International Journal on Information Sciences and Computing, Vol.2, No.1, July 2008

A NEW SECURITY ON NEURAL CRYPTOGRAPHY WITH QUERIES EMPLOYED TO CRYPTO-COMPRESSION OF MEDICAL IMAGES IN TELEMEDICINE SYSTEM

Prabakaran N.[1], Velu C.M.[2], Vivekanandan P.[3]
[1]Department of Mathematics, Anna University, Chennai, India
E-mail : [1]prabakaran_om@yahoo.com, [2]vivek@annauniv.edu

Abstract

There is a necessity to protect the confidential medical images data from an unauthorized access when the exchange of medical information is taken place among the patients and doctors. We can generate a common secret key using neural cryptography, which is based on synchronization of Tree Parity Machines (TPMs) by mutual learning. In the proposed TPMs replacing random inputs with queries are considered, which depend on the current state of the neural network. Then, TPMs hidden layer of each output vectors are compared. That is, the output vectors of hidden unit using Hebbian learning rule, left-dynamic hidden unit using Random walk learning rule and right-dynamic hidden unit using Anti-Hebbian learning rule are compared. Among the compared values, one of the best values is received by the output layer. The medical image is encrypted using Rijndael Encryption and compressed using Huffman Coding, in order to fully dissimulate the visual information of the medical image, which is produced as Crypto-Compressed and Encrypted Medical Images (CCEMI). A network with queries generates a secret key, which can be used to encrypt and decrypt of medical images. We have shown that it is more difficult to break a secret key by brute force attack.

Key words: Neural Cryptography, Medical Images, Rijndael algorithm, Crypto-Compression.

I. INTRODUCTION

The process of neural synchronization is driven by the sequence of input vectors, which are really used to adjust the weights of the TPMs according to the learning rule. As these are selected by the partners participating in the key exchange, A and B have an important advantage over E, who can only listen to their communication. Up to now the partners just avoid repulsive steps by skipping some of the randomly generated input vectors.However, they can use their advantage in a better way. For this purpose the random inputs are replaced by queries. That is A and B choose alternately according to their own weight vectors. In fact, the partners ask each other questions and learn only the answers, on which they reach an agreement [10].

Telemedicine represents a valuable resource for delivering health-related services to remote, suburban areas, providing greater access to health care for consumer and health professionals. The real-time telemedicine is called 'store and forward', in which medical image is sent to a provider at a distant site for their evaluation. This does not allow for a dialogue between the patient and doctor [15].

The medical images are transmitted by network, security of medical images and reducing the size of the medical images is important. This can be done in the compression and encryption technique. The Rijndael encryption and compression constitute a pair that is difficulty reconcilable due to crypto-compression, which is to increase the weight of the file and to decrease the image quality [12]. The Rijndael encryption is symmetric cipher model since the same key (256-bit) is used for encryption and decryption of the Discrete Cosine Transform (DCT) coefficients. It is produced as Crypto-Compressed and Encrypted Medical Image (CCEMI).

This paper is organized as follows. In Section 2, the basic algorithm for neural synchronization with queries is given and definition of the order parameters of proposed TPMs are explained. In Section 3, the generation of queries is described. Overview of medical images in telemedicine is briefly explained in Section 4. The crypto-compression technique and Rijndael encryption are presented in Section 5. In Section 6, the security of medical image is discussed. Finally, conclusion is shown in Section 7.

II. NEURAL SYNCHRONIZATION

The weight vectors of the two neural networks begin with random numbers, which is generated by Pseudo-Random Number Generator (PRNG). In this network random inputs are replaced by queries. That is A and B choose alternatively according to their own weight vectors. The partners A and B receive a common input vector at each time, their outputs are calculated and then communicated. If they agree on the mapping between the current input and the output, their weights are updated according to the learning rule.

A. A Structure of Tree Parity Machine

The TPMs consist of K-hidden units, Y-left dynamic hidden units [1] and Z-right dynamic hidden units [2], each of them being a perceptron with an N-dimensional weight vectors **w.**

The network structure of this TPM is shown in Fig.1. The components of the input vectors **x** are binary.

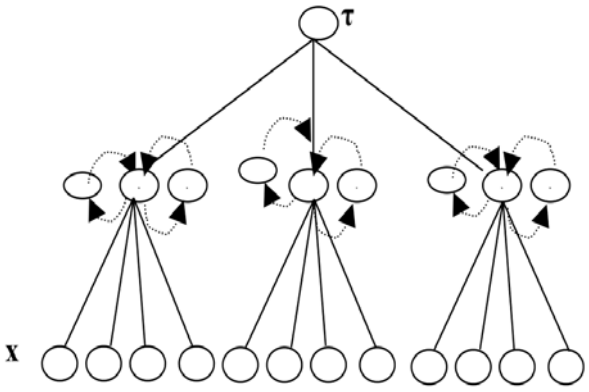

Fig .1 A structure of Tree Parity Machine with K=3, Y=3, Z=3 and N=4.

$x_{ij} \in \{-1,+1\}, x_{im} \in \{-1,+1\}, x_{ik} \in \{-1,+1\}$ (1)

and the weights are discrete numbers between -L and +L

$w_{ij} \in \{-L,-L+1,\ldots,L-1,L\}, w_{im} \in \{-L,-L+1,\ldots, L-1,L\},$
$w_{ik} \in \{-L,-L+1,..,L-1,L\}.$ (2)

where L is the depths of the weights of the networks.

The TPM reads the input vectors using queries. These input vectors are correlated with the present weight vector. $w_k(t)$ At odd time steps, the partner A generates an input vector which has a certain overlap to its weights. w_k^A. At even time steps, the partner B generates an input vector which has a certain overlap to its weights. w_k^B. It is based on the queries to improve the security of the systems.

The index i = 1,.....,K denotes the i^{th} hidden unit of TPM, m=1,.....,Y left-dynamic hidden unit of the TPM, k=1,.....,Z right-dynamic hidden unit of the TPM and j =1,....,N denotes the N components.

The different transfer functions for hidden layer are given below

$$\sigma_i = \text{sign}\left(\sum_{j=1}^{N} w_{ij} \bullet x_{ij}\right) \quad (3)$$

$$\delta_i = \tanh\left(\sum_{m=1}^{N} w_{im} \bullet x_{im}\right) \quad (4)$$

$$\Upsilon_i = \arctan\left(\sum_{k=1}^{N} w_{ik} \bullet x_{ik}\right) \quad (5)$$

where equation (3) is the transfer function of the hidden unit [5], the equation (4) the transfer function of the left-dynamic hidden unit and the equation (5) the transfer function of the right-dynamic hidden unit.

The K-hidden units of σ_i, Y-left dynamic hidden units of δ_i and Z-right dynamic hidden units of δ_i define common output bits of hidden layer of the network and are given by

$$\beta_a = \prod_{i=1}^{K} \sigma_i \quad (6)$$

$$\beta_b = \prod_{i=1}^{Y} \delta_i \quad (7)$$

$$\beta_c = \prod_{i=1}^{Z} \Upsilon_i \quad (8)$$

where equation (6) is the output for the hidden units, equation (7) the output for the left-dynamic hidden units and equation (8) the output for the right-dynamic hidden units.

The two TPMs compare the hidden layer's output bits and then update the weights from hidden units, left-dynamic hidden units and right-dynamic hidden units as well as partners A and B that are trying to synchronize their weights.

$$\psi_i^{A,B} = comp(\beta_a, \beta_b, \beta_c) \quad (9)$$

$$\phi_i^A = w_{ij}^A \, x_{ij}^A \, \tau^B \psi_i^A \quad (10)$$

$$\phi_i^B = w_{ij}^B \, x_{ij}^B \, \tau^A \psi_i^B \quad (11)$$

where equation (9) represents comparison of the output of hidden, left dynamic and right dynamic hidden units of A and B. The equation (10) and (11) represent output of hidden, left and right dynamic hidden units of A and B respectively.

$$\tau = \prod_{i=1}^{K} \phi_i \quad (12)$$

The equation (12) represents the output vector of the output unit of the TPM.

B. Learning Rules

The two TPMs are trained by their mutual output bits τ^A and τ^B using queries as well as receive common input vectors x_i and corresponding output bit t of its partner at each training steps. The following are the learning rules:

(I) If the output bits are different, $\tau^A \neq \tau^B$, nothing is changed.

(ii) If $\tau^A = \tau^B = \tau$, the hidden, left and right- dynamic hidden units are trained by queries, which have an output bit identical to the common output $\phi_i^{A/B} = \tau^{A/B}$,

(iii) To adjust the weights, we consider three different learning rules. They are

(a) Hebbian Learning rule for hidden units

$$w_i^A(t+1) = w_i^A(t) + x_i \tau^A \Theta(\tau^A \phi_i^A)\Theta(\tau^A \tau^B)$$
$$w_i^B(t+1) = w_i^B(t) + x_i \tau^B \Theta(\tau^B \phi_i^B)\Theta(\tau^A \tau^B) \quad (13)$$

(b) Random walk learning for left-dynamic hidden units [6]

$$w_i^A(t+1) = w_i^A(t) + x_i \Theta(\tau^A \phi_i^A)\Theta(\tau^A \tau^B)$$
$$w_i^B(t+1) = w_i^B(t) + x_i \Theta(\tau^B \phi_i^B)\Theta(\tau^A \tau^B) \quad (14)$$

(c) Anti-Hebbian learning for right-dynamic hidden units

$$w_i^A(t+1) = w_i^A(t) - \phi_i x_i \Theta(\tau^A \phi_i^A)\Theta(\tau^A \tau^B)$$
$$w_i^B(t+1) = w_i^B(t) - \phi_i x_i \Theta(\tau^B \phi_i^B)\Theta(\tau^A \tau^B) \quad (15)$$

C. Order Parameters with Queries

The overlap of the weights belong to the i^{th} hidden unit, j^{th} left-dynamic hidden unit and k^{th} right-dynamic hidden unit in the two parties are given below

$$R_i^{A,B} = \frac{W_i^A \cdot W_i^B}{N},\ R_j^{A,B} = \frac{W_j^A \cdot W_j^B}{N},\ R_k^{A,B} = \frac{W_k^A \cdot W_k^B}{N} \quad (16)$$

$$Q_i = \frac{W_i^A \cdot W_i^A}{N}, Q_j = \frac{W_j^A \cdot W_j^A}{N}, Q_k = \frac{W_k^A \cdot W_k^A}{N} \quad (17)$$

The equation (16) represents overlap between two hidden units, two left-dynamic hidden units and two right-dynamic hidden units of A and B respectively [3]. The equation (17) represents weight distribution of hidden units, left-dynamic hidden units and right-dynamic hidden units of A's TPM.

The distance between two corresponding hidden unit, left-dynamic hidden units and right-dynamic hidden units are defined by the (mutual) overlap is given below

$$\rho_{ijk}^{A,B} = \frac{R_i^{A,B}}{\sqrt{Q_i^A Q_i^B}} + \frac{R_j^{A,B}}{\sqrt{Q_j^A Q_j^B}} + \frac{R_k^{A,B}}{\sqrt{Q_k^A Q_k^B}} \quad (18)$$

More precisely, the probability of having different results in the i^{th} hidden unit, j^{th} left-dynamic hidden unit and k^{th} right-dynamic hidden unit of the partners A and B is given by the well-known generalization error for the perceptron [4,9]

$$\varepsilon_p^i = \frac{1}{\pi} \arccos(\rho_{ijk}) \quad (19)$$

where equation (19) represents the generalization error for hidden, left-dynamic and right-dynamic unit of two TPMs.

The quantity is ε_p^i a measure of the distance between the weight vectors of the corresponding hidden units, left-dynamic hidden units and right-dynamic hidden units, these values are independent. The values ε_p^i determine the conditional probability P_r for a repulsive step and P_a for a attractive step between two hidden units, left-dynamic hidden units and right-dynamic hidden units given identical output bits of the two TPMs. In the case of identical distances, $\varepsilon_p^i = \varepsilon$ the values of K, Y and Z ?are found as K=3, Y=3 and Z=3.

$$P_a = \frac{1}{2} \frac{(1-\varepsilon)^9 + 3(1-\varepsilon)^3 \varepsilon^2}{(1-\varepsilon)^9 + 9(1-\varepsilon)^3 \varepsilon^2} \quad (20)$$

$$P_r = \frac{6(1-\varepsilon)^3 \ \varepsilon^2}{3(1-\varepsilon)^9 + 9(1-\varepsilon)^3 \varepsilon^2} \quad (21)$$

The equation (20) and (21) represent probability of attractive and repulsive steps between two hidden units, two left-dynamic hidden units and two right-dynamic hidden units of A and B respectively

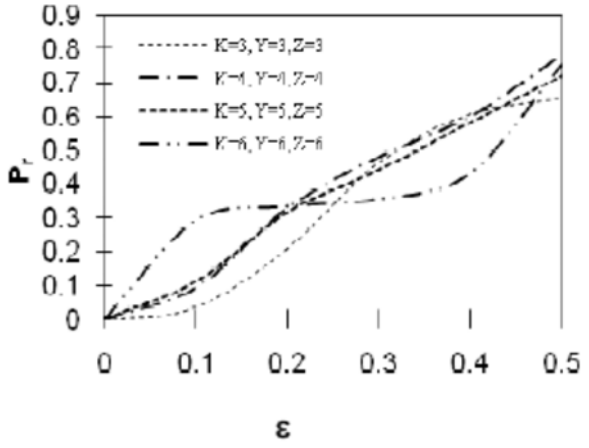

Fig .2 Probability P_r^B of repulsive steps for synchronization with mutual interaction under the condition $\tau^A = \tau^B$.

From the above fig. 2, we are able to determine the probability of repulsive steps occur more frequently in E's TPM than in A's and B's for equal overlap $0 < \rho < 1$. So, the partners A and B have a clear advantage over a simple attacker in neural cryptography. The dotted line shows for P_r^E a simple attack.

The attacker E can assign a confidence level to each output σ_i^E, δ_i^E and γ_i^E of its hidden units, left-dynamic hidden units and right-dynamic hidden units. For this task the local field [7, 9] is given by

$$h_{ijk} = \frac{w_i \cdot x_i}{\sqrt{N}} + \frac{w_j \cdot x_j}{\sqrt{N}} + \frac{w_k \cdot x_k}{\sqrt{N}} \qquad (22)$$

where equation (22) represents the local field of hidden unit, left-dynamic and right-dynamic hidden units of an attacker's TPM.

Then the prediction error, the probability of different output bits for an input vectors 'x' inducing a local field h_{ijk} is given below

$$\varepsilon(\rho_{ijk}, h_{ijk}) = \frac{1}{2}\left[1 - erf\left(\frac{\rho_{ijk}}{\sqrt{2(1-\rho_{ijk}^{\ 2})}} \frac{|h_{ijk}|}{\sqrt{Q_i}}\right)\right] \qquad (23)$$

where equation (23) represents prediction error of the local field of hidden units, left-dynamic and right-dynamic hidden units of an attacker's TPM.

III. GENERATION OF QUERIES

As both inputs $x_{i,m}$ and weights $w_{i,m}$ are discrete, there are only (2L+1) possible results for the product $x_{i,m} \times w_{i,m}$. Therefore, a set of input vectors consisting of all permutation, which do not change h_i, can be described by counting the number $c_{i,l}$ of products with $x_{i,m} \times w_{i,m} = l$.

$$h_{ijk} = \frac{1}{\sqrt{N}}\left[\sum_{l=1}^{L}\left(l(c_{i,l} - c_{i,-l}) + l(c_{j,l} - c_{j,-l}) + l(c_{k,l} - c_{k,-l})\right)\right] \qquad (24)$$

where equation (24) is the number of inputs and weights in the local field of TPM.

But the sum $n_{ijk,l} = c_{i,j} + c_{j-l,} + c_{i,l} + c_{j-l} + c_{k,l} + c_{k,-l}$ is equal to the number of weights with $|w_{i,j}| = |l|$ and thus independent of 'x'. Consequently, one can write h_{ijk} as a function of only L variables, because the generation of queries cannot change 'w'.

$$h_{ijk} = \frac{1}{\sqrt{N}}\left[\sum_{l=1}^{L}\left(l(2c_{i,l} - n_{i,l}) + l(2c_{j,l} - n_{j,l}) + l(2c_{k,l} - n_{k,l})\right)\right] \qquad (25)$$

where equation (25) is the inputs and sum of current weights vectors of the local field of two TPMs.

The inputs 'x' is associated with zero weights, are chosen randomly, because they do not influence the local field. The other input bits $x_{i,m}$ are divided into L groups according to the absolute value $l = |w_{i,m}|$ of their corresponding weight. In each group, $c_{i,l}$ inputs are selected randomly and set to $x_{i,m} = \text{sign}(w_{i,m})$. The remaining $n_{i,j} - c_{i,j}$ input bits are set to $x_{i,m} = -\text{sign}(w_{i,m})$.

The maximum possibilities of the weight vectors of an attacker's TPM is given by

$$l_{\max} = (4L + 2)^{(K+Y+Z)\bullet N} \qquad (26)$$

Then $$\ln(l_{\max}) = (K + Y + Z) \cdot N \ \ln(4L+2) \qquad (27)$$

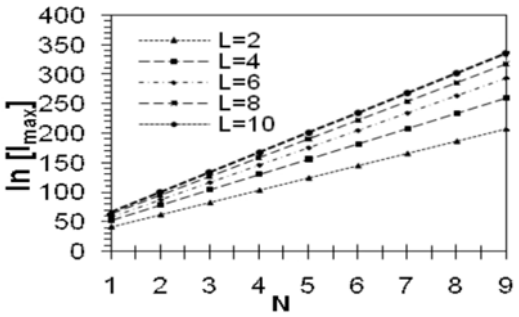

Fig .3 The possible values of l_{max} of the weight vectors of an attack's TPM.

From the above fig. 3, we are able to predict a large number of possible weight vectors, in which weights are chosen by queries. In each time step, either A or B generates the input vectors. The attacker E cannot gain the useful information from analyzing queries.

IV. OVERVIEW OF MEDICAL IMAGES IN TELEMEDICINE

The use of medical images and tele-communication technologies to support long- distance clinical health care, patient and professional health-related education, public health and health administration are explained [15]. Telemedicine represents a valuable resource for delivering the health-related services to remote, sub-urban areas, providing greater access to health care for consumers and health professionals.Telemedicine mainly uses video conferencing equipment. This is an interactive technology and enables patients and health care providers at distant sites to interact 'face-to-face'. Technological advances now allow for these interactions to occur using a desktop computer. An alternative to real-time telemedicine is called 'store and forward', in which medical

image is sent to a provider at a distant site for their evaluation.

V. CRYPTO-COMPRESSION OF MEDICAL IMAGE

The medical image is partitioned into 8x8 blocks, shifted form unsigned integers to signed integer and input to the Forward DCT (FDCT). At the output from the decoder, the inverse DCT outputs 8x8 sample blocks to form the reconstructed image. Then, each DCT coefficients is divided by its corresponding constant in a standard quantization table and followed by rounding to the nearest integer. This output value is normalized by the quantizer step size [13]. Dequantization is the inverse function, which is the normalization, is removed by multiplying by the step size, which returns the result to a representation appropriate for input to the IDCT. All of the quantized coefficients are ordered into the 'zig-zag' sequence. This ordering helps to facilitate entropy coding by placing low-frequency coefficients before high-frequency coefficients. The medical images, which specifies entropy coding method is Huffman coding. The 2-step process of entropy encoding converts the zig-zag sequence of quantized coefficients into an intermediate sequence of symbols and converts the symbols to a data stream in which the symbols no longer have externally identifiable boundaries.

In each block the 64 DCT coefficients are set up from the lowest (upper left corner) to the highest frequencies (lower right corner). The most important visual characteristics of the image are placed in the low frequencies while the details are situated in the higher frequencies. The HVS (Human Visual system) is most sensitive to lower frequencies than to higher ones [12, 14].

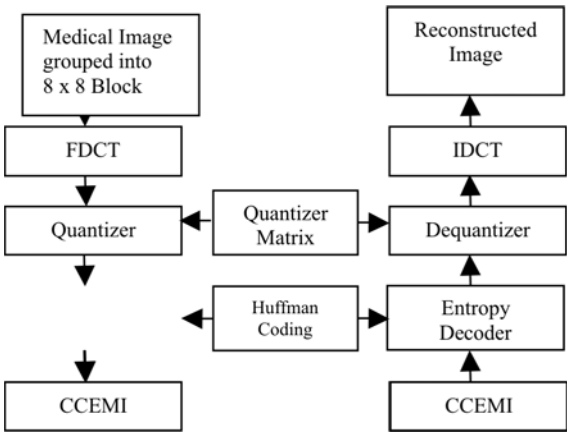

Fig .4 DCT-Based Encoder and Decoder processing steps of CCEMI

The Rijndael takes the data block from the FDCT coefficients as input and performs several transformations for encryption and decryption. These are represented as two-dimensional array of bytes. The total number of rounds is 24 for encryption and decryption of data block using symmetric secret key which is generated by two TPMs. To encrypt a block of data in AES, we first perform an Add Round Key step (XORing a sub-key with the block) by itself. There are regular rounds that involve 4 steps

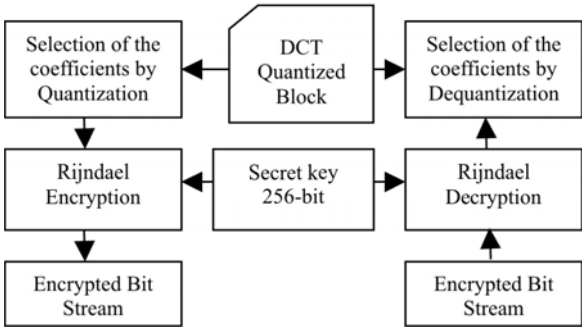

Fig .5 Rijndael Encryption and Decryption processing steps of medical images

First step, The ByteSub step, where each byte of the block is replaced by its substitution in an S-Box. Second step, the ShiftRow step, considering the block to be made up of bytes 1 to 16, these bytes are arranged in a rectangle and shifted. Third step the MixColumn step, a matrix multiplication is performed; each column is multiplied by the matrix. Finally the Add Round Key, this is simply XOR in the sub-key for the current round.

VI. THE SECURITY OF MEDICAL IMAGES

Here we find that queries enhance the security of the neural key-exchange protocol a lot for given synchronization time. The Crypto-Compressed and Encrypted Medical Images (CCEMI) has more secured during the transition due to crypto-compression technique using Huffman coding with Rijndael algorithm. Rijndael uses S-boxes as non-linear components. A secret key is 256-bit which offer a sufficiently 1.16×10^{77} of possible keys. The attacker will take 10^{56} years to break the secret key using brute force attack.

VII. CONCLUSION

In the proposed TPMs, the synchronize time of the attacker is increased by the three transfer functions in the hidden unit using Hebbian learning rule, left-dynamic hidden unit using Random walk rule and right-dynamic hidden unit using Anti-Hebbian learning rule included queries in the training process of the TPMs. The queries increase the probability of repulsive steps for the attacker during the synchronization. In addition, the method obtains a new parameter, which can be adjusted to give optimal security. The CCEMI reduces the time for transmission and the space on disk. A 256-bit secret key is generated by two TPMs and it is used in Rijndael

algorithm. For the attacker to find all possibilities of secret key, it will take trillion years against the brute force attack.

ACKNOWLEDGEMENT

We thank **Dr. J.M. Ashok Kumar**, Cardiologist, Apollo Hospitals, Chennai for his valuable suggestions and comments. Also we thank **Dr. C.S. Vijay Shankar**, Cardiovascular and Thoracic Surgery and **Dr. Ganapathy**, Neuro Surgeon Telemedicine section for his valuable feedback.

REFERENCES

[1] N. Prabakaran, P. Loganathan and P. Vivekanandan, 2008, Neural cryptography with multiple transfer function and multiple learning rule, International Journal of Soft Computing, vol. 3 (3), pp.177-181.

[2] N. Prabakaran, P. Karuppuchamy and P. Vivekanandan,2008 A new approach on neural cryptography with dynamic and spy units using multiple transfer functions and learning rules. Asian Journal of Information Technology, Vol. 6 (7).

[3] N. Prabakaran, P. Saravanan and P.Vivekanandan,2008, A new technique on neural cryptography with securing of electronic medical records in telemedicine system, International Journal of Soft Computing, vol. 3 (5), pp. 390-396.

[4] A. Engel and C. Van Den Broeck,2001, Statistical mechanics of learning (Cambridge University Press, Cambridge)

[5] W. Kinzel and I. Kanter, Interacting neural networks and cryptography, 2002, Advances in Solid State Physics, by B. Kramer (Springer, Berlin) Vol. 42, pp.383-391 [cond-mat/0203011]

[6] A. Ruttor, W. Kinzel, L. Shacham and I. Kanter,2004, Neural cryptography with feedback, Physical Review E, vol. 69, pp.1- 8.

[7] A. Ruttor, W.Kinzel and I.kanter, Neural cryptography with queries,2002, In: Physics Rev. E. Vol. 25, No. 01, pp.01-12.

[8] A. Ruttor, I. Kanter and W. Kinzel, 2006, Dynamics of neural cryptography. [cont-mat/061257].

[9] A. Ruttor, I. Kanter, R. Naeh and W. Kinzel,2006, Genetic attack on neural cryptography. [cond-mat/0512022v2]

[10] A. Ruttor, 2006, Neural synchronization and cryptography, Ph. D. Thesis.

[11] A.J.Penrose, Neil A., Dodgson, Extending lossless image compression, Eurographics UK'99, 13-15 Apr 1999, Fitzwilliam College. Cambridge.

[12] Jean-Claude Borie, William Puech and Michel Dumas, Crypto-compression system for secure transfer of medical images, University of Montpeillier II, France.

[13] Gergory K. Wallace, The JPEG still picture compression standard, Apr 1991, ACM.

[14] W. Puech et. J.M. Rodrigues. Crypto-Compression of medical images by selective encryption of DCT, University of Motpeillier II, France.

[15] http://www.utahtelehealth.net/faqs.html, Utah telehealth network, Telemedicine FAQ's, University of Utah Health Sciences Center.

Mr. Prabakaran .N Faculty- Department of Mathematics - Anna University. He has published four papers in international journals and six papers in national and international conferences. His areas of interest are Network Security and Wireless Technology.