International Journal of Computer Applications (0975 – 8887)
Volume 114 – No. 2, March 2015

Application of Predictive Coding in Neuroevolution

Heman Mohabeer
Department of Electrical and Electronics Engineering
University of Mauritius
Reduit

K.M. Sunjiv Soyjaudah
Department of Electrical and Electronics Engineering
University of Mauritius
Reduit

ABSTRACT

This paper presents promising results achieved by applying a new coding scheme based on predictive coding to neuroevolution. The technique proposed exploits the ability of a bit, which contains sufficient information, to represent its neighboring bits. In this way, a single bit represents not only its own information, but also that of its neighborhood. Moreover, whenever there is a change in bit representation, it is determined by a threshold value that determine the point at which the change in information is significant. The main contributions of this work are the following: (i) the ratio of the number of bits to the amount of information content is reduced; (ii) the complexity of the overall system is reduced as there is lesser amount of bit to process; (iii) Finally, we successfully apply the coding scheme to NEAT, which is used as a biometric classifier for the authentication of keystroke dynamics

General Terms

Artificial intelligence, Neural Network, Neuroevolution.

Keywords

NEAT; Predictive Coding; Biometric; coding scheme

1. INTRODUCTION

Artificial neural network (ANN) consists of a mathematical way to describe the way the human brain function that can perform machine learning, character and pattern recognition. The success of ANN depends much on the coding scheme. Normally, raw data contain impurities that have a major impact when processing information. The coding scheme is responsible for retaining vital information from the raw data. In this way, systems that are developed to process these data can be less complex without any compromise to their efficiency.

Several neural coding strategies have been developed over the years. The aim has been to present ANN with such a coding scheme that would minimize the search space when probing a solution. A small search space results in more precise solution. Neuroevolutionnary techniques such as combination of genetic algorithm (GA) and ANN have partly been contributors to the development of new coding schemes. However, the main reason for the failure of neuroevolution in real life problems is that the latter problems are poorly represented at the abstraction level [1]. This means that the coding schemes used to encode these real life problems are either poor or carry insufficient amount of information, which clearly define them in the state space [2]. Moreover controversy still persists as to how the information is encoded across patterns of activity within a population of neural units in a biological neural network. Normally, a coding scheme favors only one type of distribution. Since the external stimulus to which the system is exposed is not constant, it is impossible to devise which distribution technique has a definite advantage as compared to the other. Previously the neural architecture was arranged in such a way that different location in the architecture represented different distributions [3]. It is argued in [4] that a good coding scheme should have both local and distributed representations based on the response to different stimuli such that no application is subjected to a specialized way of processing. The key strength is expected to reside in allowing both distributions within the system. Another method of deploying both local and distributed representations is based on the response to different stimuli and on the use of the same architecture and locations, that is, the entire ANN instead of using only specialized nodes within the ANN [1]. A round robin method is applied so that the whole system allows both types of processing simultaneously. The latter has a clear advantage as compared to using only a specialized type of distribution since it utilizes the whole architecture and makes the system less complex by reusing the same architecture rather than a different one for each type of representation. However the efficacy of the representation lies on the coding scheme as the information carrying capacity determines which representation gives the best performance for a given problem.

In this paper, a novel system is proposed that makes use of both local and distributed data representations in the ANN in an attempt to emulate the way the human brain processes information. The stimuli to which the ANN is exposed determine the data representation as to whether local or distributed processing should be applied and orient the evolution to render efficient architecture. In this way, the system becomes autonomous so that it is allowed to search for its own efficient architecture. Moreover, the data representation across the structure is also determined by the system during evolution. This leads to a more efficient representation of the data in the state space. Finally, the approach mention above paves way to a new coding scheme based on predictive coding. The aforesaid scheme is inspired by the way the eye transmits minimal information to the human brain while leaving the brain to construct full image. The coding scheme proposed is stimuli dependent and controls both the evolution and the data representation of the architecture. Moreover, the minimum description length (MDL) algorithm [20], which enables efficient predictive coding, is successfully applied to neuroevolution while paying particular attention to the growth of the nodal structure of the ANN during evolution.

The remainder of the paper is structured as follows: in section II, we start with a detailed description of the underlying theories related to this experiment. Next, we provide a detailed description of the data used to conduct the experiments. Following that, we depict the methodology adopted and conduct an exhaustive experimentation to evaluate the efficiency of our proposed system. Finally, a conclusion and future work close the paper.

2. BACKGROUND

Neural coding strategies have a direct impact on the performance of ANN. They are responsible for the complexity of the structure and directly affect the architecture of the ANN when attempting to solve a task. The tuning properties and number of nodes that are active to represent a single stimulus are the two main attributes that are taken into consideration when designing neural coding strategies [5]. However, these two attributes cannot distinguish the difference between local and distributed codes. Rather than identifying a qualitative difference between distinct coding schemes, these properties describe two types of code that may be generated by a single underlying representational mechanism. Hence, the terms ‘‘local’’ and ‘‘distributed’’ are confined to a single coding mechanism, rather than identifying two distinct, mutually exclusive, coding schemes [6]. The following sub-section provides a description of these properties. Figure 1 displays two simple neural networks, each containing five nodes, (nodes are depicted as circles and active nodes are shaded). Each row illustrates the response of the two networks to a different stimulus. The network on the left responds using a local code while the network on the right produces a distributed representation. The following two properties are essential to understand the importance of the methodology proposed in this paper; tuning properties and size of dynamic nodes.

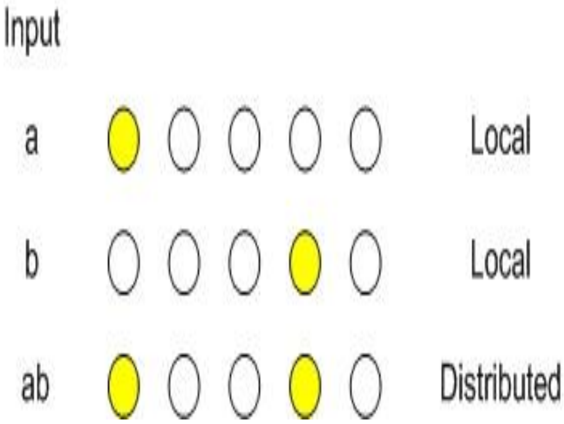

Fig. 1. Example of the representations of local and distributed coding scheme

2.1 Tuning Properties

Tuning properties [6] is an important parameter that controls the effectiveness of the response provided by the neural network to a given stimulus. The weight is a key factor that is directly involved with the tuning properties. The weight is modified for each node with respect to the exposure undergone towards a stimulus to improve its performance. The activation function is a generalization that represents the rate of action potential triggered in a neuron. The action potential is the required limit to enable firing of the neuron in response to a given stimulus. The sigmoid function is one example of activation function that transforms the weighted sum of the nodes of the input activities into nonlinearity [7]. In this way, a network of nodes using this activation function can generate a local code if individual nodes respond most strongly to input patterns. Such a network could also alternatively generate a distributed code if the node responds to sub-features of each stimulus since it is capable of generating either local or distributed codes. Moreover, tuning properties have been claimed as a distinguishing feature of coding schemes [8].

2.2 Size of Dynamic Nodes

Another property which is an inherent feature of coding is the number of nodes that responds to an individual input. A node is rendered active during the lifetime of an ANN system if it represents an input at every instance for which, the whole architecture is exposed to inputs. However, considering the case of a specific input, such input can either be broadcast within the architecture or reside within a specific node depending on the representation local or distributed. The strength of the node upon activation depends on the extent to which it is solicited during its lifetime. If it contributes to the overall performance of the ANN, then for each iteration of the system during training, the strength of that particular node is further reinforced. Moreover, in neuroevolution, the number of active nodes is determined by their fitness value, that is, the fitness value provides evidence for their level of contribution to the overall performance of the system [9]. A local code is characterized as one in which a single node is active, while a distributed code is characterized as one in which more than one node is simultaneously active [10] [11].

2.3 Predictive Coding

Predictive coding (PC) depends on the assumption that a system tries to learn an internal model of the external environment and uses this model to actively predict incoming signals [12] [13]. PC can be used as a framework for removing redundancy in data and thus it enables a coding system which is more proficient in the neural system. It has been postulated in [14] [15] [16] that by allowing the transmission of only the predicted path of an incoming signal, the system is able to reconstruct the full signal merely by making use of the limited dynamic range of neurons. PC is an intrinsic part of the human brain. Inspiration has been drawn from this in order to build robust artificial neural network systems. Equation (1) represents a mathematical model using Bayes theorem to construct the PC coding scheme in ANN. I represent the input of the image, R represents the hidden internal model parameter, P (I) represents a normalizing constant and H is the coding length. In equation (1), the length of H is equivalent to maximizing the posterior probability of the parameters under the PC coding assumption. In this way, the regularized sum square error function is minimized.

$$p\left({}^{r}/_{l}\right) = \frac{p\left(\frac{l}{r}\right)p(r)}{\sum_{i=1}^{n} p(\frac{l}{r})p(r)}$$

$$Length\ of\ H_1 = -\log\ p({}^{l}/_{r})$$

$$Length\ of\ H_2 = -\log\ (r)$$

$$Length\ of\ H = \log\left(\sum\nolimits_{i=1}^{n} p\left(\frac{l}{r}\right)p(r)\right) - \log\left(p\left(\frac{l}{r}\right)\ p(r)\right) + Length\ of\ H_2$$

In order to develop an efficient ANN system which uses predictive coding successfully, the general principle of Minimum Description Length (MDL) has been devised as follows: a table of labeled instance is devised the desired output is then removed. These are replaced with a description of the neural network. This principle has been adopted in the paper.

3. NEAT ALGORITHM

Neuroevolution of Augmented Topologies (NEAT) has been developed by K. Stanley in 2002. The basic concept developed was to start from minimalistic topologies and gradually evolve until the rightmost structure is achieved, which can effectively solve a problem. The idea has been borrowed from the way an organism in nature increased in complexity since the first cell. Hence, NEAT is allowed to find highly sophisticated and complex neural network. Prior to NEAT, a topology was chosen for evolving networks before the experiment was started. The weight space is explored through the crossover and mutation. This concept of fixed topology focused on the weight optimization determined the functionality of a network. Another instance, was to evolve both the weight and the topology at the same time [17]. It was argued whether evolving both weight and topology provided an advantage over a fixed topology. In fact, since a fully connected network could in principle approximates any function [18] it was regarded as wastage of valuable effort permuting over different topologies. Moreover, several technical challenges such as searching for a genetic representation allowed disparate topologies to cross over in a meaningful way [19]. The above process enabled to find a way whereby topological innovation that needed a few generations, be optimized and protected from disappearing from the population prematurely [20] had to be tackled. NEAT was developed to address these challenges. Actually, the advent of NEAT allowed neuroevolution to make one step closer to real life evolution.

3.1 Speciation

Speciation is the evolutionary process by which new biological species arise. This process occurs due to changes in genetic composition of a population with each passing generation. The two most common types of speciation are; allopatric and parapatric. Allopatric speciation, which is the most documented, consists of a single population that has somehow become geographically separated into two populations and then, evolves into two different species [21]. However, should the two populations be reunited they would then be unable to interbreed. Another type of speciation is parapatric speciation [22], in which there is no geographical separation between regions, and nonetheless a population is separated into multiple populations due to the large size of the region. Some interaction between populations still occurs, but overall, they remain separated to the extent that they can eventually no longer interbreed. NEAT allows both allopatric and parapatric speciation to occur so that the population speciates in such a way to facilitate individuals initially compete primarily within their own niche instead of the whole population. As such, topological innovations are protected and much time is devoted in the optimization of their structure before they have to compete to other niches within the population. The philosophy that is promulgated via speciation follows the concept that new ideas must be given the chance to reach their potential instead of being prematurely eliminated. Speciation has also been applied in the cooperative coevolution of modular systems of multiple solutions [17].

3.2 Complexification

Complexification, in the context of neuroevolution, is the gradual increase of a neural network structure by adding new structure without modifying the existing structure. In other word, complexification refers to expanding the dimensionality of the search space while preserving the values of the majority in dimensions. This process is derived from biological genes which duplicates and grows across each generation. Biologically, key innovations are brought forward by gene duplication across the overall body morphology over the course of natural evolution [23]. NEAT tries to capitalize on this property by emulating the process of complexification. It is essential to note that NEAT has not strictly followed the process of biological duplication as mentioned in [24]. Moreover, NEAT tries to expand the length of the size of the genome as in [25] and makes it possible to search a wide range of increasingly complex network topologies that are protected in their own niche respectively.

3.3 Recombination of Historical Marking

The concept of historical marking is fundamental to NEAT. Marking helps to trace individuals throughout each generation. Like so, historical marking allows the recombination of two arbitrary neural networks with different structures. The fundamental process is carried out by scanning the parent structure for the structural elements with matching historical markings. An offspring is built by copying the parent with a higher fitness value. This is followed by a random exchange of internal parameters of all matching nodes and connections with another parent. The little amount of computation required makes historical marking one of the best ways of tracking genes.

4. METHODOLOGY INVOLVED, INCLUDING DATA CAPTURE

To provide evidence of the efficacy of the new coding technique, we have performed the implementation of this scheme in a biometric classification system consisting of keystroke dynamics (KD). Keystroke dynamics (KD) concern the capture of data generated from hand gestures, often when typing via a keyboard. KD is regarded as soft biometrics since it consists of traits instances that are created in a natural way, and used by humans to distinguish their peers. Consistent research has been maintained on KD as it is an efficient way of controlling access from a distance, that is, unlike other biometric systems KD offers the possibility to capture and authenticate live user data without the user being physically present at the location where he/she wishes to gain access.

For this experiment, data have been collected for hundred different users. These data were captured using a toolkit constructed using excel. The toolkit captured the following information: key code, key depression and release and dwell time. The following set of rules was taken into consideration while performing the capture of data:

- Static verification was used.
- Only dwell time was considered.
- Typing proficiency were not a requirement

These sets of rules were carefully observed and a sample of user was chosen at random to initiate the data capture process. The main feature used for the experimentation is the hold time. The hold time refers to the amount of time the user holds on to a key. The user was given the following letter to type: “T, H, U, R, S, D, 1, Y. The typeset was kept small so as to ensure that NE remains fairly complex post complexification and thus could be easily controlled.

The features that were captured consisted of Holds (time taken to hold a key when pressed, that is, key depression and release) and were recorded in a database. The database created a template for each user and this was used for authentication purposes. A key concept that has been introduced in this work prior to allowing NEAT to perform classification is normalization. This scheme is expected to perform well if prior knowledge about the average score and the score variations of the matcher is available [26]. If we do not have any prior knowledge about the nature of the matching algorithm, then we need to estimate the mean and standard deviation of the scores from a given set of matching scores. The normalized scores (S_k') are given by

$$S_k' = (S_k - \mu)/\sigma$$

where S_k is the raw data, μ is the arithmetic mean and σ is the standard deviation of the given data. If the input scores are not Gaussian distributed, z-score normalization does not retain the input distribution at the output. This is due to the fact that mean and standard deviation are the optimal location and scale parameters only for a Gaussian distribution. Post normalization, predictive coding is applied to the data. At the encoder a model template is generated from the initial normalized template using the setup provided below:

```
Encoder (Model, Data)
    for X = 0 to $X_{max}$
        for Y = 0 to $Y_{max}$
            Error [X, Y] = Data [X, Y] – model [X, Y]
```

The decoder, which consists of artificial neural network is trained with the following model:

```
Decoder (model)
    for X = 0 to $X_{max}$
        for Y = 0 to $Y_{max}$
            Data [X, Y] = Model [X, Y] + Error [X, Y]
```

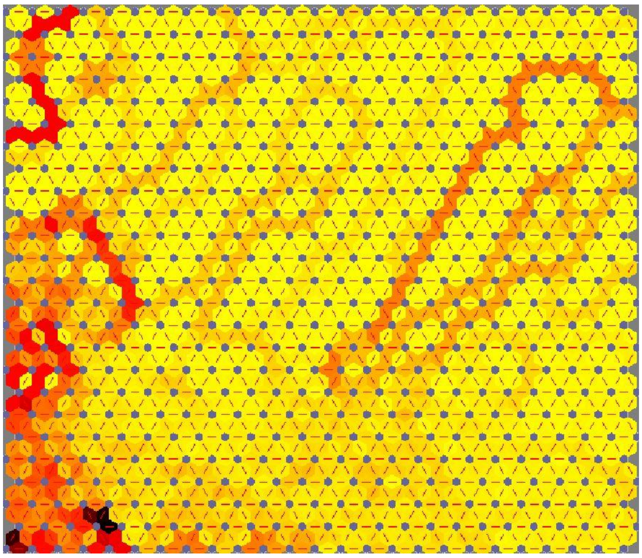

Fig. 3 Cluster of biometric data prior to encoding

Fig. 3 displays the different set of clusters obtained post initial training with NE. The aim is to show that indeed the data collected can be grouped based on certain properties each set have in common. The boundary, which is clearly visible segregates the data within the template. Thus, using the minimum description length (MDL) it becomes much easier to distinguish elite data within each group to represent the cluster. In this way, the data is encoded using predictive coding while the model is used to train the ANN. It is worthwhile to note that there is a definite advantage using this method since the amount of data involved to train the system is relatively few. Hence, the complexity of the ANN is lesser. The clustering of data is an important phase as it undermines the importance of the correlation among data. Accordingly, a single data or a set of data can be defined within each cluster to represent the overall group within a boundary post clustering. NEAT is invariably trained with the model so as to be able to anticipate the flow of the incoming data and predict the outcome/output accordingly. For this experiment, the choice of NEAT has been evident as it enables the system to act as an autonomous agent, thereby inhibiting its own network topology and weight assignment. Moreover, due to the proposed coding scheme employed, NEAT will be rendered more effective as it is expected to have a lower complexity and more dedicated to learning rather than self-adaptation and complexification.

4.1 Applying Predictive Coding to Image

Predictive coding has proven to be effective for lossless image compression. Predictive coding estimates a pixel color value based on the pixel color values of its neighboring pixels. However, they can also be used to estimate the binary value of the neighboring pixels since the change between one pixel to another is rather smooth. To be able to apply predictive coding to image, the following methodology is adopted. The first step includes the conversion of image into binary form. This is rendered possible using the following syntax:

```
I2 = im2uint8 (I1)
RGB2 = im2uint8 (RGB1)
I = im2uint8 (BW)
X2 = im2uint8 (X1,'indexed')
gpuarrayB = im2uint8 (gpuarrayA, ___)
```

The image is converted into 8-bit unsigned integers. Sending image over a transmission channel is more complex due to the high amount of data in representing the image. The addition of noise further renders the deconstruction and reconstruction of image more challenging. When adopting predictive coding to encode this data, the coding scheme allows the possibilities of reconstructing a set of data based on a given estimate. If there is an abrupt change in the pattern of data or the estimate is too high, then another set of estimates is performed during sampling of the data. This is rendered possible due to the clustering of the data that occurs as shown in fig 3.

Fig 4 provides a schematic diagram of the system. X [n] is any data introduced and recovered to and from the system. The predictor at both ends consist of the same. The predictor segregates the data into two parts: the model and the residual. The residual is passed through the coder. On the decoding side, NEAT is used instead of traditional decoder. The predictor which capture the model trains the NEAT so as to be able to recover data X [n] from the residual. To train NEAT 100 iterations were used. The choice for this amount of training is evident since due to the fact that NEAT is autonomous, hence, depending on the complexity achieved to provide optimal results, NEAT were given the freedom to select its own topology and weight. In this way, once the peak complexity is reached NEAT the system stabilizes and the iteration renders no further output as shown in fig 5. It was noted that the number of iterations is dependent on the predictor (the model generated by the predictor was determinant as to the complexity of the network).

5. RESULTS AND DISCUSSION

The results obtained were quite promising as compared to other encoding scheme. In the case of Biometric classification there was significant improvement in the results obtained as compared to other direct encoding scheme. The reason appears to lie on the low complexity achieved by the NEAT algorithm in order to solve the problem. Furthermore, a lower complexity also meant that more nodes were dedicated to incoming bits. The amount of information processed when employing predictive coding is significantly lower for the same output when considering the case of direct encoding. The results of the output from NEAT were computed using the mean square error (MSE).

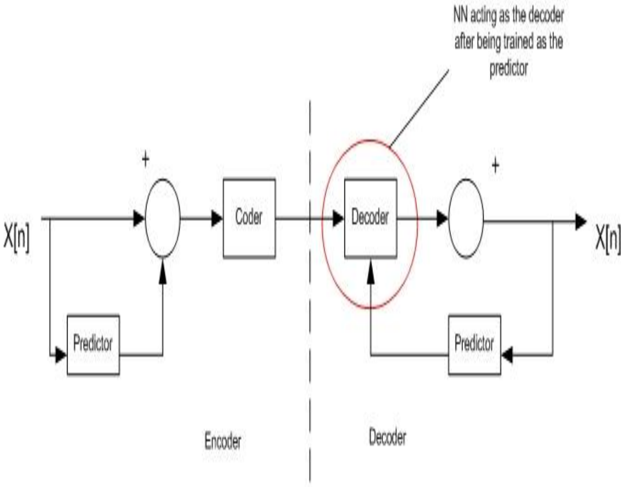

Fig. 4 Modified approach of a typical encoding and decoding system employing predictive coding. The predictor is used to train the decoder rather than the traditional way of acting after the decoder. X [n] is the data input and output.

To understand the concept of the MSE consider the following: If d is the desired output of an output neuron and y is the actual output of the neuron, the square error is (d - y) squared. If two output neurons exist, then the mean square error for these two neurons is the average of the two square errors. Table 1 compares the MSE for each user computed after 100 training. These values represent the average of ten MSE values obtained per hundred trainings for each user, which means that overall each user was trained 1000 times. The error is marginal as compared to using NEAT via direct encoding for each user. The MSE for two particular users were noted to have a rather higher divergence. Investigation of data for user 2 revealed that the data were highly redundant while for the data captured for user 8 had very little redundancy. As such, it could be concluded that user 2 had a higher typing proficiency than user 8.

Fig. 5 compares the efficiency of a typical user (User 1) over hundred when employed with two different types of coding: predictive coding and direct coding. It clearly displays that the efficacy of predictive coding is superior as compared to direct coding. It is noted that during the first twenty training direct coding gives a much better efficiency than predictive coding. However, this can be easily explained based on the fact that predictive coding consist of a minimal amount of data (only the residual part). Hence, initially much focus is devoted into complexifying NEAT rather than attempting to solve the data. Consequently, after twenty training the efficiency significantly increases.

Table 1 MSE of each user post Training of Classifier

User	*MSE (predictive coding)*	*MSE (NEAT)*
User 1	0.0041	0. 0173
User 2	0.0038	0.4640
User 3	0.0052	0.0321
User 4	0.0010	0.0794
User 5	0.0015	0.0664
User 6	0.0027	0.0418
User 7	0.0034	0.0746
User 8	0.0017	0.0026
User 9	0.008	0.0338
User 10	0.0049	0.0401

The second simulation was carried out using image compression. Since the redundancy in the image is much higher the residual data extracted is minimalistic. The compression ratio achieved by extracting the model data to residual data is approximately 1:75, which is higher than most compressors available. Post training of NEAT using 100 training the recovery of the image was roughly about 95 %. Moreover, when viewing the image after decompression it was impossible to depict the difference between the initial images to that of the decoded one as shown in fig. 6. In fact, it can be postulated that a trivial amount of noise could have been introduced in the residual data.

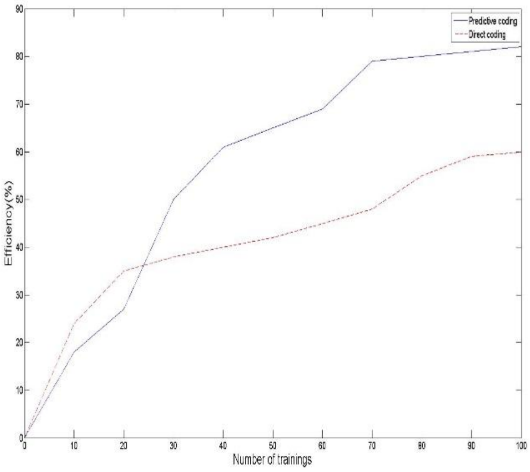

Fig. 5 A comparison of the performance of NEAT after 100 training for user 1 (see table 1) using predictive coding and direct coding

The residual data conveyed via a residual coefficient undermines the noise factor introduced during the processing of the image. A lower residual coefficient results in more noise being introduced. However, a high coefficient might result in high complexity of the network due to insufficient modelling of the data. Fig. 6 displays noise filter amplitude for three different residual coefficients. A higher coefficient results in lower noise, thus rendering a smoother picture. The upper left (residual coefficient = 0.1) picture displays the signal amplitude of the image reconstruction by the NEAT algorithm. Note the spikes present, which demonstrates the image rendering to be fuzzy. The upper right consists of a residual coefficient of 0.3 while the lower middle consists of a residual coefficient of 0.6.

Fig. 6 Original image on the left side while decoded image on the right side. It is almost impossible to distinguish the difference between the two images after 100 trainings.

6. CONCLUSION AND FUTURE WORK

It can be concluded that predictive coding is an efficient means of coding data for NEAT. In fact, the main advantages of predictive coding as compared to the other coding scheme has been the reduction of complexity. NEAT, works like an autonomous agent searching for the right balance between the weight and the topology for a given stimulus. As such, these parameters depend on the amount of data and their redundancy that need to be processed. A higher redundancy signifies a low information containment and thus the topology of NEAT waste lots of nodes in processing these redundant information. This in turns renders the overall system slow and inefficient. Predictive coding offers a simple approach by building a model out of the data input and uses this to train the network. The model contains only vital information that helps NEAT to reconstruct the information after reception of the residual data. In this way, the performance of NEAT is greatly improved.

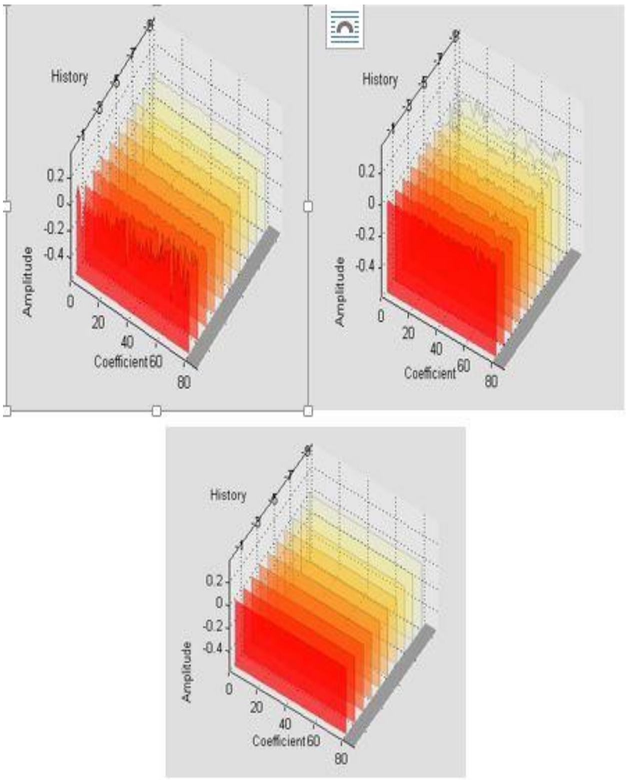

Fig. 7 Reconstruction of the image by NEAT (displayed as signal amplitude

It would be interesting to apply predictive coding to real world data as predictive coding provide a mean of reducing the complexity of such data. Furthermore, the methodology employed below might pave new ways of coding data for deep learning, which consist of a complex ANN with multiple hidden nodes. Finally, it would be interesting to observe the behavior of predictive coding in those areas where direct and indirect encoding were previously adopted. As such, new evolutionary techniques could be built around this coding scheme.

7. ACKNOWLEDGMENTS

The financial support of the Tertiary Education of Mauritius is gratefully acknowledged.

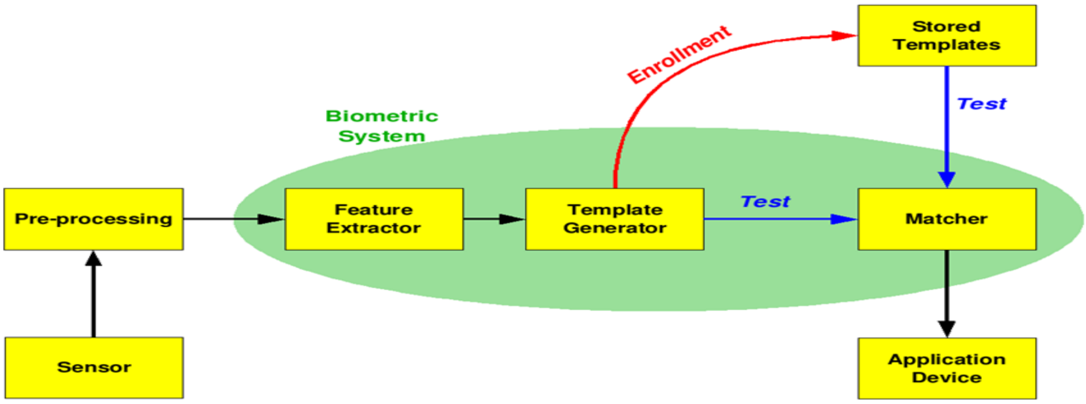

Fig. 2 Schematic diagram of a typical biometric system

8. REFERENCES

[1] Mohabeer H, Soyjaudah K.M.S 2012 Application of Predictive Coding in the Evolution of Artificial Neural Network. IEEE 3rd International conference on Cognitive Infocommunication (CogInfoCom), 775-780.

[2] Coates A, Ng A.Y 2011 The importance of Encoding versus Training with Sparse Coding and Vector Quantization. Proc. Of the 28th International Conference on Machine Learning, Bellevue, 921- 928.

[3] Ahmadian K, Gavrilona M 2012 Chaotic Neural Network for Biometric Pattern Recognition. Advances in Artificial Intelligence, Vol. 11.

[4] Rogers T. T, McClelland J. L 2003 Semantic Cognition: A Parallel Distributed Processing Approach. MIT press.

[5] Wilke Stephan D 2001 Neural Coding of Dynamic Stimuli. Proc. of the International Conference on Artificial Neural Networks. 1081-1086.

[6] Spratling M.W, Johnson M. H 2004 Neural coding strategies and Mechanisms of Competition. Cognitive System Research, Vol. 5(2), 93-117.

[7] Cybento G 1989 Approximation by superposition of a sigmoidal function. Math. Control Signals Systems, 303-314.

[8] Ducker D. M, Kerr W. T, Aguirre G. K 2009 Distinguishing Conjoint and Independent Neural Tuning for Stimulus Features with fMRI Adaptation. J. Neurophysiol, Vol. 101(6). 3310-3324.

[9] Silva F, Urbano P, Oliviera S, Christensen A. L 2012 odNEAT: An algorithm for Distributed Online, Onboard Evolution of Robot Behaviors. MIT press.

[10] Thorpe S. J, 1995 Localized versus distributed representations. In M. A. Arbib (Ed.) The handbook of brain theory and Neural Networks. Cambridge, MA: MIT press, 549- 552.

[11] Page M 2000 Connectionist modelling in psychology: a local manifesto. Behavioral and Brain Sciences. Vol. 23(4), 443-467.

[12] Frsiton K, Kiebel S 2009 Cortical Circuits for Perpetual Interference. Neural network. Vol. 22, pp 1093-1104.

[13] Brown E. C, Brune M 2001 The Role of Prediction in Social neuroscience. Frontiers in Human Neuroscience.

[14] Huang Y, Rao R. P. N 2011 Predictive coding. WIREs cognitive science, vol. 2(5).

[15] Bubic A, Cramon D. Y. V, Schubotz R. I 2010 Prediction, Cognition and the Brain. Frontiers in Human Neuroscience. vol. 4 (25).

[16] Jehee J. F. M, Ballard D. H 2009 Predictive Feedback can Account for Biphasic Responses in the Lateral Geniculate Nucleus. Computational Biology. Vol. 5(5).

[17] Yao X 1999 Evolving Artificial Neural Networks. Proceedings of the IEEE. Vol. 87(9). 1423- 1447.

[18] Cybenko G 1989 Approximation by superpositions of a sigmoidal function. Mathematics of Control, Signals, and Systems. Vol. 2(4), 303–314.

[19] Radcliff J. D 199 Genetic set recombination and its applications to neural network topology optimization. Neural Computing and Applications. Vol 1(1). Vol. 67-90, DOI: 10.1007/BF01411376.

[20] Stanley K.O, Mikkulainen R 2002 Evolving Neural Network Through Augmented Topologies, Evolutionary Computation. Vol. 10(2).

[21] Tao Y et al 2003 Genetic Dissection of Hybrid Incompatibilities between Drosophila and D. Mauritiana II Mapping Hybrid Male Sterility Loci on the third Chromosome, Genetics. Vol. 164, 1399-1418.

[22] Gavrilets S, Li H, Vose M. D 2000 Patterns of Parapatric Speciation. Evolution. Vol. 54 (4), 1126-1134.

[23] Martin A. P 1999 Increasing genomic complexity by gene duplication and the origin of vertebrates. The American Naturalist. Vol. 154(2), 111–128.

[24] Force A. et al 1999 Preservation of duplicate genes by complementary, degenerative mutations. Genetics. Vol. 151, 1531–1545.

[25] Lindgren K, Johansson J 2001 Coevolution of strategies in n-person prisoner’s dilemma. In Crutchfield, J., and Schuster, P., editors, Evolutionary Dynamics - Exploring the Interplay of Selection, Neutrality, Accident, and Function. Reading, MA: Addison-Wesley.

[26] Jain A.K, Nandakumar K, Ross A 2005 Score Normalization in Multimodal Biometric Systems. Pattern Recognition. Vol. 38(12), 2270- 2285.