International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-2 Issue-3, February 2013

Hamming Distance Polygram Substitution Algorithm for Coding Optimization & Security

Atul S. Joshi, P.R.Deshmukh

Abstract*— The joint approach of integrating selective encryption & coding optimization is presented in this paper. Binary bit stream of the input is divided into the plaintext chunk of 64 bits. Random Key of 128 bits is generated. Key bits are then selected randomly. These randomly selected bits are change again randomly according to plaintext bits. Hamming distance is calculated in between the plaintext & changed key bits. Based on this Hamming distance codebook is form. Index of the codeword is treated as a cipher text which is itself a compressed code. Two levels of encryption is achieved in this work which makes the algorithm more secured than other encryption algorithm. The proposed algorithm is compared for standard test image on the basis compression performance & computational complexity. The result taken shows better performance of the proposed method over other standard methods

Index Terms*— Hamming distance, Polygram substitution, Key, Encryption, Compression.

I. INTRODUCTION

Multimedia data security using encryption techniques became more important now a days because of the transmission of the data on open network. The large size of audiovisual data requires a considerable amount of computational cost.[1] – [3].Encryption algorithm should be strong enough that the time and resources lost by the attacker while decoding the code and tracking the algorithm should be more than actual value of information. The sender generates the message containing the information to be communicated. This message is in plain text and therefore cannot be transmitted on an insecure channel. Hence this message is encrypted using the encryption algorithm to generate cipher text. A secret key is used by the encryption algorithm to generate cipher text which is known only to the sender and the intended receiver. This cipher text can be interpreted only by those individuals whose know how it was encrypted i.e. who have the decryption algorithm and the secret key. The intended receivers decrypt the message by running the decryption algorithm and obtain the readable copy of the message. The time and resources taken to recover the plaintext, measures the strength of a cryptographic method [4]. The early shift ciphers were very vulnerable to attacks. However the substitution cipher is more complex than a standard shift cipher[5]. The proposed algorithm is based on Polygram substitution which provides complex cipher. Polygram substitution permits arbitrary substitution for the group of plaintext bits.

Encryption is performed on the selective key bits by changing these bits with respect to the decimal value of the plaintext bits. Encryption algorithms have a property to randomize the source data and thus negatively affect compression performance. The attempt is made here in this work to create an encryption that preserves also the compressibility of the data. To compress the data Hamming distance of the transformed key bits with the actual key bits are taken into account and Polygram substitution is used to optimize the bits. The compression in this algorithm not only conserve the bandwidth require for data transmission but also provide another level of security for the data. This algorithm reverse the order of compression &encryption i.e. Encryption prior to compression without compromising compression efficiency or information theoretic security.

The proposed Joint selective encryption & compression algorithm is immune to security attacks because of large size of the randomly generated key and flexible as well as adaptive nature of the algorithm to generate different ciphers even for the same plaintext with a lower computational cost.

Simulation results are tested for the image input on the parameters of Compression Ratio and PSNR.These parameters when compared with the other algorithm provides better results.

II. PRPPOSED SCHEME

Hamming Distance Polygram Substitution (HDPS) algorithm work on the two different approaches for similar and dissimilar bits in the plaintext bit stream. Both these approaches are explained below using mathematical model. This algorithm uses a joint mechanism of encryption & compression. The encryption is a selective encryption .
Analysis and experimental results indicated that the selective encryption algorithm is not suitable for the multimedia applications. the basic concept of selective encryption
is to select the most important coefficients from either final results or intermediate steps of a compression system, and then encrypt them . Another problem with these algorithms is that they decrease compression efficiency. However result of HDPS algorithm shows that irrespective of the selective encryption technique used in the algorithm, HDPS gives better performance in compression as well as in computational complexity.

Manuscript published on 28 February 2013.
***Correspondence Author(s)
Prof. Atul S. Joshi*, Dept. of Electronics & Telecommunication, Sipna college of Engineering & Technology, Amravati(M.S.), India.
Dr. P. R. Deshmukh, Dept. Computer Science & Engineering Sipna College of Engineering & Technology, Amravati (M.S.), India.

MATHEMATICAL MODEL

X → Input binary data of 8 bits
Y→ Randomly generated binary Key of 8 bits
Z_E→ Encrypted binary output
Z_C→ Compressed binary output
W_C→ Decompressed binary output
W_D→ Decrypted binary output
X = {Xi} & Y = {Yi } Where i = 0 to 7

I. DISIMILAR BITS APPROCH

$\sum_{i=0}^{i+1} Xi \cdot 2^i = \Delta Y_i$
$Z_E = \Delta Y = \{\Delta Yi\}$
$H_d [\Delta(Y_1\ Y_0), (Y_1\ Y_0)] = 1$
$H_d [\Delta(Y_3\ Y_2), (Y_3\ Y_2)] = 1$
$H_d [\Delta(Y_5\ Y_4), (Y_5\ Y_4)] = 1$
$H_d [\Delta(Y_7\ Y_6), (Y_7\ Y_6)] = 1$
i.e. $H_d [\Delta(Y_3Y_2Y_1Y_0), (Y_3Y_2Y_1Y_0)] = H_d [\Delta(Y_7Y_6Y_5Y_4), (Y_7Y_6Y_5Y_4)] = 2$
Hence Hd $[([Y_i]_i^{i+4}, [\Delta Y_i]_i^{i+4})]_i^{i=NO.OF\ MSB} = 2$

i.e. Suppose for Y & Y' represents changed bit , then for Hamming distance of Two there are Total Six possible numbers as

1. $Y_3Y2Y1'Y0'$
2. $Y_3Y2'Y1Y0'$
3. $Y_3'Y2Y1Y0'$
4. $Y_3Y2'Y1'Y0$
5. $Y3'Y2Y1'Y0$
6. $Y3'Y2'Y1Y0$

These Six numbers can be code using Three bits as

1. Y3Y2Y1'Y0' → 001
2. Y3Y2'Y1Y0' → 010
3. Y3'Y2Y1Y0' → 011
4. Y3Y2'Y1'Y0 → 100
5. Y3'Y2Y1'Y0 → 101
6. Y3'Y2'Y1Y0 → 110

Hence the resultant code is the pair of **Two 3-bits** numbers (Total Six bits) that results into compression. Let S_{ic} is the set of Index of Hamming Distance between ΔY& Y. Since total numbers of 4-bit number having Hamming distance of **Two**, are **Six**, they can be index by using 3-bits.

i.e. $\{S_{ic}\} = \{001,010,011,100,101,110\}$

Thus $Z_C = S_{ic}$
$W_C = S_{ic}$
$\Delta Y_i = f [(Y_i , W_i)]$
$W_D = i \in \Delta Y_i = \sum_{i=0}^{i+1} Xi \cdot 2^i = [X_i]_{i=0}^{i+1}$
Hence $W_D = X$

II. SIMILAR BITS APPROCH

If $[X_i]_i^{i+1} = [X_i]_{i+1}^{i+2}$
i.e $[X_i]_i^{i+1} = [X_i]_{i+1}^{i+2} = 00$
Then Z_E = aaaa
& Z_C = 00
W_c = aaaa
& W_D = 0000

III. COMPUTATIONAL COMPLEXITY

Computational Complexity of the algorithm is evaluated by calculating the number of operations requires for the Plaintext Set up, Key Set up, Encryption &Compression [37] of the algorithm.

1. Plaintext Setup Cost (P_S):

When multimedia input is provided to the algorithm; it is first converted to Binary bitstream.Let us consider that the single operation is require for conversion of the input to binary bit stream **(Con).**Another operation is require for the division of binary bit stream into 64 bits each **(Div).**There after single operation is requires to form a chunks of 4 bits in each 64 bit frame **(C_4).**Hence for 64 bit frame the number of C_4 operations requires to be 64/4 = 16.

Total Plaintext Setup Cost is the algebraic sum of operations mentioned above.

i.e. **P_S = 1(Con) + 1(Div) + 16(C_4) = 18**

2. Key Setup Cost (K_S):

The Key setup cost includes all the computations prior to actual encryption process of the first bit of the plaintext [1].Let us assume that the single operation is requires for random key generation of 128 bits by using Pseudorandom Sequence generator for each 64 bits **(K_G).**Key bits are selected thereafter randomly equal in number with the plaintext bits **(K_B).** Then chunks of 4 bits are formed from total 64 K_S bits that requires 16 operations **(K_4).**Thus total Key setup cost is given by

K_S = 1(K_G) + 1(K_B) + 16 (K_4) = 18

3. Encryption Cost (E):

Selective Encryption technique is employed in proposed algorithm. It starts with the finding of decimal value of the bits from every chunk of 4 bits **($\sum_{i=0}^{i+1} Xi \cdot 2^i$)**. If this is consider as a single operation then for each chunk 2 operations are require. Since each 64 bit frame has 16 chunks, total 16* 2 = 32 such operations are requiring. According to decimal value of bits, A single bit of K_4 gets change. Hence total 32 K_B bits gets change requires 32 operations **(ΔK_B).**Hence total Encryption Cost is

E = 32 ($\sum_{i=0}^{i+1} Xi \cdot 2^i$) + 32($\Delta K_B$) = 64

4. Compression Cost (C):

In the proposed scheme compression requires Hamming Distance (HD) calculations and Polygram Substitution. (PS). Cipher text (Ci) available as a input to compression process is the combination of K_G & ΔK_B bits**.** Compression procedure starts with the formation of the codebook on the basis of Hamming distance. Using 4 bit binary total 16 codes are possible. For each 4 bit chunk of the cipher text code (Ci_4) total 6 codes are at a Hamming Distance of 2 from that particular code. These 6 codes are index using lesser bits. Each Ci_4 is entering in the codebook against the 6 codes of Hamming distance 2 from particular Ci_4 and its 6 indices. For codebook formation each Ci_4 requires 6 operations for 6 codes and another 6 operations are requires for 6 indices. It means that each Ci_4 requires 12 operations. Since there are 16 Ci_4 in 64 bits, the number of operations requires to be 12*16 operations **(C_O)**. The algorithm compares the bit of cipher text with the bits of K_G, chunk by chunk. Since in 64 bits there are 16 chunks, 16comparisions are requires **(C_P).**Comparison is to find out Hamming distance code which exist in codebook from each 4 bit chunk of K_G by noticing the changed bits. For 16 chunks of K_G, requires 16 such operations **(Hd_C).**Each (Hd_C) is there after replace by the corresponding Codebook Index of 3 bits results into compressed code further requires 16 operations **(C_I).** Thus total Compression Cost is given by

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

$C = 12 * 16 (C_O) + 16 (C_P) + 16 (Hd_C) + 16 \quad (Ci)$
$= 240$

Total Computational Cost of the algorithm is

$CC = P_S + K_S + E + C$
$= 18 + 18 + 64 + 240$
$= 340$ **operations.**

However the Average value of the Computational Cost is less than 340 operations for maximum input frame size of 64 bits. The above cost is calculated by considering first 2 bits of the C_4 are not similar with the next two bits of the plaintext. When they are similar algorithm work with different approach results into lesser value of Encryption and Compression cost as explained below

5. Encryption Cost for Similar Bits (E_S):

In this case algorithms need not to calculate $\sum_{i=0}^{i+1} Xi \cdot 2^i$. Also ΔK_B operations are not require. There is Direct Assignment of the the alphabets like aaaa depend upon the 2 bit similarity for C_4 irrespective of the K_B. Hence for 16 chunks, 16 such operations are reqires**(D_A).**

Hence $E_S = 16(D_A) = 16$

Average value of the Encryption Cost (AE)

$= E + E_S / 2 = 64 + 16 / 2$ Thus, AE = 40

6. Compression Cost for Similar Bits (C_S):

When the first 2 bits of C_4 is similar with next 2 bits, steps taken by the algorithm is comparatively simple results into very less computational cost. In this case there is no need of codebook formation and further operations. However encrypted D_A is replace by 2 bits depend on the form of D_A.If 16 such D_A are consider in 64 bits, then 16 replacement operations are requires**(R_2).**

Hence $C_S = 16 (R_2)$ = 16

Average value of the Compression Cost (AC)

$= C + C_S / 2$

$= 240 + 16 / 2 = 128$

Thus, AC = 128

The average value of computational cost of the algorithm is

$ACC = P_S + K_S + AE + AC$
$= 18 + 18 + 40 + 128$
$= 204$ **operations.**

The another approach handled by the algorithm for Similar bits not only reduce the average computational cost but also it provide another level of security as the cipher text is the mixing of the code provided by both these approaches. Because of this fact it is too much difficult to guess the plaintext by the adversary.

IV. RESULT & DISCUSSION

The HDPS algorithm is tested on the basis of the compression performance . MATLAB 2010a Simulation tool used for the testing of JPEG 2000 standard Leena test image. The results of HDPS are compared with the Optimized Multiple Huffman Table (OMHT) .The comparison of the result & Leena image is given as below.

Fig 1. Leena JPEG 2000 Test Image

ALGORITHM	HDPS	OMHT
DATA FILE SIZE (BYTES)	145254	145254
COMPRESSION RATIO	26.199	24.08
PSNR	28.287	27.92

Fig 2. Result Comparision

The above result showe that the Compression mechanism used by the HDPS is superior over OMHT algorithm. This is because of the fact that in HDPS image is compressed by taking into account the codebook values .However for OMHT image is compressed by converting it into single vector.

The another result & comparision is based on the Key set up cost & Encryption cost.

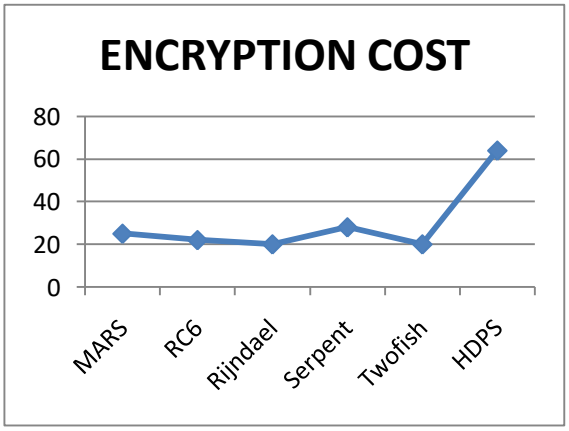

Fig 3. Encryption Cost

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

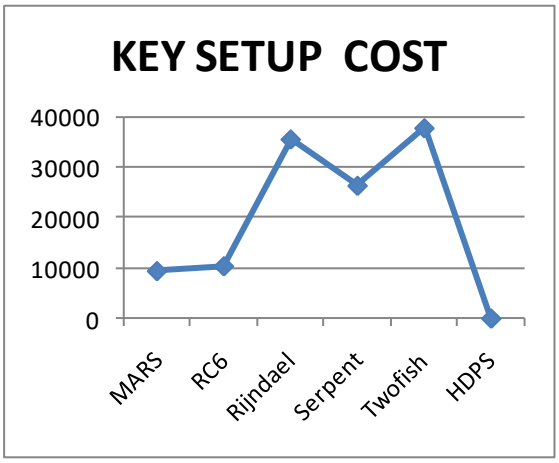

Fig 4 Key Setup Cost

From Fig 3, it is clear that Encryption cost of the HDPS algorithm is high as[37] compare to other algorithms. But this high value of the encryption cost is compensated by the Key Setup cost as shown in fig 4. Hence overall average computational cost is very less with scarifying the compression performance as well as security.

V.CONCLUSION

Hamming Distance Polygram Substitution (HDPS) algorithm based on joint selective encryption and compression approach proves superior in computational complexity as well as compression performance over standard algorithms.

VI. ACKNOWLEDGMENT

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 respect more than I can say to my caring & loving Guru Sant Shri Achyut Maharaj, and my Parents for their inspiral support.

REFERENCES

1. D. Terry and D. Swinehart, "Managing stored voice in the etherphonesystem," *ACM Trans. Comput. Syst.*, vol. 6, no. 1, pp. 3–27, Feb. 1988.
2. L. Diez-Del-Rio *et al.*, "Secure speech and data communication over the public switching telephone network," in *Proc. ICASSP-94*, Apr. 1994.
3. J. Meyer and F. Gadegast, "Security Mechanisms for Multimedia Datawith the Example MPEG-1 Video," Technical Univ. Berlin, Berlin, Germany, Project Description of SECMPEG, May 1995.
4. Intel Netstructure 7110 E-Commerce Accelerator Fact Sheet [Online].Available: www.intel.com
5. M. Droogenbroeck and R. Benedett, "Techniques for a selective encryption of uncompressed and compressed images," in *Proc. ACIVS*, Ghent, Belgium, Sep. 2002.
6. H. Beker and F. Piper, *Secure Speech Communications*. New York: Academic, 1985.
7. B. Goldburg, S. Sridharan, and E. Dawson, "Design and crypt-analysis of transform-based analog speech scramblers," *IEEE J. Select. Areas Commun.*, vol. 11, no. 5, p. 735, Jun. 1993.
8. S. Sridharan, E. Dawson, and B. Goldburg, "Fast Fourier transform based speech encryption system," in *Commun., Speech and Vision, IEE Proc. I*, vol. 138, 1991, p. 215.
9. C. Kuo and M. Chen, "A new signal encryption technique and its attack study," in *Proc. IEEE Int. Carnahan Conf. Security Technology*, Oct. 1991, p. 149.
10. E. Dawson, "Design of a discrete cosine transform based speech scrambler," *Electron. Lett.*, vol. 27, no. 7, p. 613, Mar. 1991.
11. F. Ma *et al.*, "Wavelet transform-based analogue speech scrambling scheme," *IEEE Electron. Lett.*, vol. 32, no. 8, p. 719, Apr. 1996.
12. [12] V. Milosevic *et al.*, "Hadamard transform application in speech scrambling," in *Proc. IEEE Int. Conf. Digital Signal Processing*, Jul. 1997.
13. B. Goldburg, S. Sridharan, and E. Dawson, "Cryptanalysis of frequency domain analogue speech scramblers," in *Proc. Inst. Electr. Eng.Commun., Speech Vis.*, vol. 140, Aug. 1993, p. 235.
14. C. Xydeas *et al.*, "Speech scrambling prior to Ipc coding," in *IEE Colloq. Security and Cryptography Application to Radio Systems*, 1994, p. 9/1.
15. W. Zeng and S. Lei, "Efficient frequency domain selective scrambling of digital video," *IEEE Trans. Multimedia*, vol. 5, no. 1, p. 118, Mar. 2003.
16. G. A. Spanos and T. B. Maples, "Performance study of a selective encryption scheme for the security of networked, real-time video," in *Proc. Int. Conf. Computer Commmunication and Networking*, Oct. 1995.
17. "Security for real-time MPEG compressed video in distributed multimedia applications," in *Proc. 15th IEEE Int. Phoenix Conf. Comput. Commun.*, Mar. 1996.
18. I. Agi and L. Gong, "An empirical study of secure MPEG video transmissions,"in *Proc. ISOC-SNDSS'96*, Feb. 1996.
19. L. Qiao and K. Nahrstedt, "A new algorithm for MPEG video encryption," in *Proc. 1st Int. Conf. Imaging Sci., Syst., Technol.*, Jul. 1997.
20. L. Tang, "Methods for encrypting and decrypting MPEG video data efficiently," in *Proc. 4th ACM Int. Conf. Multimedia*, Nov. 1996, p. 219.
21. L. Qiao *et al.*, "Is MPEG encryption by using random list instead of zigzag order secure?," in *Proc. IEEE Int. Symp. Consumer Electronics*, Dec. 1997.
22. L. Qiao and K. Nahrstedt, "Comparison of MPEG encryption algorithms," *Int. J. Comput. and Graph.*, vol. 22, no. 3, Jan. 1998.
23. C. Shi and B. Bhargava, "A fast MPEG video encryption algorithm," in *Proc. 6th ACM Int. Conf. Multimedia*, Sep. 1998
24. C. Shi, S.Wang, and B. Bhargava, "MPEG video encryption in real-time using secret key cryptography," *Proc. PDPTA'99*, vol. 6, p. 2822, Jun.1999.
25. H. Cheng and X. Li, "Partial encryption of compressed images and videos," *IEEE Trans. Signal Process.*, vol. 48, no. 8, p. 2439, Aug. 2000.
26. C.-P. Wu and C.-C. J. Kuo, "Fast encryption methods for audio-visual data confidentiality," in *SPIE Int. Symp. Information Technologies 2000*, vol. 4209, Nov. 2000, p. 284.
27. A. Servetti and J. Martin, "Perception-based partial encryption of compressed speech," *IEEE Trans. Speech Audio Process.*, vol. 10, p. 637, Nov. 2002.
28. "Perception-based selective encryption of g.729 speech," in *Proc. ICASSP 2002*, vol. 1, May 2002, pp. 621–624.
29. A. Barbir, "A methodology for performing secure data compression," in *Proc. 29th Southeastern Symp. System Theory*, Mar. 1997, p. 266.
30. H. Bergen and J. Hogan, "A chosen plaintext attack on an adaptive arithmetic coding compression algorithm," *Comput. Secur.*, vol. 12, p. 157, 1993.
31. J. Cleary *et al.*, "On the insecurity of arithmetic coding," *Comput. And Secur.*, vol. 14, p. 167, 1995.
32. J. Lim, C. Boyd, and E. Dawson, "Cryptanalysis of adaptive arithmetic coding encryption scheme," in *Proc. ACISP*, 1997, pp. 216–227.
33. I. H. Witten and J. G. Cleary, "On the privacy afforded by adaptive text compression," *Comput. and Secur.*, vol. 7, pp. 397–408, 1988.
34. D. Gillman, M. Mohtashemi, and R. Rivest, "On breaking a Huffman code," *IEEE Trans. Inform. Theory*, vol. 42, no. 3, p. 972, May 1996.
35. W. B. Pennebaker and J. L. Mitchell, *JPEG Still Image Data CompressionStandard*. New York: Van Nostrand Reinhold, 1993.
36. C.-P.Wu and C.-C. J. Kuo, "Efficient multimedia encryption via entropy codec design," in *Proc. SPIE Int. Symp. Electronic Imaging 2001*, vol. 4314, Jan. 2001, p. 128.
37. J. Nechvatal *et al.*, "Report on the Development of the Advanced Encryption Standard," National Institute of Standards and Technology, U.S. Dept. Commerce, Tech. Rep., Oct. 2000.
38. E. Filiol and C. Fontain, "A new ultrafast stream cipher design: COSciphers," in *Proc. 8th IMA Conf. Cryptography and Coding*, Dec. 2001.//www.halcyon.com/pub/journals/21ps03-vidmar

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

AUTHOR PROFILE

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

Dr. 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 SGBAU Amravati University, Amravati (India). His areas of interest are Digital Signal Processing, VLSI Design and Embedded Systems.

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)