ISSN:2231-0711

Cinu Joseph et al | IJCSET(www.ijcset.net) | January 2016 | Vol 6, Issue 1, 40-42

A Survey of Encryption then Compression using Auxillary Information

Cinu Joseph[#1], Jasmy Davies[*2] , Vince Paul[*3]

[#1]*Cinu Joseph, M.Tech Computer Science,*
Sahrdaya College of Engineering,Kodakara

[*2]*Jasmy Davies,Assistant Professor,*
Sahrdaya College of Engineering,Kodakara

[*3]*Vince Paul,Assistant Professor,*
Sahrdaya College of Engineering,Kodakara

***Abstract*—Most commonly, image encryption has to be conducted prior to image compression. This has led to the problem of how to design a pair of image encryption and compression algorithms such that compressing the encrypted images can still be efficiently performed. is a highly efficient image encryption-then-compression (ETC) system. The proposed image encryption scheme operated in the prediction error domain can able to provide a high level of security. An arithmetic coding-based approach can be used to efficiently compress the encrypted images. Most of the existing ETC solutions induce significant penalty on the compression efficiency. Image hiding approach after the encryption phase is also used for getting more security. Gradient Adjusted Predictor (GAP) is used to find predicted values of the image.**

***Keywords*— Gradient Adjusted Predictor, Encryption then Compression, Compression then Encryption**

INTRODUCTION

In any application if content owner wants to send to securely and efficiently transmit an image I to a recipient via an untrusted channel provider .Content owner 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.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 .Therefore there is a need of encryption prior to image compression.

LITERATURE REVIEW

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, Alice is always interested in protecting the privacy of the image data through encryption. Nevertheless, Alice has no incentive to compress her data, and hence, will not use her limited computational resources to run a compression algorithm before encrypting the data. This is especially true when Alice uses a resource-deprived mobile device. In contrast,the channel provider Charlie 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 Charlie, who typically has abundant computational resources. A big challenge within such Encryption-then-Compression (ETC) framework is that compression has to be conducted in the encrypted domain, as Charlie does not access to the secret key K. The possibility of processing encrypted signals directly in the encrypted domain has been receiving increasing attention . At the first glance, it seems to be infeasible for Charlie to compress the encrypted data, since no signal structure can be exploited to enable a traditional compressor.

Although counter-intuitive, Johnson et al showed that the stream cipher encrypted data is compressible through the use of coding with side information principles, with- out compromising either the compression efficiency or the information-theoretic security[5] .In addition to the theoretical findings, many practical algorithms to losslessly compress the encrypted binary images are there. Schonberg et al later investigated the problem of compressing encrypted images when the underlying source statistics is unknown and the sources have memory[6],[7] .

By applying LDPC codes in various bit-planes and exploiting the inter/intra correlation , Lazzeretti and Barni presented several methods for loss-less compression of encrypted grayscale/color images[8]. Furthermore, Kumar and Makur applied an approach to the pre-diction error domain and achieved better lossless compression performance on the encrypted grayscale/color images . Aided by rate-compatible punctured turbocodes ,Liu et. al developed a progressive method to losslessly com- press stream cipher encrypted grayscale/color images

More recently, Klinc et al. extended Johnsons framework to the case of compressing block ci-pher encrypted data .To achieve higher compression ratios, lossy compression of encrypted data was also studied. Zhang et al proposed a scalable lossy coding framework of encrypted images via a multi-resolution construction [7] .A compressive sensing (CS) mechanism was utilized to compress encrypted images resulted from linear encryption. A modified basis pursuit algorithm can then be applied to esti-mate the original image from the compressed and encrypted data. Another CS-based approach for encrypting compressed images Zhang designed an image encryption scheme via pixel-domain permutation ,and demonstrated that the encrypted le can be efficiently com-pressed by discarding the excessively rough and fine information of coefficients in the transform domain . Recently, Zhang et al suggested a new compression approach for encrypted images through multi-layer decomposition .Extensions to blind compression of encrypted videos were developed .Despite extensive efforts in recent years, the existing ETC systems still fall significantly short in the compression performance, compared with the state-of-the-art lossless/lossy image and video coders that require unencrypted inputs. The primary focus of this work is on the practical design of a pair of image encryption and compression schemes, in such a way that com-pressing the encrypted images is almost equally efficient as compressing their original, unencrypted counterparts. Mean-while, reasonably high level of security needs to be ensured.

If not otherwise specified, 8-bit gray scale images are assumed .Both lossless and lossy compression of encrypted images will be considered. Specifically, we propose a permutation-based image encryption approach conducted over the prediction error domain. A context-adaptive arithmetic coding (AC) is then shown to be able to efficiently compress the encrypted data. Due to the high sensitivity of prediction error sequence against disturbances, reasonably high level of security could be retained.

PROPOSED ETC SYSTEM

We present the details of the three key components in our proposed ETC system, namely, image encryption conducted by Alice, image compression conducted by Charlie, and the sequential decryption and decompression conducted by Bob.

A. *Image Encryption Via Prediction Error Clustering and Random Permutation*

From the perspective of the whole ETC system, the design of the encryption algorithm should simultaneously consider the security and the ease of compressing the encrypted data. To this end, we propose an image encryption scheme operated *over the prediction error domain. For each pixel $I_{i,j}$ of the image I to be encrypted, a prediction* $\bar{I}_{i,j}$

or MED , according to its causal surroundings. In our work, the GAP is adopted due to its excellent de correlation capability. The prediction result $\bar{I}_{i,j}$ can be further refined to $\tilde{I}_{i,j}$ through a context-adaptive, feedback mechanism. The algorithmic procedure of performing the image encryption is then given as follows:

Step 1: Compute all the mapped prediction errors $\tilde{e}_{i,j}$ of the whole image I.

Step 2: Divide all the prediction errors into L clusters $\mathbf{C}_k$, for $0 \le k \le L-1$, where k is determined by (5), and each $\mathbf{C}_k$ is formed by concatenating the mapped prediction errors in a raster-scan order.

Step 3: Reshape the prediction errors in each $\mathbf{C}_k$ into a 2-D block having four columns and four rows

Step 4: Perform two key-driven cyclical shift operations to each resulting prediction error block, and read out the data in raster-scan order to obtain the permuted cluster $\tilde{\mathbf{C}}_k$.

Step 5: The assembler concatenates all the permuted clusters $\tilde{\mathbf{C}}_k$, for $0 \le k \le L-1$, and generates the final encrypted image .

Step 6: Pass I_e to Charlie, together with the length of each cluster $|\tilde{\mathbf{C}}_k|$, for $0 \le k \le L-2$.The values of $|\tilde{\mathbf{C}}_k|$ enable Charlie to divide I_e into L clusters correctly. In comparison with the file size of the encrypted data, the overhead induced by sending the length $|\tilde{\mathbf{C}}_k|$ is negligible.

B. *Lossless Compression of Encrypted Image Via Adaptive AC*

An adaptive AC is then employed to losslessly encode each prediction error sequence $\tilde{\mathbf{C}}$ In to a binary bit stream $\mathbf{B}_k$. Note that the generation of all $\mathbf{B}_k$ can be carried out in a parallel manner to improve the throughput. Eventually, an assembler concatenates all $\mathbf{B}_k$ to produce the final compressed and encrypted bit stream $\mathbf{B}$

C. *Sequential Decryption and Decompression*

Upon receiving the compressed and encrypted bit stream $\mathbf{B}$, Bob aims to recover the original image I. According to the side information $|\mathbf{B}_k|$, Bob divides $\mathbf{B}$ into L segments $\mathbf{B}_k$, for $0 \le k \le L-1$, each of which is associated with a cluster of prediction errors. For each $\mathbf{B}_k$, an adaptive arithmetic decoding can be applied to obtain the corresponding permuted prediction error sequence $\tilde{\mathbf{C}}_k$. As Bob knows the secret key K, the corresponding de-permutation operation can be employed to get back the original $\mathbf{C}_k$.

Let $\mathbf{CS}_k$ and $\mathbf{RS}_k$ be the secret key vectors controlling the column and the row shift offsets for $\mathbf{C}_k$. Here, $\mathbf{CS}_k$ and $\mathbf{RS}_k$ 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.

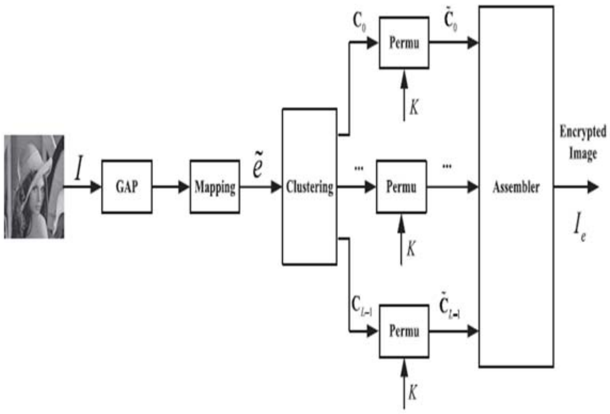

Fig. 1 A schematic diagram of Image Encryption

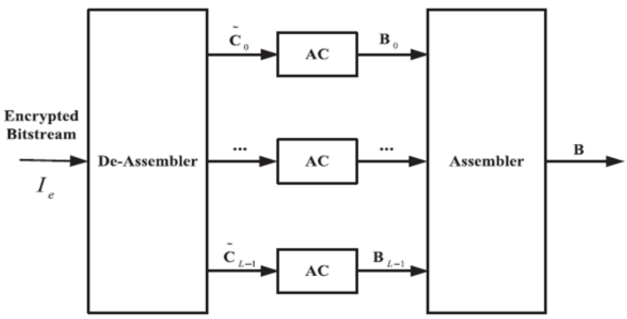

Fig. 2 Schematic diagram for compressing Encrypted data

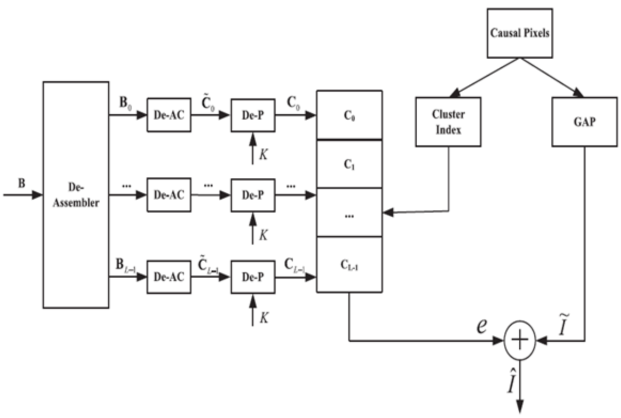

Fig. 3 Sequential Decompression and Decryption

CONCLUSIONS

We have designed an efficient image Encryption-then-Compression (ETC) system . Within the pro-posed framework, the image encryption has been achieved via prediction error clustering and random permutation. Highly efficient compression of the encrypted data has then been realized by a context-adaptive arithmetic coding approach .Proposed scheme ensures high level of security . Efficient utilization of the channel bandwidth can be also obtained using the proposed approach.

ACKNOWLEDGMENT

First and foremost I thank Almighty God for all the blessings endowed on me. I deeply express my sincere thanks to Mrs.Deepa Devassy, HOD of Computer Science And Engineering Department of Sahrdaya College Of Engineering And Technology for his whole hearted support. I also would like to express my Sincere gratitude to my Project guide
Mrs. Jasmy Davies and to my Project coordinator Mr Vince Paul ,who supported me throughout the project.I would like to extend my gratitude to all other faculty members for their help and advices.

REFERENCES

[1] Zhou, X. Liu, and O. C. Au, On the design of an e_cient encryption-then-compression system, in Proc. ICASSP, 2013, pp. 28722876..

[2] Bianchi, A. Piva, and M. Barni, On the implementation of the discrete Fourier transform in the encrypted domain, IEEE Trans. Inf. Forensics Security, vol. 4, no. 1, pp. 8697, Mar.2009..

[3] Bianchi, A. Piva, and M. Barni, Encrypted domain DCT based on homomorphic cryptosystems

[4] Bianchi, A. Piva, and M. Barni, Composite signal representation for fast and storagee_cient processing of encrypted signals, IEEE Trans. Inf. Forensics Security, vol. 5, no.1, pp. 180187, Mar. 2010.

[5] Barni, P. Failla, R. Lazzeretti, A.-R. Sadeghi, and T. Schneider, Privacy-preserving ECG classi_cation with branching programs and neural networks, IEEE Trans. Inf. Forensic Security, vol. 6, no. 2, pp. 452468, Jun. 2011.

[6] Erkin, T. Veugen, T. Toft, and R. L. Lagendijk, Generating pri- vate recommendations effciently using homomorphic encryption and data packing, IEEE Trans. Inf. Forensics Security, vol. 7, no. 3, pp. 10531066, Jun. 2012.

[7] Johnson, P. Ishwar, V. M. Prabhakaran, D. Schonberg, and K. Ramchandran, On compressing encrypted data, IEEE Trans. Signal Process., vol. 52, no. 10, pp. 29923006,Oct. 2004

[8] Schonberg, S. C. Draper, and K. Ramchandran, On blind compres-sion of encrypted correlated data approaching the source entropy rate, in Proc. 43rd Annu. Allerton Conf.,2005, pp. 13.