International Journal of Innovative Research in Information Security (IJIRIS) ISSN: 2349-7017(O)
Issue 2, Volume 4 (April 2015) ISSN: 2349-7009(P)
www.ijiris.com

Image Encryption-Then-Compression System via Prediction Error Clustering and lossless encoding

SHREEDHAR BM*
Electronics & Communication Engg
VTU

VISHALA IL
Electronics & Communication Engg
VTU

HEMAVATHI N
Electronics &Communication Engg
VTU

Abstract— The images have to be encrypted before compression to give high level security. We design a highly efficient image encryption-the –compression system using lossless and lossy compression. The image encryption design to operate in the prediction error domain to provide in a sensible way to give high level security. An arithmetic coding based approach can be exploited to efficiently compress the encrypted images. The proposed compression approach applied to encrypted images is only slightly worse, in terms of compression efficiency than the disambiguation lossless/lossy image coders which take original unencrypted images as inputs. In existing Encryption then compression induced the significant penalty on compression efficiency. The password generation and encryption are all done by the system itself after clicking the encryption button with transparency to the user. The same encryption password is also used to decrypt the encrypted binary file. The application uses simple key generation methods of random number generation and combination.

Keywords— Compression of encrypted image, gradient based prediction and Arithmetic code, encrypted domain signal processing

I. INTRODUCTION

Encryption is the process of encoding message or information in such a way only authorized persons can see it. Encryption does not itself prevent interception, but denies the message content to the interceptor. Decryption is the process of decoding encrypted information it can be accessed again by authorized users. Decryption is generally the reverse of encryption. It is the process of decoding the data which has been encrypted into a secret format. Encryption is the most effective way to achieve data security. We want read encrypted file, we must have access to a secret password that enables we to decrypt it. Encryption is the conversion of data into a form, called a cipher text. Decryption is the process of converting encrypted data back into its original form. Unencrypted data is called plaintext, Encrypted data is refers to as cipher text.
The encryption-decryption cycle shown in fig1.

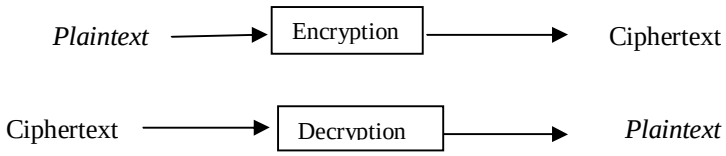


Fig1: Encryption-Decryption cycle

Compression is the reduction in size of data in order to save space and transmission time. When a file is compressed, it takes up less disk space than an uncompressed version and can transferred to other system more quickly. For data transmission, compression can be performed on just the data content or on the entire transmission unit depending on a number of factors. Arithmetic coding is a form of entropy encoding used in lossless data compression. Arithmetic coding differs from other forms of entropy encoding, such as Huffman coding. Arithmetic coding is arguably the most optimal entropy coding technique if the objective is the best compression ration since it usually achieves better result than huffanman coding. Arithmetic coding is a data compression technique that encodes data by creating a code string which represents a fractional value on number line between 0 and 1. Lossless data compression is a class of data compression algorithm that allows the original data to be perfectly reconstructed from the compression data. Lossless data compression is used in much application. Lossy compression is a class of data encoding that uses an approximation for representing the content that has been encoded. Lossy compression is most commonly used to compress multimedia data (audio, video, and still images), especially in application such as streaming media and internet telephony. The advantages of lossy methods over lossless methods is that in some cases a lossy method can produce a much smaller compressed file than any lossless method, while still meeting the requirements of the application. Data decompression is a reverse operation of data compression.

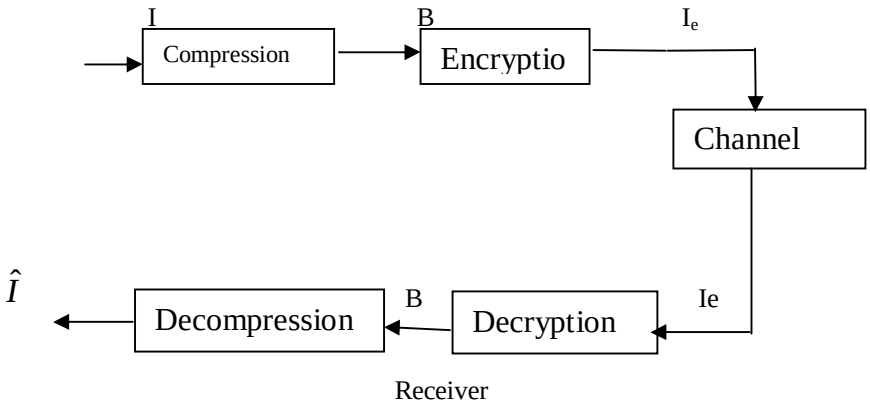


Fig.2: (a) Traditional Compression-then-Encryption (CTE) system

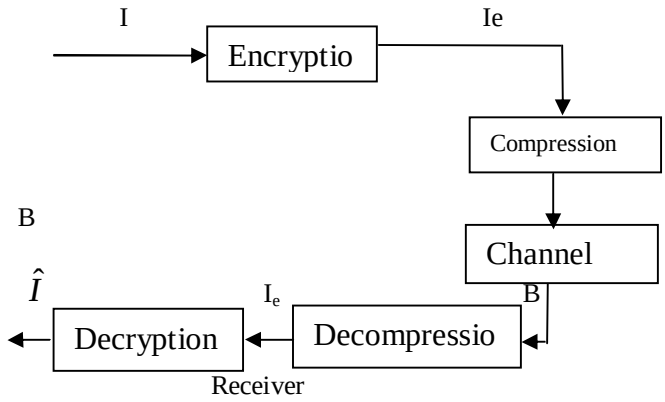


Fig.2 :(b) Encryption-then-Compression (ETC) system.

In Figure2 (a) shows the compression-then-Encryption (CTE) system. Transmitter wants to securely and efficiently transmit an image I to a receiver, via an untrusted channel provider. Transmitter first compress I into B and then encrypts B into I_e using an encryption function EK(.),where K denotes the secret key, as shown in fig1.The encrypted data I_e is then passed to channel, channel simply forwards it to receiver, upon receiving I_e receiver sequentially performs decryption and decompression to get a reconstruction image $\hat{I}$.

The above compression-then-Encryption(CTE) paradigm meets the requirements in many secure transmission scenarios, in order to apply the compression and encryption needs to be reversed in some other situations. A transmitter is always interested in protecting the privacy of the image data through encryption. Nevertheless, transmitter has no incentive to compress data, and hence will not use available limited computational resource to run a compression algorithm before encrypting the data.In Figure2(b) shows the Encryption-then-Compression (ETC),a data can transmitter wants to securely and efficiently transmit an image I to a receiver, via an untrusted channel provider. . Transmitter first encryption I into I_e and then encrypts B into I_e using an encryption function EK (.), where K denotes the secret key, as shown in fig2. The encrypted data I_e is compressed then passed to channel, simply forwards it to receiver, upon receiving I_e receiver sequentially performs decompression and decryption to get a reconstruction image $\hat{I}$.

II. REVIEWS ON RELATED RESEARCH

J. Zhou, X. Liu, and O. C. Au, “On the design of an efficient encryption then-Compression system,” [1] Image encryption has to be conducted prior to image compression. In this paper how to design a pair of image encryption and compression algorithms such that compressing encrypted images can still be efficiently performed is explained. This paper introduced a highly efficient image encryption-then compression (ETC) system.

The proposed image encryption scheme operated in the prediction error domain is able to provide a reasonably high level of security. More notably, the proposed compression approach applied to encrypted images is only slightly worse, unencrypted images as inputs. In contrast, most of the existing ETC solutions induce significant penalty on the compression efficiency.

Bianchi, A. Piva, and M. Barni, "On the implementation of the discrete Fourier transform in the encrypted domain[2]," Signal processing modules working directly on encrypted data provide an elegant solution to application scenarios where valuable signals must be protected from a malicious processing device. In this paper, we investigate the implementation of the discrete Fourier transform (DFT) in the encrypted domain, by using the homomorphism properties of the underlying cryptosystem. Several important issues are considered for the direct DFT, the radix-2, and the radix-4 fast Fourier algorithms, including the error analysis and the maximum size of the sequence that can be transformed. We also provide computational complexity analyses and comparisons. The results show that the radix-4 FFT is best suited for an encrypted domain implementation in the proposed scenariosM. Johnson, P. Ishwar, V. M. Prabhakaran, D. Schonberg, and K. Ramchandran [3], "On compressing encrypted data," In this report, we will investigate the consequences of reversing this order by first encrypting and then compressing, as shown in Fig 2. Compressor does not have access to the secret key. At reset glance, it appears that not much gain can be obtained, because encrypted data looks quiet random. But we have a joint decompression and decryption at the receiver. So, decoder has access to the key. Turns out, that significant compression gain can be obtained, using the techniques from distributed source coding theory. In some cases, one can obtain gains similar to the traditional system, where encryption follows compression. In this subsection, we will look at the case when Y is to be compresses losslessly.

This is possible only when Y and K are discrete. Riccardo Lazzeretti and Mauro Barni"Lossless compression of encryption Grey level and color image" [4]. In this paper lossless compression of encrypted images relying on the analogy with source coding with side information at the decoder. It works only addressed the compression of bi-level images ,namely spares black and white images, with asymmetric probability of compressing encrypted grey level and color images, by decomposing them into bit plane. In multimedia contents need both to be compressed and protected. Where the classical way to compress and protect data requires that data are first compressed end then encrypted.X.Zhang, G.Feng, Y.Ren, and Z.Qian, "Scalable coding of encrypted images" [5], in this paper proposes a novel scheme of scalable coding for encrypted images. In encryption phase, the original pixel values are masked by a modulo-256 addition with pseudorandom number that are derived from a secret password. After decomposing the encrypted data into a down sampled sub image. The data of quantized sub image and coefficient are regarded as a set of bit streams.

At the receiver side while a sub image is decrypted to provide the rough information of the original content, the quantized coefficient can be used to reconstructed. Because of the hierarchical coding mechanism, the principal original content with high resolution can be reconstructed when more bit streams are received.A. kumar and A.Makur "Distubtion source coding based encryption and lossless compression of gray scale and color images"[6], in this paper proposed to the approach of the prediction error domain and achieved better lossless compression performance on the encrypted grayscale/color image. To achieve high compression rations, lossy compression of encrypted data also studied.

III. PROPOSED ETC SYSTEM

In this section, we present the details of the two key components in our proposed ETC system, namely, image encryption conducted by transmitter, image compression conducted by Channel,

A. Image Encryption via Prediction Error Clustering and Random Permutation

Encryption refers to set of algorithms, which are used to Convert the plain text to code or the unreadable form of text, and provides privacy. To decrypt the text the receiver uses the "Secret key" for the encrypted text. The feasibility of lossless compression of encrypted images has been recently demonstrated by relying on the analogy with source coding with side information at the decoder. Upon receiving the compressed and encrypted bit stream B, a multimedia technology for information hiding which provides the authentication and copyright protection.

International Journal of Innovative Research in Information Security (IJIRIS) ISSN: 2349-7017(O)
Issue 2, Volume 4 (April 2015) ISSN: 2349-7009(P)
www.ijiris.com

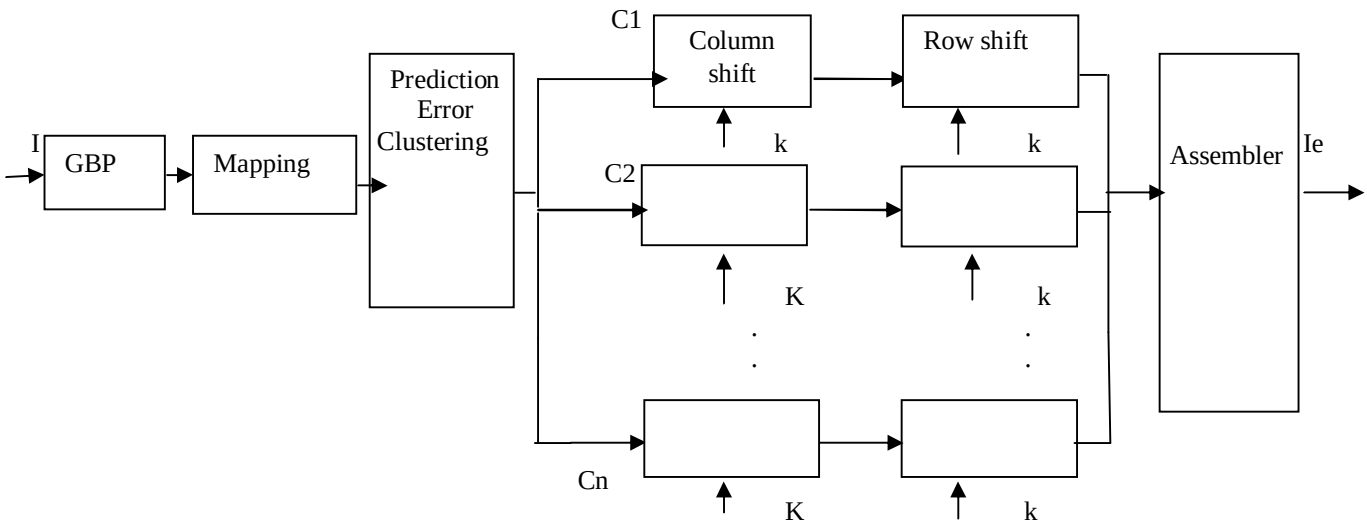


Fig:3: Schematic diagram of image encryption

The algorithmic procedure of performing the image encryption is then given as follows:

Step 1: compute all the mapped prediction errors $\tilde{e}_{ij}$ of the whole image I.
Step 2: Divide all the prediction errors into L clusters c_k, for $0 \le k \le L-1$, where k is determined by (5), and each Ck is formed by concatenating the mapped prediction errors in a raster-scan order.
Step 3: Reshape the prediction errors in each Ck into a two D block having four columns and $\lceil |Ck|/4 \rceil$ rows, where $|c_k|$ denotes the number of prediction errors in c_k.
Step 4: Perform two key-driven cyclical shift operations to each resulting prediction error blocks, and read out the data in raster-scan order to obtain the permuted cluster $\tilde{c}_k$.

Let CS_k and RS_k be the secret key vectors controlling the column and the row shift offsets for Ck . Here, CS_k and RS_k are obtained from the key stream generated by a stream cipher, which implies that the employed key vectors could be different, even for the same image encrypted at different sessions.

The random permutation is also illustrated in Fig. 3 for an input sequence S = s1 s2 . . . s16 , where the numbers within the blocks denote the indexes of the elements of S. Before permutation, the first row becomes (1, 2, 3, 4), the second row becomes (5, 6, 7, 8), etc. The column shifts are specified by a key vector CS = [3 1 0 2], with each column undergoing a downward cyclical shift in accordance with the key value associated with that column. The procedure is then repeated using another key vector RS = [1 2 31] for each of the rows. Note that such permutation operations can be realized via circular shifts, which are easily implemented in either hardware or software

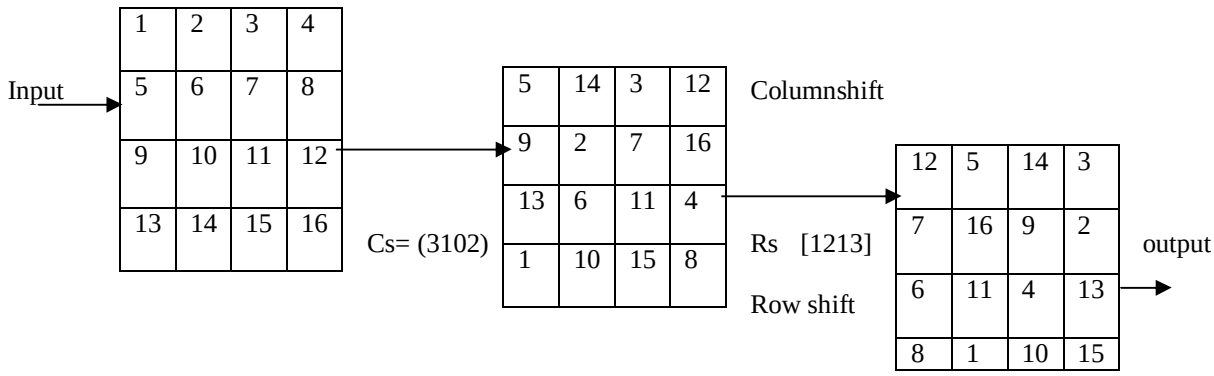


Fig.4. An example of the cyclical shifts

Step 5: The assembler concatenates all the permuted Clusters c_k, for $0 \le k \le L-1$, and generates the final encrypted image

$$I_e = C_0 C_1 .... C_{L-1}$$

in which each prediction error is represented by 8 bits. As the number of prediction errors equals that of the pixels, the file size before and after the encryption preserves.

Step6: Pass Ie to Charlie, together with the length of each cluster $|\widetilde{ck}|$, for $0 \le k \le L-2$.The values of $|\widetilde{ck}|$ enable channel to divide Ie into L clusters correctly. In comparison with the file size of the encrypted data, the overhead induced by sending the length $|\widetilde{ck}|$ is negligible.

B. Compression of Encrypted Image via Arithmetic Coding

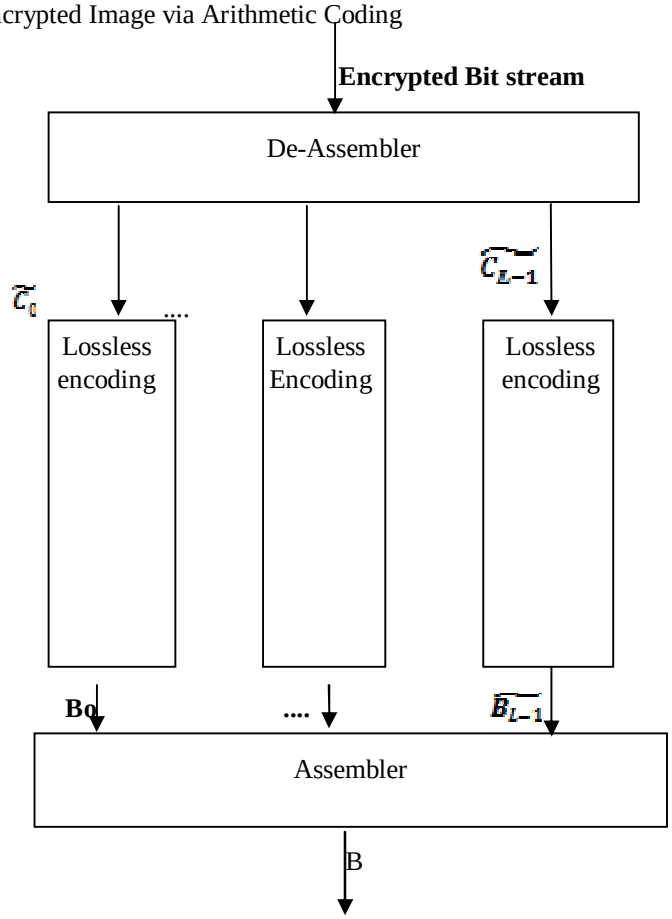


Fig. 5: Schematic diagram of compressing the encrypted data.

C. Lossless compression of Encrypted image Via Adaptive Arithmetic coding

Arithmetic coding is a form of entropy encoding used in lossless data compression. Arithmetic coding differs from other forms of entropy encoding, such as Huffman coding. Arithmetic coding is arguably the most optimal entropy coding technique, if the objective is the best compression ration since it usually achieves better result than huffanman coding. Arithmetic coding is a data compression technique that encodes data by creating a code string which represents a fractional value on number line between 0 and 1.The Arithmetic coding is especially suitable for small alphabet(binary sources) with highly skewed probabilities. Arithmetic coding is very popular in the image and video compression applications. Consider a half open interval [low, high], initially, interval is set as [0, 1) and range=high -low = 1-0 = 1.Interval is divided into cumulative probabilities of n symbols. For this example=3; p (a) =1/2, p (b) =1/4 and p(c) =1/4.We propose an image encryption scheme operated over the prediction and permutation based image encryption method and the efficiency of compressing the encrypted data. Compression of the encrypted file Ie needs to be performed in the encrypted domain, as Channel does not have access to the secret key K. In Fig. 4, we show the diagram of compression of I_e. Assisted by the side information$|\widetilde{c_k}|$, for$0 \le k \le L-2$, a de-assembler can be utilized to parse I_e into L segments$\widetilde{c_1}$, $\widetilde{c_2}$, ・ ・ , $\widetilde{c_{L-1}}$ in the exactly same way as that done at the encryption stage. An AC is then employed to encode each prediction error sequence ˜C k into a binary bit stream Bk. Note that the generation of all Bk can be carried out in a parallel manner to improve the throughput.

An assembler concatenates all Bk to produce the final compressed and encrypted bit stream B, namely, B = B0B1 ・ ・ ・BL−1

IV SOFTWARE REFERENCE MODEL

In this simulink model is used to compare PSNR and MSE scheme.

Image	Chess	
Quality Measures	PSNR	MSE
Original Image	24.88db	286.32
Encrypted Image	25.89db	156.56

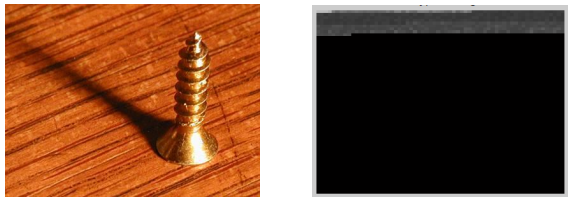
In table 1 PSNR is most commonly used to measure the quality of for image compression. The signal in this case is the original data, and the noise is the error introduced by compression. When comparing compression, PSNR is a human perception of reconstruction quality. The PSNR is calculated based on color texture based image segmentation. The PSNR range between [0, 1), the higher is better. PSNR = 20 * log10 (255 / sqrt (MSE)).Mean Square Error (MSE) is calculated pixel-by pixel by adding up the squared difference of all the pixels and dividing by the total pixel count. MSE of the segmented image can be calculated by using the Equation. The MSE range between [0, 1], the lower is better.

(a) (b)

Fig. 6 Sample Results of (a) original; (b) encrypted..

IV. RESULTS

In this section, the security of our proposed image encryption and the compression performance on the encrypted data are evaluated experimentally. Fig. 5 illustrates the Lena and Baboon images, together with their encrypted versions, from which we can see that our encryption approach is effective in destroying the semantic meaning of the images. In addition, it can be observed that the encrypted Baboon image looks ‘brighter’ than the encrypted Lena image. This is because the Baboon image contains large portion of texture regions that are difficult to compress, resulting in more large-valued prediction errors. We implement the attack strategy of directly decoding the encrypted file Ie, as described in Section IV-A. Ten images of size 512 × 512 are used as the test set. In Fig. 6; we give the PSNR results of the reconstructed images, where x-axis represents the image ID. It can be observed that all the PSNR values are around 10 dB, which is too low to convey any useful semantic information. We also evaluate the reconstruction performance under the assumption that the bounded errors only occur in the prediction errors of the first two rows, while all the remaining ones are perfectly known. Here the estimation error bound ε is set to be 5. Fig. 7 illustrates the PSNR values of the reconstructed images, where each point is the averaged result of 10 realizations. It can be seen that, even under such favorable conditions, the attacker still cannot obtain any useful visual information of the source images, because all the PSNR values are too low (around 10 dB).



(a) (b)

(a) Original Image (b) Encrypted Image

Fig. 7 Result

VI. CONCLUSIONS

In this paper, we have designed an efficient image Encryption then Compression (ETC) system. Within the proposed work, the image encryption has been achieved via prediction error clustering and random permutation. Efficient compression of the encrypted data has then been done by arithmetic coding approach. By Arithmetic Coding based, Coding can't be cracked. Both theoretical and experimental results have shown that reasonably high level of security has been retained. The coding efficiency of our proposed compression method on encrypted images is very close to that of the image codec's, which receive original, unencrypted images as inputs. The Compressed image is measured in terms of Quality measures like MSE and PSNR.

ACKNOWLEDGEMENT

We express our sincere thanks to Dr. Ravi Kumar, HOD,, Dept of E&c,Assistant Prof. Vishala I.L Dept of ECE, who have contributed towards the development of the template.

REFERENCES

[1] J. Zhou, X. Liu, and O. C. Au, "On the design of an efficient encryption-then-compression system," in Proc. ICASSP, 2013, pp. 2872–2876.

[2] T. Bianchi, A. Piva, and M. Barni, "On the implementation of the discrete Fourier transform in the encrypted domain," IEEE Trans. Inf.

[3] M. Johnson, P. Ishwar, V. M. Prabhakaran, D. Schonberg, and K. Ramchandran, "On compressing encrypted data," IEEE Trans. Signal Process., vol. 52, no. 10, pp. 2992–3006, Oct. 2004.

[4] R. Lazzeretti and M. Barni, "Lossless compression of encrypted grey- level and color images," in Proc. 16th Eur. Signal Process. Conf., Aug. 2008, pp. 1–5.

[5] X. Zhang, G. Feng, Y. Ren, and Z. Qian, "Scalable coding of encrypted images," IEEE Trans. Imag. Process., vol. 21, no. 6, pp. 3108–3114, Jun. 2012.

[6] A. Kumar and A. Makur, "Distributed source coding based encryption and lossless compression of gray scale and color images," in Proc. MMSP, 2008, pp. 760–764

