A New Approach for Information Security using Asymmetric Encryption and Watermarking Technique

Manish Gupta
Research Scholar, Rajasthan Technical University, Kota

Rajeev Gupta, PhD.
RTU,Kota

Girish Parmar, PhD.
MIT, Kota

Darpan Anand
Hindustan Institute of Technology and Management, Agra.

ABSTRACT
Rapid growth of internet and networked information systems has contributed a lot in the increase of multimedia contents. As digital contents are increasing day by day on internet it has raised requirement for the effective protection of multimedia assets (images, audios, videos). This paper proposes a new approach to protect the multimedia contents using image watermarking, asymmetric encryption and dictionary based compression. Approach presented in this paper hides target image in the host image using image watermarking and then apply RSA algorithm for protecting watermarked image from tempering and subsequently applies dictionary based compression approach to reduce size of encrypted watermarked image.

General Terms and Key Words
Image Watermarking, Cryptography, Encryption, Decryption, Image encryption, RSA, Dictionary Compression

1. INTRODUCTION
Rapid growth of Internet has contributed a lot in the increase of multimedia contents over internet including digital images and videos. Due to exponential rise in multimedia contents, it has become a trend to share information using digital images and video and at the same time it has become a challenging issue to protect these contents from tempering in order to avoid misuse and retain confidentiality of data. In order to achieve secure communication and temper proof data communication our proposed algorithm makes use of digital watermarking to hide secret information into a host image, apply asymmetric encryption (RSA algorithm) on watermarked image to prevent release of content attack and then apply dictionary based compression technique for reducing the size of encrypted watermarked image.

2. LITERATURE REVIEW
Image Watermarking is the process of unnoticeably embedding a watermark image into host image. The resulting image is called a watermarked image [1]. The watermark message should establish only bearable amount of alteration to the host image and it should be recoverable after applying signal processing operations on the watermarked image. In general, "Watermarking is a process of embedding a sequence of messages as additional information into a host file". Watermarking techniques are either "visible" or "invisible" [1]. Traceable mark ("visible watermark") of ownership has been around for centuries in the form of stamps, seals, signatures or classical watermarks. However, for known data manipulation techniques the hardly noticeable watermarks are mandatory in most of the applications. There are several other ways to classify watermarking algorithms. They can be classifying as time domain and frequency domain watermarking [2]. Time-domain watermarking algorithms embed watermarks into the host signal in their time domain. Frequency-domain watermarking algorithms embed watermarks in certain transform domain, such as Fourier domain, cosine transform domain, wavelet domain, or cepstrum domain etc. Watermarking algorithms can also be classified as fragile and robust [3]. A fragile watermark will be changed if the host data is modified. On the contrary, watermarks in robust algorithms cannot be removed by common signal processing operations.

A robust watermarking system meets the following requirements [4]

No Distortion of Original image: The data added to a file that represents the watermark should in no way reduce the quality of the host file. This includes addition of any kind of noise.

A watermark should live in the host image in spite of whatever happens to the host image, including all probable signals processing that may occur, and as well as all unfriendly attacks that unauthorized users may try. This condition is referred to as robustness of the watermark.

Efficient Computation: There should be low computational complexity for watermark embedding and extraction.

A watermark must express as much information as possible, which means the watermark data rate should be high.

Non-Referential: Watermarking has the ability, if executed correctly, to make basic verifications involving no query to a database of matching watermarks. For example if a watermark is corrupted, upon extraction it would indicate that the file has been tampered with. Potentially, a watermark could even hide file metadata in the content of the file, therefore allowing complete identification without database query.

Undetectable: A good watermark should not be detectable by those who would want to tamper with it or remove it altogether.

A watermark should, though being un-removable should be imperceptible.

Resistant to Common Distortions: A robust watermarking system will not be affected by common manipulations of an image file, such as filtering, reverberation.

- Resistant to Malicious Distortions: Malicious distortions include any form of bit cropping, encryption, coding or noise addition.

One of the techniques in the domain of the information hiding and watermarking [5, 6, 7] is LSB coding [8, 9]. A normal approach in the case of the image is to embed watermark data by fluctuation of the individual samples of the image having the amplitude resolution of 8 bits per sample.

LSB coding is the simplest way to embed data into other data type. By replacing the least significant bit of each sample by coded binary bits, we can encode a large amount of data in an image. The watermark embeder utilize all accessible host image samples m. The replacement process mapped m [j] to w [j], where w is known as watermarked image sample. The extraction procedure just retrieves the watermark samples by analysis the value of these bits. Therefore, the decoder wants all the samples of the watermarked image that were utilized at some stage in the embedding process. The alteration of the LSBs of the samples used for data hiding established a low power additive white Gaussian noise.

Images are shared over internet and other networks in digitized form. It is frequently correct that a major portion of such information is secreted. Encryption is the technique for shielding the transferred data over the various types of network [10]. There are a variety of encryption techniques to encrypt and decrypt data like image. The image encryption techniques classified as position permutation based [11], value transformation based [12, 13] and visual transformation based. One of the solutions is to use an encryption technique to encrypt the data such as the Data Encryption Standard (DES), AES, which is a symmetric encryption techniques and the RSA algorithm which is an asymmetric encryption technique [14, 15, 16].
Data Compression is requirement, entail perceptive the way information is structured and, if possible, the technique by which the data was produced. Data compression is of various types like lossy compression and lossless compression [17,18]. In lossy compression technique if we do, then data may loss for some extent that the information may not lose but in lossless compression the data is not lose at any position? In lossless compression, one of technique is dictionary coding. In dictionary coding, the word which frequency is high reduces.

We have a number of techniques to secure data as mentioned in literature, there are some benefits in watermarking and some in symmetric key encryption but we need to apply both of these, to develop a techniques which provide high data payload, robustness and security.

3. PROPOSED ALGORITHM

In order to achieve protection of digital images from tempering and secure message transfer this algorithm has been divided into three steps:

a) Step-1 : Insert Watermark into host image
b) Step-2 : Encrypt watermarked image
c) Step-3 : Compress the Encrypted data

Reverse process can be applied on the receiving end to retrieve actual data sent.

a) Decompress the Encrypted data.
b) Decrypt the decompressed output
c) Extract the message from decrypted file

Figure 1 illustrates block diagram of the proposed algorithm.

Step-1: Insert Watermark into host image
In the first step, secret information available in watermark image is to be hide in host image. For hiding this information, n least significant bits (LSB) of host image are replaced by 0. Value of n may vary from 1 to 8 and can be chosen during implementation. Now, we right shift pixels of watermark image by k bits and add host image and watermark image pixels.

host[i] = LSB(host[i] , n) = 0 $\quad 1 \leq n \leq 3$

By k-bits
watermark [j] <<< watermark[j] right shift watermark array by k positions
k = 8 – n;

Watermarked-Image [i]= host[i] + watermark[j].

Let us define output image as Watermarked-Image

Step-2 : Encrypt Watermarked Image
In second step, we perform asymmetric encryption on watermarked image to avoid release of contents. Pixels values of watermarked image are read in integer format i.e. the value may range between -127 to 128. Depending on the value obtained we define strings based on length of the value. For example if a pixels has value=2, string length for this pixel will be 1, if a pixel has value 128, length of the pixel will be 3. In order to make uniform string of length 4, for each pixel, padding is done by special character in the order defined in Table-1.

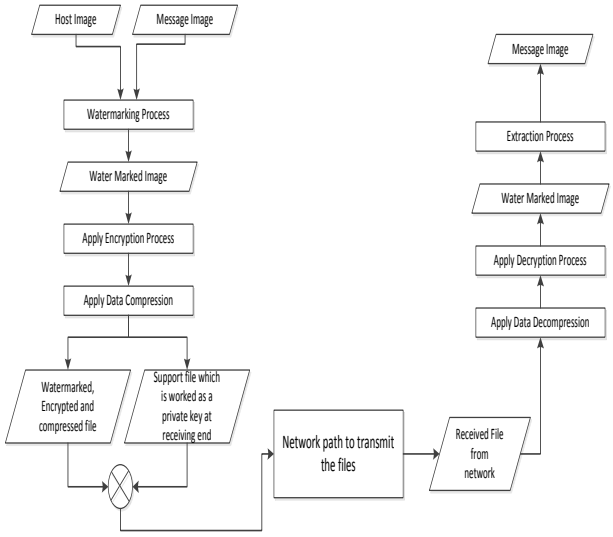

Fig. 1. Block diagram of a proposed approach.

Table 1. List Character to uniform the block length

Length of Pixel Value Obtained	Special Character used for Padding
1	`
2	~
3	#
4	No Padding

Now, we have converted gray level pixel values in string format where each value is a length-4 string. We combine the 24 pixels values at a time and make the block of 96 characters, in order to achieve good results from RSA. Now we apply the encryption on this string using RSA algorithm and change it to cipher string. After obtaining cipher text, we extract each character of this cipher string and cast its value in integer, add all integer values of each block and write in a file. In this way we get two files one is key image and other is the actual information. Figure-2 shows encryption process in detail.

Step-3 : Compress Encrypted data

In step-2 we have obtained sum of 24 pixels values after applying the encryption. Now we cast this integer value into byte and this byte is treated as key of this string. This key is appended to the cipher string at the last position and also put into an array, and process execute continues up to last pixel of image. After all pixels are processed, array of key is converted into an image (A Key Image). The encryption strings after key padding are concatenated. The whole string is passed through the compression process. Compression is zip compression technique which is one of the dictionary based approach.

Finally we have a compressed file and an image as a key for this file. We can transmit above two files and at the receiving end reverse process will be performed.

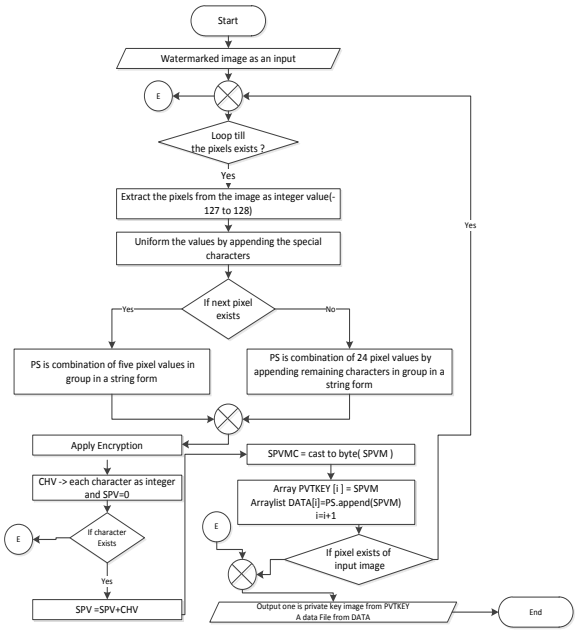

Figure 2: Flow Chart of Encryption process.

4. RESULTS

We implemented this algorithm using MATLAB-7.9.0 and JAVA 1.6 tools. The simulation parameters are as follows;

Table 2. Simulation Parameters

Image Size (M*N)	512*345, 1024*690 and 1536*1035
Number of Bits Changed	N=1,2,4,and 6
Image type	24 bit Bitmap Image(bmp)
Encryption	32 bit
Encryption Algorithms	RSA
Compression	Dictionary based zip
Simulation tools	MATLAB 7.9.0 & JAVA 1.6

We have simulated the process of mentioned methodology for the different value of replacement bit in host image by the bit in message image i.e. .n and values of n for testing are 1, 2, 4 and 6.

The Process is divided in two ends as discussed in methodology, illustrate as follows:-

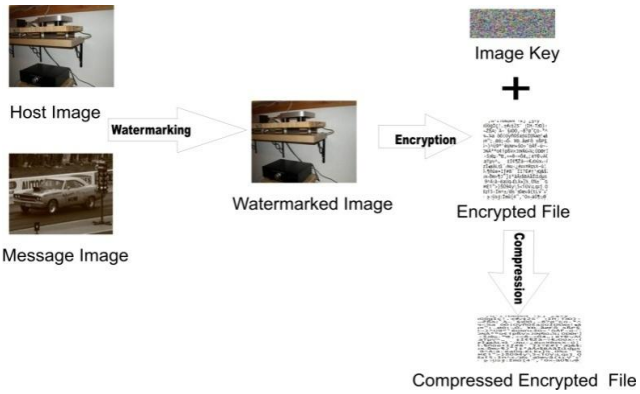

Figure 3: Sender's end process

We extract the message at receiving after successful completion of this algorithm.

We test the algorithm on the following seven performance parameters:-

(1) **Watermarking Time (S):** Total time taken for watermarking.
(2) **Peak Signal to Noise Ratio (PSNR):** It's a comparison between original message image and extracted image after decompression, decryption and extraction.
(3) **Encryption Time (S)** :
Total time taken for Encryption.
(4) **Compression ratio (%) :**
Compressed size / Uncompressed Size
(5) Space Saving Ration (%):
Compressed size / Uncompressed Size
(6) **Compression Time (%) :**
Total time taken for Compression

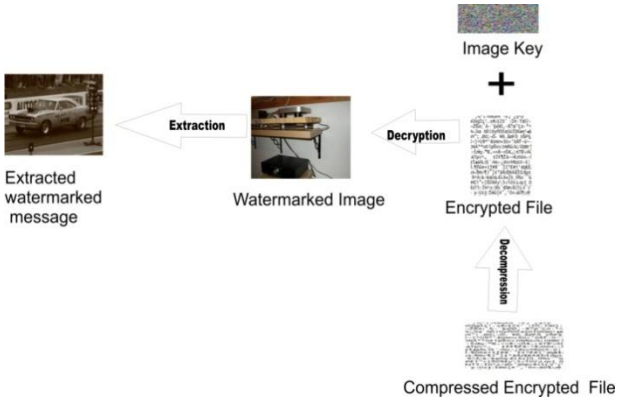

Figure 4: Receiver's end process

The tested results are illustrated in table 3.

Table 3: Performance Parameters for proposed Algorithm

Sample Number	Uncompressed File Size (KB)	Compressed File Size			Compressed Size (KB)	Performance Parameters					
		Support File	Image Key	Total		Watermarking Time (S)	PSNR	Encryption Time (S)	Compression Ratio (%)	Space saving (%)	Compression Time (S)
1	518	125	14	139	379	0.53	51.21	7.73	73.17	26.83	0.14
2	518	125	14	139	379	0.56	43.99	10.335	73.17	26.83	0.13
3	518	125	14	139	379	0.5	31.84	8.29	73.17	26.83	0.12
4	518	125	14	139	379	0.46	31.87	7.85	73.17	26.83	0.12
5	518	125	14	139	379	0.56	31.85	8.53	73.17	26.83	0.15
6	518	125	14	139	379	0.5	20.01	7.6	73.17	26.83	0.23
7	2071	495	78	573	1498	1.07	31.85	33.355	72.33	27.67	0.61
8	4658	1112	122	1234	3424	1.51	31.85	63.315	73.51	26.49	1.85

The graphical representation of different performance parameters with respect to the size of original image as mentioned in table 3 are plotted in fig 5.

5. DISCUSSION

Since we had apply the novel encryption and compression technique after the existing watermarking technique i.e. Least Significant Bit replacing method, therefore performance parameter which is shown in table 3 and figure 5 like space saving and security will be increases while at the same time complexity and the total time of process will be increases.

Currently we are focusing on the security and the space saving, i.e. why we can ignore the increment in time complexity.

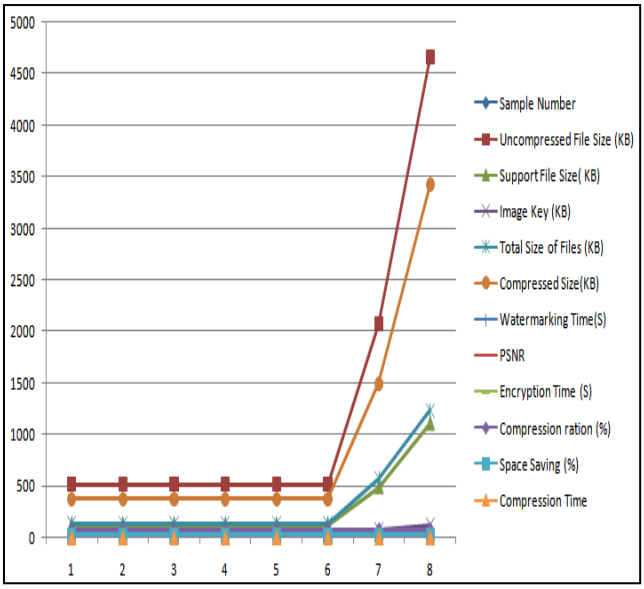

Figure 5: Graph of Performance parameters of Proposed Algorithm

The figure 5 illustrates the graphical representation of different parameters described in Table 3. In x axes, the number denotes the sample # and the variations of parameters are represented with distinct colors.

6. CONCLUSION

The LSB alteration method gives an easy means to insert message in images, but the data can be easily deciphered. The proposed method used in this paper encrypts and compress the secret color image information after embedding it in the color image. Surely the complexity of the process increases but at the same time the safety achieved as well as reduces the size by 73%, at this cost is well worth it. This cryptography which is combination of watermarking, encryption and compression can be used for other Steganography techniques also.

RSA is already accepted as a secure public key encryption algorithm and watermarking is also technique to provide the information security. Along with this we generate an image key from the encrypted watermarked image, it increases the security. We also did compression to reduce the data to transfer in network to decrease the data load on network, it make fast transfer of information.

Future scope is to test and implement the same algorithm on audio, video and other multimedia contents, the algorithm can be modified as the LSB can be on random bases with respect to the position of the image i.e. either from left or right or top or bottom etc and RSA private key and the random number information and key image can be merged into a single file.

7. REFERENCES

[1] Bender, W., Gruhl, D., Morimoto, N., and Lu, A.,"Techniques for data hiding," IBM Systems Journal, vol. 35, pp. 313-336 (1996).

[2] Kim, H.J. and Choi, Y.H. and Seok, JW and Hong, JW, Audio watermarking techniques, Intelligent Watermarking Techniques, 185—218 (2004)

[3] Borko Furht and Darko Kirovski Auerbach, Multimedia Watermarking Techniques and Applications Pages 425–439 Print ISBN: 978-0-8493-7213-1 eBook ISBN: 978-1-4200-1346-7 DOI: 10.1201/9781420013467.ch14 (2006).

[4] Wu et al. Robust and efficient digital audio watermarking using audio content analysis. Proceedings of IS & T/SPIE 12th International Symposium on Electronic Imaging, (2000).

[5] Fridrich J, Goljan M & Du R Distortion-free data embedding. Lecture Notes in Computer Science 2173: p 27–41 (2001).

[6] Lee Y & Chen L High capacity image steganographic model. IEE Proceedings Vision Image Signal Processing 147(3): p 288–294 (2000).

[7] Fridrich J, Goljan M & Du R Lossless data embedding - new paradigm in digital watermarking. Applied Signal Processing 2002(2): p 185–196 (2002).

[8] Yeh C & Kuo C Digital watermarking through quasi m-arrays. In: Proc. IEEE Workshop on Signal Processing Systems, Taipei, Taiwan, p 456–461 (1999).

[9] Cedric T, Adi R & Mcloughlin I Data concealment in audio using a nonlinear frequency distribution of prbs coded data and frequency-domain lsb insertion. In: Proc. IEEE Region 10 International Conference on Electrical and Electronic Technology, Kuala Lumpur, Malaysia, p 275–278 (2000).

[10] Younes, M.A.M.B., An Approach To Enhance Image Encryption Using Block-Based Transformation Algorithm, University Sains Malaysia (2009).

[11] Jiun-In Guo, Jui-Cheng Yen, "A new mirror-like image Encryption algorithm and its VLSI architecture", Pattern Recognition and Image Analysis, vol.IO, no.2, pp.236-247, (2000).

[12] Aloha Sinha, Kehar Singh, "A technique for image encryption using digital signature", Optics Communications, Vol-2 I 8 229-234, (2003).

[13] S.S.Maniccam, N.G. Bourbakis, "Lossless image compression and encryption using SCAN", Pattern Recognition 34,1229- 1245 (2001).

[14] J. Daemen and V. Rijmen, AES Proposal: Rijndael , AES Algorithm Submission, (1999)

[15] R. L. Rivest, M. J. B. Robshaw, R. Sidney, and Y. L. Yin, The RC6TM Block Cipher , M. I. T laboratory for Computer Science, 545 Technology Square, Cambridge, MA 02139, USA, (1998).

[16] R. F. Sewell, Bulk Encryption Algorithm for Use with RSA , Electronics Letters ,Vol. 29, No. 25, pp. 2183-2185, (1993).

[17] Popat, K. and Picard, R.W. and Massachusetts Institute of Technology. Media Laboratory. Vision and Modeling Group, Novel cluster-based probability model for texture synthesis, classification, and compression, Citeseer, (1993).

[18] Taneja, N. and Bhatnagar, G. and Raman, B. and Gupta, I., Joint watermarking and encryption for still visual data, Multimedia Tools and Applications, Springer, ,1—14 (2012).