Clustering And Permutation Based Image Encryption And Compression System

Kalyani G. Nimbokar , Dr. M.V.Sarode

(M.E. Final Year Comp Sci Engg) J.C.O.E.T, Yavatmal, kalyanifriend08@gmail.com
(H.O.D. Comp. Engg) J.C.O.E.T, Yavatmal , mvsarode2013@gmail.com

ABSTRACT :
Image encryption has to be conducted prior to image compression. In this paper we study how to design a pair of image encryption and compression algorithms such that compressing encrypted images can still be efficiently performed. In this paper, we 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.

Keywords – Compression of encrypted image, encrypted domain signal processing .

1. INTRODUCTION

Image Encryption

The image encryption is to transmit the image securely over the network so that no unauthorized user can able to decrypt the image.[5] Encryption will be defined as the conversion of plain message into a form called a cipher text that cannot be read by any people without decrypting the encrypted text. Decryption is the reverse process of encryption which is the process of converting the encrypted text into its original plain text, so that it can be read. The transformation of plain text to cipher text is called as encryption. The transformation of cipher text to plain text is called decryption. Encryption and decryption are controlled by keys.[5]

As shown in Fig. (1), assuming that the plaintext and the ciphertext are denoted by P and C, respectively, the encryption procedure in a cipher can be described as C = E Ke (P), where Ke is the encryption key and E (•) is the encryption function.Similarly, the decryption procedure is P =DKd (C), where Kd is the decryption key and D (•) is the decryption function When Ke = Kd, the cipher is called a private-key cipher or a symmetric cipher For private-key ciphers, the encryption-decryption key must be transmitted from the sender to the receiver via a separate secret channel. When Ke ≠ Kd, the cipher is called a public-key cipher or an asymmetric cipher.[6] For public-key ciphers, the encryption key Ke is published, and the decryption key Kd is kept private, for which no additional secret channel is needed for key transfer. Ciphering the complete compressed file may result in excessive computational burden and power consumption at the decoder and perhaps even the server/ encoder.[6]

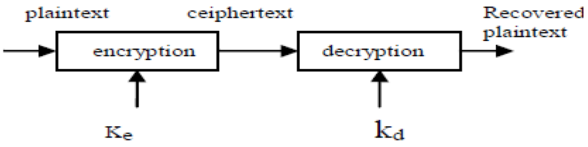

Fig.(1) Traditional Encryption Technique

Image data compression techniques are concerned with reduction of number of bits required to store or transmit images without any appreciable loss of information. Consider an application scenario in which a content owner Alice wants to securely and efficiently transmit an image I to a recipient Bob, via an untrusted channel provider Charlie. [1]Conventionally, this could be done as follows. Alice first compresses I into B, and then encrypts B into Ie using an encryption function EK (·), where K denotes the secret key as illustrated in Fig.2 (a). The encrypted data Ie is then passed to Charlie, who simply forwards it to Bob. Upon receiving Ie, Bob 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, 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.[1] 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. This type of ETC system is demonstrated in Fig.2(b).[1]

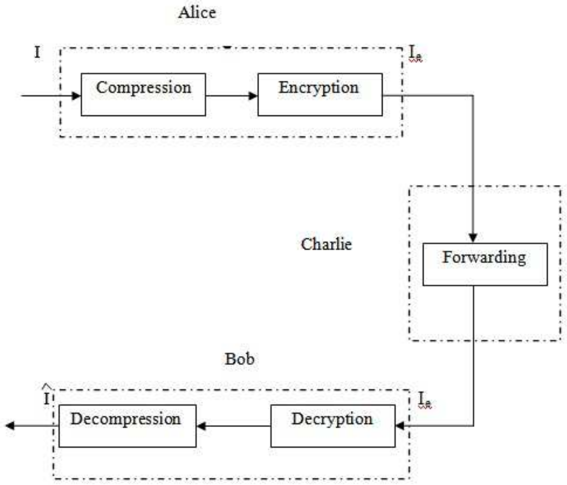

Fig.2 (a) Traditional Compression-then-Encryption (CTE) system.

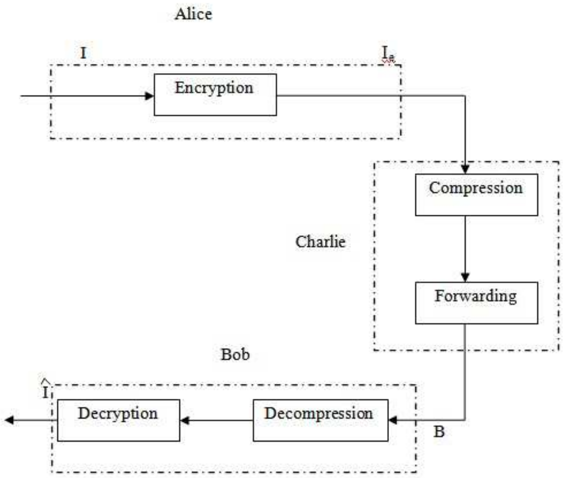

Fig.2 (b) Encryption-then-Compression (ETC) system.

Image Compression

Image compression is a process intended to yield a compact representation of an image, thereby reducing the image storage / transmission requirements. Generally, data compression is of two types: reversible compression (lossless) and non-reversible (lossy)compression.Reversible compression results in a reduction of redundant data, but the reduction is in such a way that redundancy can be subsequently restored into the data. Non-reversible compression results in the reduction of information itself in which the lost information can never be recovered. The non-reversible scheme provides more compression than its reversible counterpart [1]. Lossless versus Lossy compression: In lossless compression schemes, the reconstructed image, after compression, is numerically identical to the original image [1]. However lossless compression can only achieve a modest amount of compression. Lossless compression is preferred for archival purposes and often medical imaging, technical drawings, clip art or comics. But lossy schemes are capable of achieving much higher compression this is because lossy compression methods, especially used at low bit rates.An image reconstructed following lossy compression contains degradation relative to the original. Lossless methods concentrate on compacting the binary data using encoding algorithms - the most commonly used example is WinZip. To achieve higher levels of compression there are Lossy encoding techniques. Lossy encoding methods (such as JPEG) are varied but tend to work on the principle of reducing the total amount of information in the image in ways that the human eye will not detect. Irrelevance and redundancy of the image data is reduced by image compression because this is easy to store or transmit data in an efficient form. Reconstruct the original image by iterative reconstruction procedure. In this, flexible compression ratio is used and significantly improves the compression efficiency.

Lossy and lossless compression.

It is possible to compress many types of digital data in a way that reduces the size of a computer file needed to store it, or the bandwidth needed to transmit it, with no loss of the full information contained in the original file.[1] A picture, for example, is converted to a digital file by considering it to be an array of dots and specifying the color and brightness of each dot. If the picture contains an area of the same color, it can be compressed without loss by saying "200 red dots" instead of "red dot, red dot, ...(197 more times)..., red dot." The original data contains a certain amount of information, and there is a lower limit to the size of file that can carry all the information. Basic information theory says that there is an absolute limit in reducing the size of this data. When data is compressed, its entropy increases, and it cannot increase indefinitely. As an intuitive example, most people know that a compressed ZIP file is smaller than the original file, but repeatedly compressing the same file will not reduce the size to nothing. Most compression algorithms can recognize when further compression would be pointless and would in fact increase the size of the data. In many cases, files or data streams contain more information than is needed for a particular purpose. For example, a picture may have more detail than the eye can distinguish when reproduced at the largest size intended; likewise, an audio file does not need a lot of fine detail during a very loud passage. Developing lossy compression techniques as closely matched to human perception as possible is a complex task. Sometimes the ideal is a file that provides exactly the same perception as the original, with as much digital information as possible removed; other times, perceptible loss of quality is considered a valid trade-off for the reduced data.[1].

2.LITERATURE REVIEW

The possibility of processing encrypted signals directly in the encrypted domain has been receiving increasing attention in recent years [2]. 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, without compromising either the compression efficiency or the information-theoretic security [9]. In addition to the theoretical findings, also proposed practical algorithms to losslessly compress the encrypted binary images. By applying LDPC(Low Density parity Check) codes in various bit-planes and exploiting the inter/intra correlation, Lazzeretti and Barni presented several methods for lossless compression of encrypted grayscale/color images [5]. To achieve higher compression ratios, lossy compression of encrypted data was also studied.
Image data compression techniques are concerned with reduction of number of bits required to store or transmit images without any appreciable loss of information. Consider an application scenario in which a content owner Alice wants to securely and efficiently transmit an image I to a recipient Bob, via an untrusted channel provider Charlie. 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. This type of ETC system is demonstrated in Fig.1 [1]

3.ANALYSIS OF PROBLEM

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 compressing the encrypted images is almost equally efficient as compressing their original, unencrypted counterparts. Meanwhile, reasonably high level of security needs to be ensured. If not otherwise specified, 8-bit grayscale 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.[1] A context-adaptive arithmetic coding (AC) is then shown to be able to efficiently compress the encrypted data. Furthermore, due to the high sensitivity of prediction error sequence against disturbances, reasonably high level of security could be retained.[1]

4. PROPOSED WORK

Proposed work deals with the details of the three key components in proposed ETC system, namely, image encryption conducted by Alice, image compression conducted by Charlie, and the sequential decryption and decompression conducted by Bob. First key component Image Encryption Via Prediction Error Clustering and Random Permutation. the design of the encryption algorithm should simultaneously consider the security and the ease of compressing the encrypted data.[1] Then Second key components Lossless Compression of Encrypted Image Via Adaptive AC. The compression of the encrypted file Ie needs to be performed in the encrypted domain, as Charlie does not have access to the secret key K. And last key components Sequential Decryption and Decompression. Upon receiving the compressed and encrypted image, Bob aims to recover the original image.[1]

Permutation Based Image Encryption:

The technique involves three different phases in the encryption process.(fig .3) 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. The second phase is the key generation phase, where the values used in the encryption process are used to build a key. [7]The third phase is the identification process which involves the numbering of the shares that are generated from the secret image. These shares and the key are then transferred to the receiver. 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. Most of the encryption processes first generate the key and then do the encryption process. This technique generates a relation between the encryption process and the key. [7]

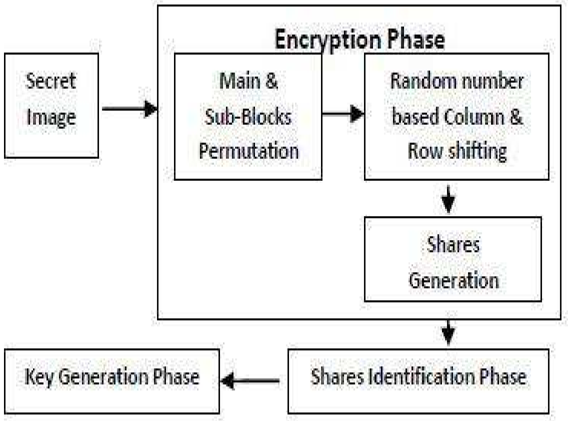
Encryption Phase
Secret Image
Main & Sub-Blocks Permutation
Random number based Column & Row shifting
Shares Generation
Key Generation Phase
Shares Identification Phase

Fig.3 Image Encryption Process

Signal processing tools working directly on encrypted data could provide an efficient solution to application scenarios where sensitive signals must be protected from an untrusted processing device. The possibility of processing encrypted signals directly in the encrypted domain (hereafter referred to as s.p.e.d., standing for signal processing in the encrypted domain) is receiving an increasing attention as a way to satisfy the security requirements stemming from applications wherein valuable or sensible signals have to be processed by a non-trusted party . The list of applications that would benefit from the availability tools is virtually endless, including: access to a database containing encrypted data or signals database access by means of encrypted queries, remote processing of private data, like medical recordings or biometric signals, by nontrusted parties, transcoding of encrypted contents, buyerseller watermarking protocols , just to mention some.[7]

CONCLUSION

In this paper we study how to design a pair of image encryption and compression technique such that compressing encrypted images can still be efficiently performed. We have seen the ETC (Encryption then compression) and CTE (compression then Encryption).we have designed an efficient image Encryption-then-Compression (ETC) system. Within framework, the image encryption has been achieved via random permutation. The analysis regarding the security of the proposed permutation-based image encryption method and the efficiency of compressing the encrypted data.

REFERENCES

1. Jiantao Zhou, Member, Ieee, Xianming Liu, Member, Ieee, Oscar C. Au, Fellow, Ieee,And Yuan Yan Tang, Fellow, Ieee" Designing An Efficient Image Encryption-Then-Compression System Via Prediction Error Clustering And Random Permutation" Ieee Transactions On Information Forensics And Security, Vol. 9, No. 1, January 2014.
2. Asha P. Ghodake, 2 Sujata Mendgudle Bharati Vidyapeeth College Of Engineering, Navi Mumbai, India Ramrao Adik Institute Of Technology, Navi Mumbai, India" Security And Privacy For Lossy Compression And Iterative Reconstruction For Encrypted Image" International Journal Of Electronics & Communication Technology.
3. Tiziano Bianchi, Member, Ieee, Alessandro Piva, Member, Ieee, And Mauro Barni, Senior Member, Ieee" Composite Signal

Representation For Fast And Storage-Efficient Processing Of Encrypted Signals" Evaluation–Copy/00$00.00 C2009 Ieee

4. .Riccardo Lazzeretti, Mauro Barni Department Of Information Engineering (University Of Siena)Via Roma 56, 53100, Siena, Italy" Lossless Compression Of Encrypted Grey-Level And Color Images" 16th European Signal Processing Conference (Eusipco 2008), Lausanne, Switzerland, August 25-29, 2008, Copyright By Eurasip.
5. Dr. Kamaljit I. Lakhtaria Mca Department, Atmiya Institute Of Technology & Science, Yogidham, Rajkot, Gujarat, India Kamaljit.Ilakhtaria@Gmail.Com" Protecting Computer Network With Encryption Technique: A Study" International Journal Of U- And E- Service, Science And Technology Vol. 4, No. 2, June, 2011.
6. Shaimaa A. El-Said Eng.Sahmed@Windowslive.Com Faculty Of Engineering Electronics And Communication Department Zagazig University Zagazig,44519, Egypt. Khalid F. A. Hussein Khalid_Elgabaly@Yahoo.Com Electronics Research Institute Microwaves Department Researches National Institute Dokki, Egypt Mohamed M. Fouad Fouadzu@Hotmail.Com Faculty Of Engineering / Electronics And Communication Department Zagazig University Zagazig,44519, Egypt." Securing Image Transmission Using In- Compression Encryption Technique" International Journal Of Computer Science And Security, (Ijcss), Volume (4): Issue (5).
7. Sesha Pallavi Indrakanti Associate Professor Department Of Computer Applications, Gvp Degree College (A), Visakhapatnam P.S.Avadhani Professor Department Of Cs And Se, Andhra University College Of Engineering(A),Andhra University, Visakhapatnam" Permutation Based Image Encryption Technique" International Journal Of Computer Applications (0975 – 8887) Volume 28– No.8, August 2011.