Volume 4, Issue 11, November 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Improvising Image Compression System by Random Permutation

Daundkar Anita Mohan, Pratima Bhati
Computer Department &University,
Pune, India

Abstract—In practical scenario encryption should be performed before the image compression. If encryption is not performed then there may be the chances of stealing the information. Therefore we have proposed system where encryption is done prior to the image compression. Here we have considered both the types of compression -Lossless and Lossy compression. Compression is done in the prediction domain where experts find out the prediction error of each pixel then encryption is applied on them by random permutation. Prediction domain provides the high level of security.Most of the existing ETC solutions induce significant penalty on the compression efficiency. In this paper we are proposing a new approach named as "Improvising Image compression System by Random Permutation." where compression is applied on the clusters of encrypted image . So image received at the receiver end has all the characteristics of original image.

Keywords—Image Compression, Encryption, Random Permutation,Clustering.

I. Introduction

In any application if content owner wants to send to securely and efficiently transmit an image I to a recipientvia an untrusted channel provider .Content owner first compresses I into B, and then encrypts B into Ie using an encryption function EK , where K denotes the secret key .The encrypted data Ie is then passed to untrusted channel who simply forwards it to recipient. Upon receiving Ie, receiver sequentially performs decryption and decompression to get a reconstructed image I. Even though the above Compression-then-Encryption (CTE) paradigm meets the requirements in many secure transmission scenarios, the order of applying the compression and encryption needs to be reversed in some other situations. As the content owner is always interested in protecting the privacy of the image data through encryption. Nevertheless, owner has no incentive to compress his data, and hence, will not use his limited computational resources to run a compression algorithm before encrypting the data. In contrast, the channel provider has an overriding interest in compressing all the network traffic so as to maximize the network utilization. It is therefore much desired if the compression task can be delegated by Channel provider, who typically has abundant computational resources.therfore there is a need of encryption prior to image compression.

So here introduces a new method which can be used to compress encrypted image by a random permutation. As the name describes randomly compression algorithm is applied then by a assembler again data get combined and form a compressed image.. Image processing is a technique of processing an input image and to get the output as either improved form of the same image and/or characteristics of the input image. In proposed system we retained original image.

II. RELATED WORK

CALIC- A Context Based Adaptive Lossless image Codec[1] describes the lossless compression by using prediction error method. Where prediction is depends on the best of eight predictors followed by Huffman coding of prediction error.

The LOCO-I Lossless Image Compression Algorithm: Principles and Standardization into JPEG-LS [2] describes lossless compression for continuous tone images. Fixed prediction algorithm is used. Method is very slower.

Lossless Compression of Encrypted Grey-Level and Color Images[3] describes compressing encrypted grey level and color images, by decomposing them into bit-planes.A few approaches to exploit the spatial and cross-plane correlation among pixels are discussed. This system is suitable for lossy compression only. Lossless compression is not possible with this system.

EncryptedDomain DCT based on homomorphic Cryptosystem[4] is one such a encrypted image Discrete cosine Transform (DCT) tool is used to process encrypted data. Homomorphic encryption is a form of encryption which allows specific types of computations to be carried out on cipher text and generate an encrypted result which, when decrypted, matches the result of operations performed on the plaintext. This is a desirable feature in modern communication system architectures. Homomorphic encryption would allow the chaining together of different services without exposing the data to each of those services, for example a chain of different services from different companies could 1) calculate the tax 2) the currency exchange rate 3) shipping, on a transaction without exposing the unencrypted data to each of those services. DCT allows a large no of processing tasks to be carried out on encrypted images like extraction of encrypted data from encrypted image, embedding watermarking in encrypted image etc. Different types of DCT method: 1D DCT, 2D DCT, CD BDCT(block based DCT). DCT performs the operation on image like The disadvantage of this method is Most of the computation time required to transform, quantize, dequantize, and reconstruct an image is spent on forward and inverse DCT calculations. Because these transforms are applied to blocks, the time required is proportional to the size of the imageThese times are much longer than for comparable functions written in a low-level language such as C. Size of the image get increases after decryption.

Composite Signal Representation for Fast and Storage- Efficient Processing of Encrypted Signals[5]analyzed the possibility of reducing the expansion factor required in signal processing encrypted domain. applications based on homomorphic encryption by packing together several signal samples into a unique composite word. provided a general framework extending an idea put forward and derived precise conditions that permit to process the underlying signal by operating directly on the composite words thus achieving a significant gain from a computational complexity perspective. Problem that is left for future research is the development of an efficient protocol that permits to pass from the composite to the sample-wise representation without that the parties involved in the protocol share any secret information. Existing schemes, in fact, are either computationally inefficient or can only be applied to the particular case.

Lossy Compression and Iterative Reconstruction for Encrypted Image[6] describes novel scheme for lossy compression of an encrypted image with flexible compression ratio. A pseudorandom permutation is used to encrypt an original image, and the encrypted data are efficiently compressed by discarding the excessively rough and fine information of coefficients generated from orthogonal transform. After receiving the compressed data, with the aid of spatial correlation in natural image, a receiver can re- construct the principal content of the original image by iteratively updating the values of coefficients. On the other hand, the security of encryption used here is weaker than that of standard stream cipher.

Privacy Preserving ECG Classification with Branching Programs and Neural Networks[7] describes Privacy protection is a crucial problem in many biomedical signal processing applications. For this reason, particular attention has been given to the use of secure multi- party computation techniques for processing biomedical signals, whereby nontrusted parties are able to manipulate the signals al- though they are encrypted. This paper focuses on the development of a privacy preserving automatic diagnosis system whereby a remote server classifies a biomedical signal provided by the client without getting any information about the signal itself and the final result of the classification. systems prove that carrying out complex tasks like ECG classification in the encrypted domain efficiently is indeed possible in the semihonest model, paving the way to interesting future applications wherein privacy of signal owners is protected by applying high security standards. Disadvantages of this paper is complexity is very high.

On Compression of Data Encrypted With Block Ciphers[8]based on Slepian-Wolf coding and hinges on the fact that chaining modes, which are widely used in conjunction with block ciphers, introduce a simple symbol-wise correlation between successive blocks of data. The compression was shown to preserve the security of the encryption scheme. The existence of a fundamental limitation to compressibility of data encrypted with block ciphers when no chaining mode is employed. But this method is theoretically well suited but practically implementation not works properly.

III. METHODOLOGY

A. *ARITHMATIC CODING ALGORITHM:*

One Arithmetic coding, which is a method of generating variable-length codes, is useful when dealing with sources with small alphabets such as binary sources. In order to explain, let us compare arithmetic coding with Huffman coding.

Huffman coding, which is the most famous method for source coding or data compression, guarantees a coding rate R within 1 bit of the entropy H. That is, the Huffman code for a source S with an average codeword length L satisfies

$$H(S) \leq L \leq H(S) + 1 \quad (1)$$

Moreover, if we encode the output of the source in longer blocks of symbols, we are guaranteed the average codeword length per input symbol closer to the entropy. In other words, suppose we encode the sequence by generating a codeword for every n symbols, and then we have

$$H(S) \leq L_n \leq H(S) + \frac{1}{n} \quad (2)$$

where L_n denotes the average codeword length per input symbol.

However, there is still another problem. The latter approach becomes impractical since it causes an exponential growth in the size of the codebook when we try to obtain Huffman codes for long sequences of symbols. In other words, the complexity of this approach increases exponentially with block length.

Arithmetic coding is a method of encoding without this inefficiency. In arithmetic coding, instead of using a sequence of bits to represent a symbol, we represent it by a subinterval of the unit interval $[0, 1]$. In other words, we encode the data into a number in the unit interval $[0, 1]$, and this technique can be implemented by separating the unit interval into several segments according to the number of distinct symbols. The length of each segment is proportional to the probability of each symbol, and then the output data is located in the corresponding segment according to the input symbol.

This provides a way of assigning codewords to particular sequences without having to generate codewords for all sequences and alleviates the inefficiency and the complexity. Moreover, the code for a sequence of symbols is an interval whose length decreases as we add more symbols to the sequence. This property allows us to have a coding scheme that is incremental, that is, the code for an extension to a sequence can be calculated simply from the code for the original sequence.

IV. DESIGN PROCESS

ForProposed research work has three different modules which will be presented here. We will have the four phases like :Encryption of image,Compression,Decryption.Random permutation and clustering is the new methodology used for image encryption and compression. The phases are:

A. IMAGE ENCRYPTION:

The first phase is the image encryption where the image is split into blocks and these blocks are permutated. Further permutation is applied based on a random number to strengthen the encryption

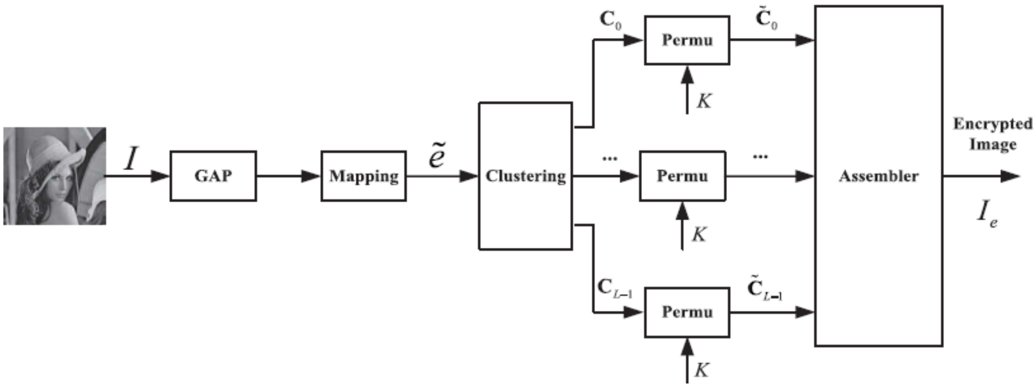

Fig1. Image Encryption

In this phase source image is applied to the image predictor GAP which converts image into pixels. Prediction error is calculated. In the clustering no of pixels grouped together and forms clusters. Once the clusters form then random permutation is applied on each cluster. At the assembler all the clusters combined and we will get encrypted image.

B.COMPRESSION PHASE:

Encrypted image is disassembled and apply arithmetic coding on each cluster. Again the clusters are assembled at the assembler and we will get compressed image.

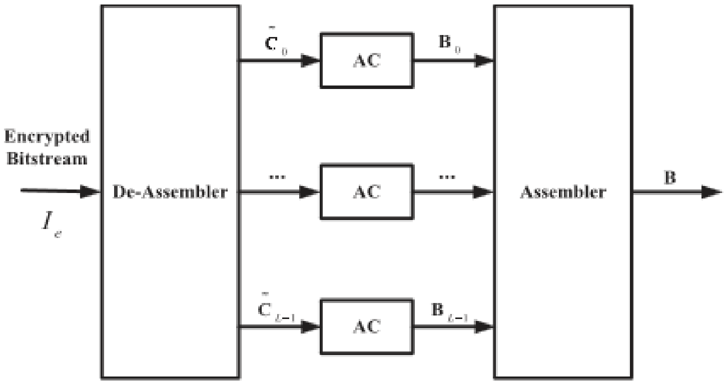

Fig.2 Compression Phase

C. DECRYPTION PHASE:

The receiver takes the help of the key to construct the secret image in the decryption process. The technique proposed is a unique one from the others in a way that the key is generated with valid information about the values used in the encryption process. At receiver receiver must know the prediction value of pixel.

IV. CONCLUSION

Proposed System is used to design a pair of image encryption and compression technique such that compressing encrypted images. The image encryption has been achieved via random permutation. And compression is achieved by using arithmetic coding where both lossy and lossless compression is considered. The analysis regarding the security of the proposed permutation-based image encryption method and the efficiency of compressing the encrypted data. For lossless compression and data hiding optical value transfer method can also be used.

REFERENCES

[1] X. Wu and N. Memon, “Context-based, adaptive, lossless image codec,” IEEE Trans. Commun., vol. 45, no. 4, pp. 437–444, Apr. 1997.

[2] M. J. Weinberger, G. Seroussi, and G. Sapiro, “The LOCO-I loss- less image compression algorithm: Principles and standardization into JPEG-LS,” IEEE Trans. Imag. Process., vol. 9, no. 8, pp. 1309–1324, Aug. 2000

[3] 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.

[4] T. Bianchi, A. Piva, and M. Barni, "Encrypted domain DCT based on homomorphic cryptosystems," EURASIP J. Inf. Security, 2009, Article ID 716357.

[5] T. Bianchi, A. Piva, and M. Barni, "Composite signal representation for fast and storage-efficient processing of encrypted signals," IEEE Trans. Inf. Forensics Security, vol. 5, no. 1, pp. 180–187, Mar. 2010.

[6] X. Zhang, "Lossy compression and iterative recobstruction for encrypted image," IEEE Trans. Inf. Forensics Security, vol. 6, no. 1, pp. 53–58, Mar. 2011

[7] M. Barni, P. Failla, R. Lazzeretti, A.-R. Sadeghi, and T. Schneider, "Privacy-preserving ECG classification with branching programs and neural networks," IEEE Trans. Inf. Forensics Security, vol. 6, no. 2, pp. 452–468, Jun. 2011.

[8] D. Klinc, C. Hazay, A. Jagmohan, H. Krawczyk, and T. Rabin, "On compression of data encrypted with block ciphers," IEEE Trans. Inf. Theory, vol. 58, no. 11, pp. 6989–7001, Nov. 2012.

[9] 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.