HAMMING DISTANCE BASED COMPRESSION TECHNIQUES WITH SECURITY

Atul S. Joshi[1], Prashant R. Deshmukh[2]

[1]Associate Professor, Department of Electronics and Telecommunication Engineering, Sipna College of Engineering and Technology, Amravati, Maharashtra State, India

[2]Professor & Head of Department of Computer Science and Engineering, Sipna College of Engineering and Technology, Amravati, Maharashtra State, India

ABSTRACT

The proposed algorithm suggests a lossless data compression with security. Input Bit stream is divided into a group of 8-bits each .Encoder deviate the available Randomly generated key according to input bit stream which results into encrypted data. Encrypted key of 64-bits is treated as unit of 4-bits each. Each 4-bits of Key will be at the Hamming distance is of Two from Deviated key. There will be total Six numbers of 4-bits which are at the Hamming distance of Two from Key. These numbers can be indexed by using 3-bits. The index of Three bits of a number is available on the channel as a compressed bit stream. Proposed algorithm is to encrypt the file, compress it, decompress it, and finally decrypt it back to the original file. The algorithm not requires a prior knowledge of data to be compressed. Intelligent algorithm reverse the order of compression & encryption (i.e. Encryption prior to compression) without compromising compression efficiency , information theoretic security & with lesser computational cost. Proposed algorithm suitable for images, audio as well as text.

KEYWORDS: *Compression, Decompression, Encryption, Decryption, Key*

I. INTRODUCTION

Developing technology increased the need for storing data. Several applications in the field of multimedia [1] achieved a lot of attention towards data compression to conserve the Bandwidth. With the increasing amount of data stored on computers, the need for security in transmission has also gained attention towards encryption. Compression aids encryption by reducing the file size, "the compression scheme shortens the input file, which shortens the output file and reduces the amount of CPU required to do the encryption algorithm, so even if there were no enhancement of security, compression before encryption would be worthwhile. However, concerning compression after encryption it is stated; "If an encryption algorithm is good, it will produce output which is statistically indistinguishable from random numbers and no compression algorithm will considerably compress random numbers" [2]. On the other hand, if a compression algorithm succeeds in finding a pattern to compress out of an encryption's output, then a flaw in that algorithm has been found. This algorithm reverse the order of compression & encryption (i.e. Encryption prior to compression) without compromising compression efficiency or information theoretic security. Proposed algorithm is designed to achieve both compression and confidentiality by using Symmetric keys. Algorithm is suitable for images, videos as well as text. We develop framework based on Hamming distance for

joint encryption and compression Distributed source coding [3] has emerged as an alternative to achieve low-complexity compression for correlated sources. Johnson et al. proved that reversing the order of compression and encryption to compress the encrypted data can still achieve significant compression [4]. Computational cost of the present work is also lesser. **Paper is organized as follows:** Section 2 discusses related work in this area. Section 3 provide proposed scheme. In Section 4 evaluation methodology is describe along with comparison results with existing methods. Future scope is described in Section 6.

II. RELATED WORK

Wavelet predictive algorithm results well for relatively small data set. If the high degree of image compression is to be achieved then wavelet algorithm closely approximates the original data sets . However algorithm would be no useful for the text compression because there is no underlining deterministic process in natural language text [5].Huffman procedure proposed by Wolfe and Chanin [6] creates optimal code for set of symbol and probability subject to constraints that symbol be coded at one time. It is very effective as both the frequency and probability occurrence of the source symbol are taken into account. But since tree progressively spares results in lengthy search procedure for locating the symbol. Daniel Hillel Schonberg [7] presents practical distributed source code that provide framework for compression of encrypted data. Since encryption masks the source code, traditional compression algorithm is ineffective. M. A. Haleem, K.P. Subbalakshmi & R. Chandramouli [8] proposed a Joint encryption & compression scheme. It reduces complexity of compression process & at the same time use cryptographic principles to ensure the security. Dr. V.K. Govindan & B.S. Shajee [9] Mohan proposed better encoding scheme which offers higher compression ratio & better security towards all possible ways of attack. This algorithm compression transforms the text into some intermediate form which can be compressed with a better efficiency. Dictionary based code encoding techniques suggested by H. Lekatasas and J. Henkel [10] provide good compression efficiency as well as fast decompression mechanism. The basic idea is to take advantage of commonly occurring instruction sequence by using a dictionary & repeatedly occurrence are replaced by codeword that point to index of dictionary[11].

III. PROPOSED SCHEME

$X \rightarrow$ Input binary data of 8 bits

$Y \rightarrow$ Randomly generated binary Key of 8 bits

$Z_E \rightarrow$ Encrypted binary output

$Z_C \rightarrow$ Compressed binary output

$W_C \rightarrow$ Decompressed binary output

$W_D \rightarrow$ Decrypted binary output

$X = \{Xi\}$ & $Y = \{Yi\}$ Where i = 0 to 7

$\sum_{i=0}^{1} Xi \, . \, 2^i = \Delta (Y_1 Y_0)$

$\sum_{i=2}^{3} Xi \, . \, 2^i = \Delta (Y_3 Y_2)$

$\sum_{i=4}^{5} Xi \, . \, 2^i = \Delta (Y_5 Y_4)$

$\sum_{i=6}^{7} Xi \, . \, 2^i = \Delta (Y_7 Y_6)$

$Z_E = \Delta Y = \{\Delta Yi\}$ ---------------------------------- **(1)**

$H_d [\Delta(Y_1 Y_0), (Y_1 Y_0)] = 1$

$H_d [\Delta(Y_3 Y_2), (Y_3 Y_2)] = 1$

$H_d [\Delta(Y_5 Y_4), (Y_5 Y_4)] = 1$

$H_d [\Delta(Y_7 Y_6), (Y_7 Y_6)] = 1$

Thus $H_d [\Delta(Y_3Y_2Y_1Y_0), (Y_3Y_2Y_1Y_0)] = H_d [\Delta(Y_7Y_6Y_5Y_4), (Y_7Y_6Y_5Y_4)] = 2$

Suppose $S = \{Si\} = \{ 001,010,011,100,101,110 \}$

Thus $Z_C = (Z_C'' Z_C')$ - **(2)**

Where Z_C'' & $Z_C' \in S$

$W_C = (W_C'' W_C'$

Where $W_C' = f [Z_2', (Y_3Y_2Y_1Y_0)]$

$W_C' = \sigma \Delta(Y_3Y_2Y_1Y_0) \sim [Z_2', (Y_3Y_2Y_1Y_0)]$ &

$W_C'' = \sigma \Delta(Y_7Y_6Y_5Y_4) \sim [Z_2', (Y_7Y_6Y_5Y_4)]$

i.e. $W_C' = \Delta(Y_3Y_2Y_1Y_0)$ & $W_C'' = \Delta(Y_7Y_6Y_5Y_4)$ - - - - - - - - - - - - - - - **(3)**

$W_D = \{W_{Di}\} = \Delta Yi \, . \, 2^i = X_i$ Thus $W_{D\,i} = X_i$

Hence $\mathbf{W_D = X}$ - **(4)**

IV. EVALUATION METHODOLOGY OF PROPOSED WORK

In the proposed joint encryption & compression scheme MATLAB tool is used for simulation. As a input Text, Audio & Image is provided to Encoder. Another input to encoder is Binary Key of 64 bits is generated through pseudorandom generator & it is transmitted towards receiver through secure channel. Decoder which is joint decryptior & decompressor, reconstruct data again. Compression Ratio & Security Performance is tested.

4.1 DEGREE of SECURITY

The key size and the randomness of the encrypted plaintext are two major factors to analyze the degree of security. The amount of time that required breaking a cryptosystem can be measured by $T = 2^{k-1} t$ .Here k is the size of the encryption key, t is the amount of time needed for encryption of plaintext. In proposed algorithm the size of the key is 64 bits & Key generation is random & it is independent of input bit stream. Symmetric key [12] is used to process the data results into fast encryption as compare to asymmetric key algorithms. Technique used for encryption of input data using key is not similar for all chunks. Encryption method changes depend on the parity of sequence of input bit stream. This offers good encryption strength in proposed algorithm.

4.2 COMPRESSION RATIO & SAVING IN COMPUTATION

Speed of both compression and decompression is important. Implementation of this algorithm achieves a average compression ratio of about 70% with saving in compression & decompression time [13]. In most of the other algorithm data undergoes initial key addition and substitution. Each round requires row shifting and column mixing operation followed by the addition of a round key and substitution. In the proposed scheme, compression is based on Hamming distance of 'Two' bit key & encrypted data. Total Six numbers with Hamming distance of two are indexed using three bits & these bits used for compression. In short since algorithm has to search for only six numbers at a time hence computational cost of the said algorithm is challenging. A bar chart given below is drawn on the basis of the observation on the ratio differences between the various algorithms.

4.3 COMPARISION of RESULTS

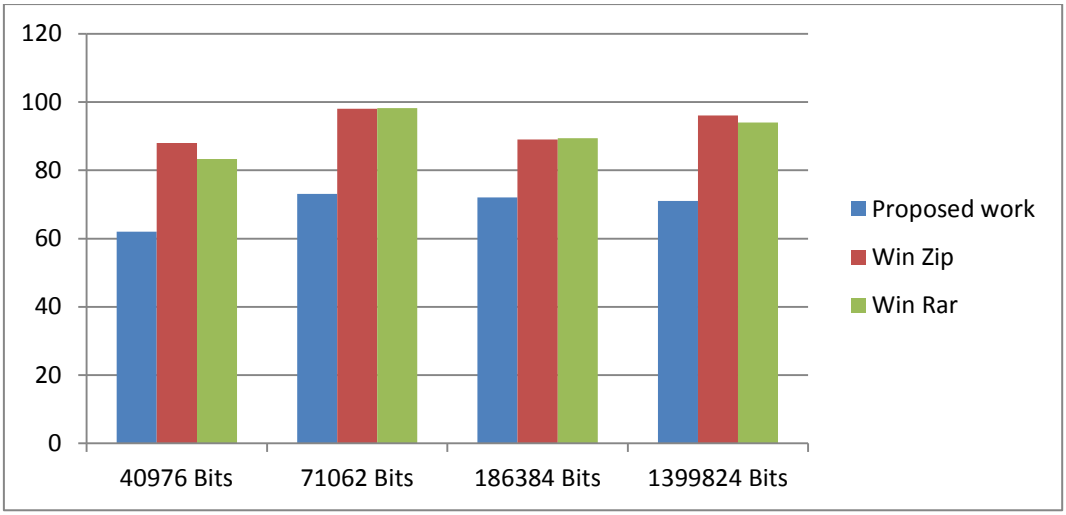

Figure 1: Percentage Compression of Various Algorithms

X-axis reflects the percentage compression ratio of the file after converting it to binary bit stream applying algorithm, while the Y-axis reflects the bit size. Results are taken for the key size of 64 bits. Each bar has a unique color in order to identify the algorithm. Blue bar represents our proposed algorithm. Red bar represents WinZip algorithm. Green bar represents WinRar algorithm. Text input of 40976 bits is taken from the file 'Geeta in English'. Audio input of 71062 bits is taken from the 'Audio Word Wave' file. Stored Image input of 186384 bits is taken from 'Flower' image. Fourth input of 1399824 bits is taken from 'Photo Gallery' For the First input we obtained 62% compression, where as WinZip & WinRar provide 88.02%& 83.34% respectively. For the Second input we obtained 72.01% compression, where as WinZip & WinRar provide 98%& 98.16% respectively. . For the Third input our algorithm provide compression of 72%, where as WinZip & WinRar provide 89% & 89.42% respectively. For the Fourth input our algorithm provide compression of 71.06%, where as WinZip & WinRar provide 96.08% & 94.02% respectively.

V. CONCLUSION

The proposed work is based on symmetric key joint encryption & compression algorithm provides lossless data compression with security. Reversal the order of encryption & compression is not affecting compression efficiency & information theoretic security Computational complexity is less as compare to other algorithms. On an average compression of about 69.026 % approximately 70% is achieved with proposed scheme shows better performance as compare to WinZip & WinRar.

VI. FUTURE SCOPE

The authors believe that there are additional research opportunities in this work with variable Key size & with variable chunks size of input binary bit stream.

ACKNOWLEDGEMENT

First of all I would like to record my immense gratitude toward Respected Supervisor Dr. Prashant Deshmukh whose guidance and conclusive remarks had a remarkable impact on my work. I am also indebted to all my colleagues who supported me during this work. Last but not least, as always, I owe more than I can say to my exceptionally loving Guru Achyut Maharaj, my Parents, my wife & daughter Adya whose support pave every step of my way.

REFERENCES

[1]. C.-P. Wu and C.-C. J. Kuo, "*Efficient multimedia encryption via entropy codec design*," in security and Watermarking of Multimedia Contents III, vol. 4314 of Proceedings of SPIE,pp.128–138, San Jose, Calif, USA, January 2001.

[2]. A. Hauter, M.V.C., R. Ramanathan. *Compression and Encryption. CSI 801Project Fall 1995*. December 7, 1995 [cited 10 March2006]; Available from: http://www.science.gmu.edu/~mchacko/csi801/proj-ckv.html.
[3]. Seon-Won Seong & P. Mishra , " *Bitmask based code compression for embedded system* " , IEEE transaction on computer aided design of integrated circuit & system , vol. 27 , No. 4 , April 2008 , pp 673-685.
[4]. S. Shani, B.C. Vemuri , F. Chenc Kapoor, " *State of art image compression algorithm*" , October 30, 1997.
[5]. ChairatRittirong, Yuttapong Rangsanseri &Punya Thitimajshima , " *Multispectral Image Compression using Median Predictive Coding and Wavelet transform*", in GIS proc., 1999
[6]. A. Wolf & A. Chanin , " *Executing compressed program of embedded RISC architecture*", in poc. Int. symp. Micro, 1992, pp 81-91
[7]. Daniel Hillel Schonberg , " *Practical Distributed Source Coding & its application to the compression of Encrypted data*" , Technical Report No. UCB/EECS-2007-93, July 2007
[8]. M.A. Haleem , K.P. Subbalakshmi , R. Chandramouli , "*Joint encryption & compression of correlated sources*", EURASIP Journal on Information Security , Jan. 2007
[9]. Dr. V.K. Govindan , B.S. Shajee Mohan , " *IDBE – An intelligent Dictionary Based Encoding Algorithm for text data compression for high speed Data transmission*", Proceeding of International conference on Intelligent signal processing , Feb 2004.
[10]. H. Lekats, J. Henkel , " *Design of one cycle decompression hardware* " in poc. Des. Conf. 2002 , pp 34-39
[11]. C. Castelluccia and A. Francillon, Tiny RNG, '*a cryptographic random number generator for wireless sensor network nodes"*. In:5th International Symposium on Modeling and Optimization inMobile, Ad Hoc, and Wireless Networks. IEEE, New York, NY,USA, 2007
[12]. Gred E. Keiser , " *Local area network* " , Tata Mc Graw Hill Edition , 1997, pp 443-497
[13]. R. L. Dobrushin, "*An asymptotic bound for the probability error of Information transmission through a channel without memory using the feedback*," Problemy Kibernetiki, vol. 8, pp. 161–168, 1962.
[14]. Riyad Shalabi and Ghassan Kanaan , "*Efficient data compression scheme using dynamic Huffman code applied on Arabic*" , in journal of computer science Dec 2006
[15]. Irina Chihaia and Thomas Gross , "An analytical model for software only main memory compression", in proc.,ACM Int. Conf. Series , vol. 68
[16]. Behrouz Forouzan , "*Introduction to data communication and networking* ", Tata McGraw Hill Edition 1999
[17]. J. Prakash & C. Sandeep , " *A simple & fast scheme for code compression for VLIW processor* " , in proc., DCC , 2003.
[18]. Montserrat Ros & Peret , " *A Hamming distance based VLIW/ EPIC code compression technique* ", Proceeding of International conference on compilers, Architecture & synthesis for Embedded system , 2004..
[19]. Chia – Wei Lin , Ja – Ling Wu & Yuh – Jue Chang , "*Two Algorithm for constructing efficient Huffman code based reversible variable length code*" , IEEE transaction on communication ,vol. 56 , No. 1, January 2008 , pp 81-88
[20]. E. Celikel and M. E. Dalkılıç," *Computer and Information Sciences"*ISCIS 2003, Lecture Notes in Computer Science, Vol.2869/2003.

AUTHORS INFORMATION

Atul Joshi is currently working as a Associate Professor in Department of Electronics & Telecommunication Engineering, at Sipna College of Engineering & Technology. He is pursuing his PhD in Electronics. His areas of interest are Communication Engineering, Communication Network & Electronic Circuits Design.

Prashant Deshmukh is currently working as Head of CMPS & IT Department, at Sipna College of Engineering & Technology, Amravati (India). He has completed his Ph.D. in the faculty of Electronics Engineering From SGBAmravati University, Amravati (India). His areas of interest are Digital Signal Processing, VLSI Design and Embedded Systems.