Research Paper

Volume : 2 | Issue : 3 | Mar 2013 • ISSN No 2277 - 8179

Compression & Security Techniques on image

Engineering

KEYWORDS: Compression, Security

Syed Mahmud Hossein District Officer, Regional Office, Kolaghat, D.V.E.& T.,Govt. of W.B.

ABSTRACT

Compression and Encryption techniques are getting very popular day by day as compression reduces file size & encryption ensures the security of a particular file which is to be sent over some unreliable network like internet. Our main objective was to encrypt a image file, compress it using any compression algorithm and send it over the mail such that the receiver gets the image file in more compressed format. On the way of doing this, at first we digitized a image file and encrypted it using selective Encryption algorithm. After that compressed it using Huffman algorithm. Information security is the most challenging question to protect the data from unauthorized user. This proposed method may also protect the data from hackers. It can provide the data security, the exact coded value are necessary for decoding time by using only exact coded value provided by encoded as well as decoded time, this proposed method protect the data by using ASCII code this types of data security is applied in tier one & two. In tier three we have use selective encryption techniques for better securities. Speed of encryption and security levels are two important measurements for evaluating any encryption system. This method are use only particular available ASCII code are able to encryption purpose or pattern are use for selection purpose also. The reverse process has been applied to get the original image file but in more compressed format. When a user searches for any an image, a encrypted compressed sequence file can be sent from the data source to the user. The encrypted compressed file then can be decrypted & decompressed at the client end resulting in reduced transmission time over the Internet.

1. Introduction

Compression[1] is used to minimize the amount of memory needed to represent an image. Images often require a large number of bits to represent them, and if the image needs to be transmitted or stored, it is impractical to do so without somehow reducing the number of bits. The problem of transmitting or storing an image affects all of us daily. Image compression is the application of Data compression on digital images. In effect, the objective is to reduce redundancy of the image data in order to be able to store or transmit data in an efficient form.The process of encoding plain text messages into cipher text messages is called as Encryption. The reverse process of transforming a cipher text message into plain text message is called as Decryption.

Selective encryption, where a part of compress text is encrypted keeping the remaining part unencrypted, can be a viable proposition for running encryption system in resource constraint devices. Selective encryption is the process of selecting a part of a whole compress text, to begin the process of encryption, keeping the remaining portion of the compress text in the clear in such a way that the security is not compromised. Mere only a fraction(fc)[fc denote that the fraction of is select from compress text] of whole compress text or plaintext is selected for encryption. Selection of the r part is vital for the security point of view. The criteria for selection of r vary according to the user. As r increases, the security level also increases at the cost of increased time of encryption.

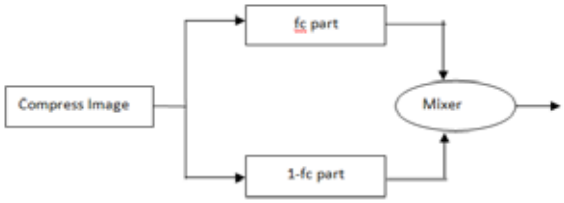

Figure 1.1

The encryption process is selective encryption of compressed text is shown in the figure 1.1. In this process of selective encryption, the property of reconstruction information of same algorithm is used in small fraction of compressed text is utilized[2]. This approach reduces the time complexity for encryptions and decryption due to encryption of the part of the compressed data, and also it reduces the storage and communication cost.

This approach is applicable on compressed image data and on other uses as consumer applications, texts has to be compressed due to constraint of the network bandwidth before communicating, it also needs to be encrypted to maintain confidentiality.

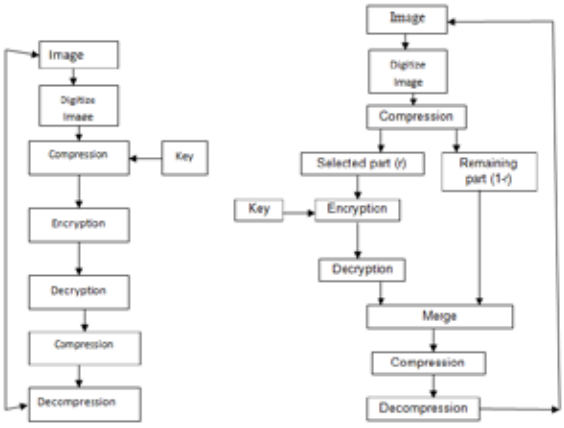

Figure 1.2 Figure 1.3

Through much work has to be done on selective encryption of images, videos, speech etc and not much work has been done on the selective encryption of compressed text data.

There is a similarity between the process of data compression and process of encryption. The goal for both the processes is to reduce the redundancy in the source message. According to Shannon [3], for perfect lossless compression algorithm, the average bit rate is equal to the source entropy. The redundancy of a source message is the difference between average bit rate and the source entropy of the message. The purpose of reducing the redundancy in case of compression algorithm is to conserver the storage space or communication bandwidth. The goal of reduction of redundancy in case encryption process is to thwart the different cryptanalysis attack based on statistical property of the source message. If we combine both the process of compression and encryption as shown in the fig- 1.2 & 1.3, then we can utilize another property of the compression algorithm that the decompression information are concentrated in a few portion of the bit stream, to selectively encrypt those portion of the bit stream which has got more impact on the reconstruction of the text during decompression process, keeping the remaining uncompressed bit stream in the clear.

For a perfect compression scheme, the plain text of the unencrypted portion of the message is statistically independent of the encrypted plain text message. So by knowing the unen-

crypted plain text, cryptanalyst can not infer anything for the encrypted plain text.

Due to the combination of the process of compression and the process of encryption, two benefits are realized:

1. Conservation of storage space and communication bandwidth
2. Encryption cost is reduced.
3. The attacks on the basis of statistical property of the source bit stream are thwarted.

We use compression & selection encryption techniques for the general purpose of sequence data delivery to the client. Existing image search engines do not utilise digitize image sequence compression algorithms & encryption for high security for client side decryption & decompression, i.e. where a encrypted compressed image sequence is decrypted & decompressed at the client end for the benefit of faster transmission & information security. Because most of the existing image sequence compression algorithms aim for higher compression ratios or pattern revealing, rather than client side & decryption decompression, their decompression times are longer than necessary information security. This makes these compression techniques unsuitable for the "on the fly" decompression. We use a encrypted compression technique designed for client side decrypted followed by decompression in order to achieve faster sequence secure data transmission to the client.

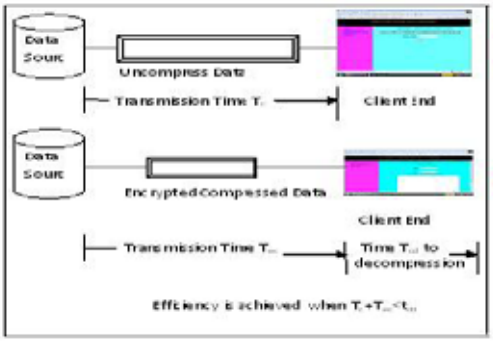

Fig 1.4

If encrypted compressed sequence data is sent from the data source to be decrypted decompressed at the client end and the decryption to decompression time along with the encrypted compressed file transmission time is less than the transmission time for uncompressed data transfer from the source to the client, then efficiency is achieved. Fig. 1.4 illustrates the situation. Note that the sequence data should be kept pre-compressed within the data source.

A digitized Sequence compression algorithm with reduced decompression time and moderately high compression rate is the preferred choice for efficient sequence data delivery with faster data transmission. As our target is to minimize decompression time and high information security, we use similar compression techniques to those used in [4], based on a "Two Pass" approach, meaning, that the file is compressed followed by encryption or decrypt followed by decompressed while reading it. Unlike "three pass" algorithms there is no need to re-read the input file. Our compression technique is essentially a symbol substitution compression scheme that encodes the sequence by replacing four consecutive nucleotide sequences with ASCI characters. Our technique to find the best solution for a client side decompression technique.

Information security Methodology
This techniques can provide three tier information security

- In tier one , the input image is digitized by MATLAB program by using digit, character or special character , after digitized the output file contain 256 ASCII code so, the output file is information secure than input file. This techniques can provide a information security.
- In tier two ; apply Huffman Technique : This process help compress the image as well as provide the security
- In tier three ; Selective Encryption

For better security purpose apply selective encryption techniques on compress output file, shown in Fig-2. If you apply selection encryption techniques in input file , the information security is very less due to less number of character contain the input file and also selection option are less. The compress output contain more characters than input file, in that situation applied selective encryption techniques, enjoy strong information security and selection option also more.

»""g"ai·ia(aiÂt±lmcrÈi|:²c(Â±!g¶ckt´t"c²(t±g%"g!²|,g)7i
p:i@±g%c²ctÀt!|glÈlaic´gcrk;cciitm&)±lc´tt"

Figure-2

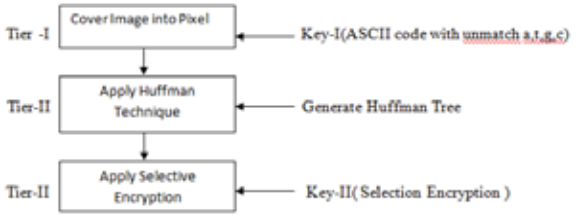

Fig-3.1

Selective encryption are apply in three ways i) Select only single character (any character from 0-255 ASSCII character ii) select Numeric numbers only iii) Pattern selection. For selection encryption purpose, generate private and public key.

- How the selectivity Algorithm improves on the standard approach :
"This is a sample file. The file has repeated instances of the text(word) file. The reason the text(word) file has been repeated in this file is because the text(word) file must be encrypted."

Here,

- Suppose it requires approximate time X for any character for a given set of keys.
- The time for any other computation is not necessary to be considered as
- X >>Time for linear operations like read/write/compare, etc.
- Number of characters = 173
- Confidential information is the text(word) "file" (4 characters string)
- Number of confidential characters = 4*6 =24
- Time savings
- Standard approach = ((173-24)/173)*100% = 86.127%
- My approach = ((173-4)/173)*100% = 97.688%

Flow chart

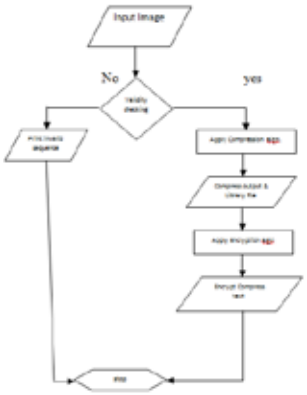

Fig-3.2 Show how to get Encrypt compress file.

Volume : 2 | Issue : 3 | Mar 2013 • ISSN No 2277 - 8179

Fig-3.3 show the reverse method to get original data

Software's:
In this project we have used MATLAB to generate the pixel matrix of the RGB color image we want to compress.
In our project we have used C-programming language to develop five programs

i) Compression Program(using Huffman Algorithm).
ii) Decompression Program(using Huffman Algorithm).
iii) Keygeneration Program.
iv) Encryption Program.
v) Decryption Program.

2. Working principle
The aim of our project is to take an image and compress the image i.e to reduce the original size of that particular image while keeping in mind to maintain the quality of the original image. The following logical steps are followed
i) Generation of Pixel-matrix from the image (that is to be compressed) using MATLAB.
ii) Encrypting the pixel matrix using selective encryption algorithm.
iii) Compression of the pixel matrix using C- program.
iv) Decompression of the compressed file.
v) Decryption of the decompressed image.
vi) Restoring the image

3. Excremental Result
Conversion of image to Pixel-Matrix in fig-4.1 & 4.2 using MATLAB Code and Huffman coding are using for compression shown in Fig-5written in[5]

Fig-4.1
O/p File:
Fig-4.2

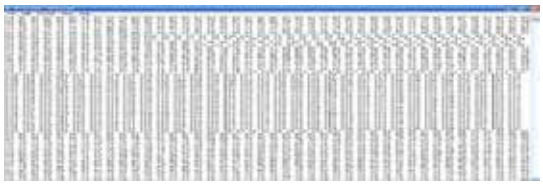

Fig-5 :Compression of the Encrypted Image by Huffman Algorithm

3.1: Selective Encryption & Decryption Algorithm
Selective Encryption Algorithm
1. Input File Name with Path.
2. Select Number or a Specific String.
3. Use RSA algorithm for encryption of the selected Number or Specified String.
4. Generate an Auxiliary File to keep the Flag for the specific regions of the Encrypted data.
5. Generate the Encrypted Output file.
6. Generate the Public Key and Private Key ultimately.

The Encryption Process:
We have developed the Encryption process of RSA Algorithm using C-program.After that, we take the pixel-matrix as the input file of that C-program.

HUFFMAN DECOMPRESSION PROGRAM IN C
O/P FILE:

Fig-6
Decryption of the decompressed image:
The output that comes from previous stage is actually nothing but the encrypted file.Now this file is taken as the input of the decryption program of the RSA algorithm which is implemented using C-program.

Selective Decryption Algorithm
1. Open Encrypted and Auxiliary File.
2. Input Encryption Option.
3. Read Encrypted data from Auxiliary File.
4. Use Private key to Decrypt data using RSA Module.
5. Get the Decrypted Output file.

O/P

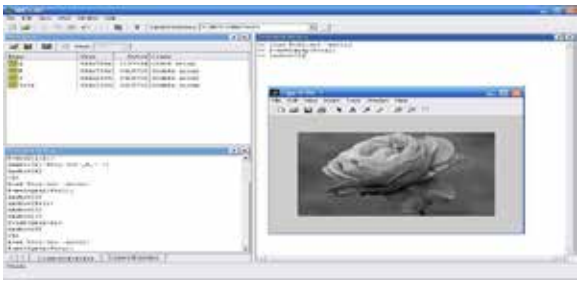

Fig-7 Original File

4. Future work
In order to compare the overall performance, we conducted further studies involving sending actual sequence files of varying sizes (without compression) to measure the calculated time (Tc) needed for the transmission from the source to the destination. Then we compressed those files using both compression & encryption algorithms. The total time T, defined as the sum of the encryption compressed file transmission time (Tec) plus the client side decompression time (Tdd), is measured by both these methods.

5. Conclusion
In our project, we have encrypted the pixel matrix of the image using Selective encryption algorithm & compressed the encrypted file using Huffman Compression Technique. The image

used is size of 994kb. And the encrypted file size is 2.38mb. But it is not convenient to send this huge file over network or Internet. So we have compressed the file & the compressed file size is 726kb. Thus the compression ratio achieved is 71%.Decoding the compressed file we get back the pixel-matrix of the original image from which the image could be restored. However there is one drawback. The mat2gray () command in MATLAB creates the grayscale image from the corresponding pixel-matrix. We still have not been able to restore the matrix to the RGB image.

REFERENCE

[1]H. Kobayashi and R. Bahl, " Image Data Compression by Predictive Coding I : Prediction Algorithm," IBM J. Res. Develop. 18, 164(1974), Proceding Paper [2] H. Cheng and X. Li, "Partial Encryption of Compressed Images and Video," IEEE Transactions on Signal Processing, 48(8), 2000, pp. 2439-2451. [3] C. E. Shannon, "Communication theory of secrecy systems," Bell Systems Technical Journal, v. 28, October 1949, pp. 656-715. [4] Chen, L., Lu, S. and Ram J. 2004. "Compressed Pattern Matching in DNA Sequences". Proceedings of the 2004 IEEE Computational Systems Bioinformatics Conference (CSB 2004) [5] Md. Syed Mahamud Hossein, N. Biswas, " Image Compression and Encryption", International Journal for Electro Computational World Knowledge Interface, vol.1, Issue 3, Nov.2011, pp 1-10.