (DCT) of an image. The DCT has the property that, for a typical image, most of the visually significant information about the image is concentrated in just a few coefficients of the DCT. For this reason, the DCT is often used in image compression applications. Then, the content owner divides the original and interpolated images into a number of 8 and performs 2D discrete cosine transform in blocks sized 8×8 each block.

$$\begin{bmatrix} P(8i+1,8j+1) & \cdots & P(8i+1,8j+8) \\ \vdots & \ddots & \vdots \\ P(8i+8,8j+1) & \cdots & P(8i+8,8j+8) \end{bmatrix} = \mathrm{DCT}\left\{ \begin{bmatrix} p(8i+1,8j+1) & \cdots & p(8i+1,8j+8) \\ \vdots & \ddots & \vdots \\ p(8i+8,8j+1) & \cdots & p(8i+8,8j+8) \end{bmatrix} \right\} \qquad (3)$$

$$\begin{bmatrix} G(8i+1,8j+1) & \cdots & G(8i+1,8j+8) \\ \vdots & \ddots & \vdots \\ G(8i+8,8j+1) & \cdots & G(8i+8,8j+8) \end{bmatrix} = \mathrm{DCT}\left\{ \begin{bmatrix} g(8i+1,8j+1) & \cdots & g(8i+1,8j+8) \\ \vdots & \ddots & \vdots \\ g(8i+8,8j+1) & \cdots & g(8i+8,8j+8) \end{bmatrix} \right\} \quad (4)$$

The content owner calculates the square roots of the average interpolation distortion in each sub-band by viewing the coefficients as 64 sub-bands,

$$\sigma^{(u,v)} = \sqrt{\frac{\sum_{i=0}^{N_1/8-1}\sum_{j=0}^{N_2/8-1}\left[P(8i+u,8j+v)-G(8i+u,8j+v)\right]^2}{N_1N_2/64}}, \quad 1 \le u,v \le 8 \quad (5)$$

BITS REPLACEMENT BASED DATA HIDING

LSB technique can be used wherever we want to store confidential information on a standalone PC or one which is shared among several users. Wherever this kind of information is to be preserved in a manner that only legal user should be able to retrieve it whenever needed, by simple way LSB is a better solution
[4].
Least Significant Bit (LSB) encoding is the easiest of the techniques used for embedding secret or confidential information in digital images. For a gray scale bitmap (BMP), using the LSB of each byte (8 bits) in an image, a secret message of size 1/8th of the Cover image can be stored. This can be easily done by directly substituting every bit of the secret message into every LSB. For a 24 bit color image since there are 3 bytes for every pixel, 3 bits of data can be stored in each pixel, so the capacity to store increases by 3 times thus making it 3/8 of the cover image size. If the message to be embedded is a text message a secret message of size 1/7th of the grayscale cover image can be stored and in a 24 bit color image as cover a text message of size 3/7 can be embedded.
The confidential information which is embedded in the cover image can be an image (gray scale, binary or colour image), text or even audio. As the type and size of confidential information varies, the embedding capacity varies for a particular type of cover used. LSB technique can be used either by directly replacing the cover LSB's by the secret information bits, or add one more level of security, it can be encrypted and then inserted into the LSB's of the Cover. The resultant Stego-image which holds the secret message is also a 8-bit gray scale image. Difference between the cover image and the Stego-image is not visually noticeable [5].

COMPRESSION OF ENCRYPTED IMAGES

After having the encrypted image, if the channel resource is sufficiently abundant so that any compression is needless, the channel provider may transmit the encrypted image directly. In this case, clearly, an authorized user who has the secret key can decrypt the received data to retrieve the original image without any distortion. If the channel resource is limited, the channel provider should obtain the auxiliary information from the content owner, and then perform a data compression using a quantization method before transmission. The compression procedure is as follows.

The compression will be performed in 64 DCT sub-bands with different quantization parameters. The channel provider firstly implements 2D DCT in the encrypted image with a block-by-block manner.

$$\begin{bmatrix} C(8i+1,8j+1) & \cdots & C(8i+1,8j+8) \\ \vdots & \ddots & \vdots \\ C(8i+8,8j+1) & \cdots & C(8i+8,8j+8) \end{bmatrix} = \mathrm{DCT}\left\{\begin{bmatrix} c(8i+1,8j+1) & \cdots & c(8i+1,8j+8) \\ \vdots & \ddots & \vdots \\ c(8i+8,8j+1) & \cdots & c(8i+8,8j+8) \end{bmatrix}\right\}, \quad 0 \le i \le N_1/8-1, \quad 0 \le j \le N_2/8-1 \qquad (6)$$

Then, he reorganizes the coefficients in each sub-band as a vector, which is denoted as [$C(u,v)(1)$, $C(u,v)(2)$, …, $C(u,v)(N1N2/64)$] After that, perform orthogonal transform for the vectors to calculate

$$\begin{bmatrix} D^{(u,v)}(1) \\ D^{(u,v)}(2) \\ \vdots \\ D^{(u,v)}(N_1N_2/64) \end{bmatrix} = \mathbf{A} \cdot \begin{bmatrix} C^{(u,v)}(1) \\ C^{(u,v)}(2) \\ \vdots \\ C^{(u,v)}(N_1N_2/64) \end{bmatrix} \qquad (7)$$

Here, **A** is a public pseudo-random orthogonal matrix with a size of $N1N2/64$ and it can be generated by orthogonalizing a pseudo-random matrix. By using the orthogonal transform, the reconstruction error will be uniformly scattered over all the DCT coefficients in a same sub-band, which leads to a reconstruction result with better visual quality. For each sub-band, the channel provider selects a positive real number $\Delta(u, v)$ and a positive integer $M(u, v)$, then calculates the round operation returning to the nearest integer and the mod operation gets a remainder.

The vector [$D(u, v)(1)$, $D(u, v)(2)$, …, $D(u, v)(N1N2/64)$] is converted into a sequence of integers whose values fall into [0, $M(u, v)1$], which can be represented as a binary sequence with an approximate length $N\log/642M(u, v)$. At last, the channel provider collects a down sampling sub-image of the encrypted version,

$$cD(i,j) = c(8i, 8j) \qquad (8)$$

And also collects binary sequences of [$Q(u, v)(1)$, $Q(u, v)(2)$, …, $Q(u, v)(N1N2/64)$] T, the values of $\Delta(u, v)$ and $M(u, v)$, and the second part of auxiliary information to form the compressed encrypted data.

EXTRACTION AND IMAGE RECONSTRUCTION

The decryption and the decoding of the bit stream can be achieved by reversing the encryption and encoding processes. The decryption process needs the same key used while encrypting the bit stream. The encrypted bit stream of the text can be perfectly reconstructed to its original image by decrypting the bit stream followed by decoding the bit stream. An approximate image of the original image can be constructed if the decryption and the decoding process are carried out without the extraction of the secret text [8].

At receiver side with the compressed data and the secret key, a receiver can perform the following steps to reconstruct the principal image content.

1. Decompose the compressed data to get the encrypted sub-image, the values of quantization parameters and the second part of auxiliary information.

2. Obtain the interpolated image from down sampling sub image using the bilinear interpolation method. Also, retrieve the values of *binary map* from the second part of auxiliary information.
3. The receiver will modify the estimated encrypted image according to $Q(u, v)(t)$ after performing 2D DCT in a block-by-block manner.
4. At the same time bits are extracted from compressed image and secrete message is recovered.

PERFORMANCE ANALYSIS

Performance analysis demonstrates the simulation results of proposed system to compare the traditional method. These parameters are compression ratio, Mean Square Error, Peak Signal to Noise Ratio, Correlation coefficient.

COMPRESSION RATIO

The compression ratio R is a ratio between the amounts of the compressed data and the original image data is calculated as,

$$R = \frac{\frac{N}{8} + \frac{N}{64} \cdot \sum_{u,v} \log_2 M^{(u,v)} + L_S}{8N} = \frac{1}{64} + \frac{1}{512} \cdot \sum_{u,v} \log_2 M^{(u,v)} + \frac{L_S}{8N} \quad (9)$$

Where L_S is the length of the second part of auxiliary information and we ignore the data amount of $\Delta(u, v)$ and $M(u, v)$ as it is very small.

PSNR

The phrase peak signal-to-noise ratio, often abbreviated PSNR, is an engineering terminology that defines the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the representation of the signal. The PSNR computes the peak signal-to-noise ratio, in decibels, between two images. This ratio is often used as a quality measurement between the original and a compressed image [5].
The PSNR is most often used as an important parameter to calibrate the quality of reconstruction of stegano graphic images. The signal in this case is the original image, and the noise is the error introduced by some steganography algorithm. It is most easily defined via the mean squared error. To compute the PSNR, the block first calculates the mean-squared error.

$$PSNR = 10 \log_{10} \left(\frac{R^2}{MSE} \right) \quad (10)$$

R is the maximum fluctuation in the input image data type. For example, if the input image has a double-precision floating-point data type, then R is 1. If it has an 8-bit unsigned integer data type, R is 255. Higher the value of PSNR, better the quality of the compressed or reconstructed image.

MSE

The MSE represents the cumulative squared error between the compressed and the original image, where as PSNR represents a measure of the peak error [5].

$$MSE = \frac{\sum_{M,N} [I_1(m,n) - I_2(m,n)]^2}{M * N} \quad (11)$$

Lower the value of MSE, lower the error.

CORRELATION COEFFICIENT

Correlation coefficient = corr2 (A, B). It represents correlation coefficient between A and B, where A and B are matrices or vectors of same size. Correlation is scalar double.

EXPERIMENTAL RESULTS

Different Test images were used for experiment. Here we have the results for image fabric sized 648×480

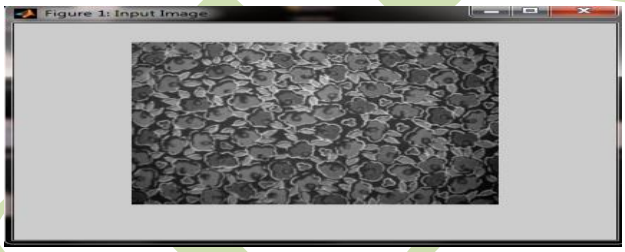

Figure 4: Input image converted to gray scale image

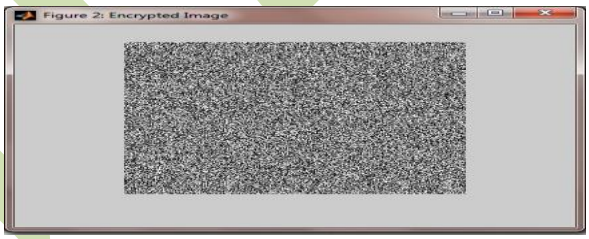

Figure 5: Encrypted image

Figure 6: Binary map

Figure 7: Decompressed image

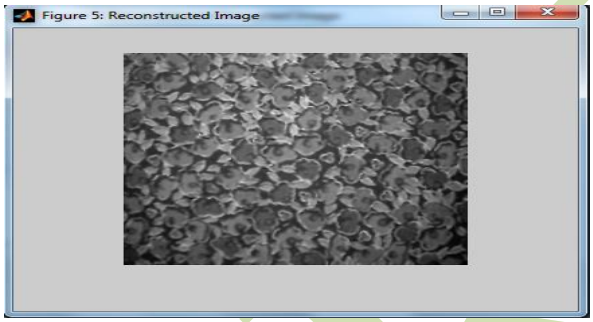

Figure 8: Reconstructed image

When producing an encrypted version, we also generated the auxiliary information. As mentioned above, the first part of auxiliary information includes the values of $\sigma(u, v)$ of 64 sub-band.

Auxiliary Information part-1:

198.5085	105.0244	78.8505	48.2548	37.4472	25.7273	15.7100	9.1573
96.1413	67.8836	56.0157	42.6166	30.9365	22.2386	14.0875	8.5646
70.2487	52.0366	43.5683	35.3668	26.3988	18.5022	12.3320	7.4729
42.5013	39.0498	32.7934	28.2068	21.4112	16.2257	10.2621	6.4323
32.1265	27.1046	24.4853	21.2376	16.6004	12.4129	8.0970	5.0701
21.1359	18.8602	17.8151	15.0306	11.8813	8.4606	5.9533	4.0121
13.8940	13.1222	12.0033	9.9885	8.1955	6.3615	4.2054	2.9110
7.6983	8.0074	7.9611	6.8242	5.4745	4.2310	2.9824	1.9591

Table 1: Table of results

Compression ratio	0.3154
Mean Square Error	6.8458
Peak Signal to Noise Ratio	39.7766
Correlation Coefficient	0.9390

Secret message : Amruta
Recovered message: Amruta

CONCLUSION

This paper proposes a scheme of compressing encrypted images with auxiliary information. Compared with previous methods, the compression performance is improved and the computational complexity is significantly reduced. On the other hand, the proposed compression approach is well-matched with the modulo-256 addition encryption, but is not suitable for other encryption methods, such as standard stream cipher or AES/DES.
In this paper we propose a novel technique to hide a data in the compressed encrypted images using LSB method. The algorithm is simple to implement. Our scheme also preserves the confidentiality of data as the embedding is done on encrypted data.

REFERENCES

1) Zhenxing Qian, and Guorui Xinpeng Zhang, Yanli Ren, Liquan Shen Feng, *" Compressing Encrypted Images with Auxiliary Information" IEE E Transactions on Multimedia, vol.16, page no.1327-1336, 2014.*

2) T Morkel, JHP Eloff and MS Olivier, "*An Overview of Image Steganography," in Proceedings of the Fifth Annual Information Security South Africa Conference (ISSA2005), Sandton, South Africa, June/July 2005.*

3) Johnson, N.F. & Jajodia, S., *"Exploring Steganography: Seeing the Unseen", Computer Journal, February 1998.*

4) H. Faheem Ahmed and U. Rizwan "*A Comparative Study On Some New Steganographic Techniques," International Journal of Advanced Research in Computer Engineering & Technology, vol. 2, Issue2 February2013.*

5) Z. Erkin, A. Piva, S. Katzenbeisser, R. L. Lagendijk, J. Shokrollahi, G. Neven, and M. Barni, *"Protection and Retrieval of Encrypted Multimedia Content: When Cryptography Meets Signal Processing," EURASIP Journal on Information Security, pp. 1 –20, 2007*

6) N. S. Kulkarni, B. Raman, and I. Gupta, "*Multimedia Encryption: A Brief Overview," Recent Advances in Multimedia Signal Processing and Communication. , SCI 231pp. 417–449, 2009.*

7) W. Liu, W. Zeng, L. Dong, and Q. Yao, *"Efficient Compression of Encrypted gray scale Images," IEEE Trans. Signal Processing, 19(4), pp. 1097–1102, 2010.*

8) X. Zhang, "*Lossy Compression and Iterative Reconstruction for Encrypted Image," IEEE Transaction on Information Forensics & Security, 6(1), pp. 53 – 58, 2011.*