Eng. & Tech. Journal, Vol.29, No.16, 2011

Message Coding and Compression with Artificial Neural Networks

Hasan M. Azzawi Hassan A. Nasir* & Ali K. Naher*

Received on: 10/1 / 2011
Accepted on: 3 /11/ 2011

Abstract

The need to overcome data preprocessing inherent in much of the classical data coding techniques commonly available led to the search for a free, easy-to-use, but flexible and powerful method. Artificial Neural networks have been attracting more and more researchers since the past decades. The distinct properties, such as learning ability, nonlinearity, fault tolerance, generalization etc., make it suitable for information protection, such as data encryption, data authentication, data detection, etc. In this paper a simple and low-cost coding method based on neural networks is proposed to be used to patterns compression. The goal of the developers is to build a tool able to store and send a coded and compressed message. The formed two-dimensional patterns are coded and compressed using the multilayer neural network with Back-propagation training algorithm. Hidden layer outputs of a trained network are sent as two-dimensional data, which represents the encoded vectors. To reconstruct the original patterns, this requires the output weights matrix and the output nodes functions which are unknown and not available in the encoded sent vectors. A compression rate of about 6:1 has been achieved.

تشفير وضغط الرسائل باستخدام الشبكات العصبية الاصطناعية

الخلاصة

ان التغلب على المعالجة الاولية للبيانات والمتضمنة في أغلب التقنيات التقليدية المتوفرة عادةً لتشفير البيانات يقود الى البحث لتكوين طريقة متحررة سهلة الاستخدام مرنة وقوية. منذ العقود السابقة والشبكات العصبية تجذب أغلب الباحثين، حيث أن خصائصها المتميزة مثل قابلية التعلم، غير الخطية، تحمل الخطأ والتعميم جعلتها مناسبة لحماية المعلومات مثل تشفير البيانات، توثيق البيانات وكشف البيانات. في هذا البحث تم اقتراح طريقة تشفير بسيطة ورخيصة الكلفة تعتمد على الشبكات العصبية وتم أستخدامها لضغط البيانات. الهدف هو بناء أداة قابلة لخزن وارسال رسالة مضغوطة ومشفرة. تم تشفير و ضغط الانماط ثنائية الابعاد باستخدام الشبكة العصبية متعددة الطبقات بمخطط تدريب التتابع العكسي. يتم إرسال مخرجات الطبقة المخفية للشبكة المدربة بصيغة بيانات ثنائية الابعاد والتي تمثل البيانات المشفرة. إن استعادة الأنماط الأصلية يتطلب معرفة مصفوفة الأوزان الخارجية ونوع الدوال في الطبقة الخارجية والتي تكون غير معروفة وغير متوفرة في البيانات المرسلة وهذا يعني أستحالة كشف المعلومات المرسلة. في هذه الطريقة بالإضافة إلى تشفير المعلومات تم تحقيق نسبة ضغط تصل الى 6:1.

1. Introduction

In the past few years a great expansion in fields of image and data processing had been achieved. Several new methods and techniques are being proposed and improved each year in fields of cryptography, Encryption [1], watermarking [2-3] and steganography [4-5]. Current developments in digital image coding tend to involve more and more complex algorithms, and require therefore an increasing amount of computation. To improve the overall system

*Electrical Engineering Department, University of Technology/Baghdad

https://doi.org/10.30684/etj.29.16.14
2412-0758/University of Technology-Iraq, Baghdad, Iraq

performance, some schemes apply different coding algorithms to separate parts of an image according to the content of this subimage; such as the coding algorithm that performs a discrete cosine transform (DCT) on the block, followed by a linear quantization, a zigzag scanning followed by a run-length coding of the parameters and finally a Huffman coding of the resulting run-length codes. Another coding algorithm is the N-level mode that performs a classification of the pixels into N clusters according to their gray level value. The clustering is performed with a fuzzy c-means clustering algorithm. The resulting pixel values are run-length coded, and then Huffman entropy coded [6].

Artificial neural networks (ANNs) can overcome some of the difficulties of the coding algorithms by interpreting images quickly and effectively. By investigating ANN properties, the proposed coding scheme has low complexity, and provides a security system for the sent information.

ANNs are composed of numerous processing elements (PEs) or artificial neuron arranged in various layers, with interconnections between pairs of PEs. They are designed to emulate the structure of natural neural networks such as those of a human brain. For most ANNs, PEs in each layer are fully connected with PEs in the adjacent layer or layers, but are not connected to other PEs in the same layer. The PEs simulates the function of the neurons in natural neural networks, while the interconnections between them mimic the functions of dendrites and axons [7 - 9].

A general mathematical equation for a modeled single artificial neuron is:

$$y(x) = g\left(\sum_{i=0}^{n} w_i x_i\right) \quad \text{.......(1)}$$

where $y(x)$ is one output axon, x is a neuron with n input dentrites $(x_0...x_n)$, w is weight and g is an activation function as shown in figure 1. The input and output layers contain number of neurons equal to number of input and output parameters, respectively [10].

Generally, ANNs can efficiently model various input/output relationships with the advantage of requiring less execution time than a procedural model. These features make the ANN approach very appealing for real-time image processing.

2. Image Quality Measure

The image quality can be evaluated objectively and subjectively. Objective methods are based on computable distortion measures. Standard objective measures of image quality are Mean Square Error (MSE) and Peak Signal-to-Noise Ratio (PSNR) which are defined as [11]:

$$MSE = \frac{1}{MN}\sum_{y=1}^{M}\sum_{x=1}^{N}[I(x,y) - I'(x,y)]^2 \quad \text{... (2)}$$

and

PSNR=20*log10(255/sqrt(MSE)) … (3)

Where I(x,y) is the original image, I'(x,y) is the approximated version (which is actually the decompressed image) and M,N are the dimensions of the images. MSE and PSNR are the

most common methods for measuring the quality of compressed images.
The compression efficiency is defined by the parameter Compression Ratio (CR) and is given by [11]:
CR=Original Data/Compressed Data ………..(4)

3. The Proposed System Implementation

The proposed coding scheme is based on ANN. It has proved to be capable of patterns coding and compression. The design of the proposed coding scheme involves two steps. The first step consists of creating a database of letter patterns and symbols and providing their input vectors. The second step consists of designing the neural net and training it according to the created database. A complete system can be shown in figure 2.

3.1 Creating the Database

In this paper the ANNs are presented with similar binary input-output data set. An input-output value of (-1) is assigned to white and a value of (1) to black. Fourty five patterns are used to train the ANNs.

To create the input vectors, each pattern from the original patterns matrix, with a simple distinct model representation, of (7x5) pixels is lined up in a single vector column with 35 positions. The idea is to use all the vectors as parallel inputs to a single network multilayer perceptron (MLP) composed by 35 neurons in the input and output layers (The input and output layers must contain number of neurons equal to number of input and output parameters, respectively).

3.2 Design and Training of ANN

Different ANN architectures are used, according to different number of hidden nodes, could lead to differences in effectiveness, and, there are networks allow a more flexible interpretation of the results. There are several variations to the basic training algorithm of the back propagation neural network provided by MATLAB that has been used in this research.

Back-propagation networks are selected for this project because they have been successfully used in various image processing applications. Each PE in the input layer received the value of one of the pixels in the input patterns. One hidden layer is used between the input and output layers. Different transfer functions are implemented in each PE such as purelin, log sigmoid and tanh sigmoid.

During the training procedure of the ANNs, the maximum acceptable sum-squared error was empirically set at $1*10^{\wedge-5}$. The training process is carried out with Levenberg-Marquardt (LM) back-propagation, until a maximum of 1000 epochs (cycles) or the maximum acceptable sum-squared error is reached. Some initial runs showed that these settings appeared to be sufficient for this application.

The number of PEs in the hidden layer is arbitrarily selected and can be varied from 6 to 35. A trial-and-error method is used to set them. The model of a small number of hidden nodes (e.g., 6 or 7) is tested and proved its inefficiency due to confusion in relation to the gray levels, providing unsatisfactory results. This is iterated until the difference between the output and the original data patterns is below a given threshold. In our 3-layers network, hidden PEs have been set to 25. This does not cause great losses in the patterns characteristics. At the end of the iterations, there are two weights matrices for coding (compression) and decoding (decompression) all the input patterns.

After that, passing any vector of the original patterns through the hidden layer will obtain the transmitted signals. They are represent the vectors encoded by the hidden weight matrix and the hidden nodes functions as shown in figure 3.a. An inverse process is then applied to reconstruct the patterns that are recovered from the neural networks by the output weights matrix and the output nodes functions (figure 3.b).

4. Results

The ANNs have been trained with the produced dataset; there are 45 patterns of 7x5 pixels, for training, and 10 patterns for testing as shown in figure 4. Many ANN models are developed with different numbers of PEs in their hidden and many back-propagation training algorithms. The performance of the ANNs is compared, The back-propagation training algorithm used TRAINLM algorithm produces reasonably good reconstruction outputs based on the provided inputs.

To test the proposed model, a base with some examples is used. Results of the performed tests show that the proposed model reconstructs the original patterns as shown in figure 5. The results indicate that ANNs can, in general, code and compress the patterns with a compression rate of about 35:6 or approximately 6:1 according to equation (4).

Although the study is limited by the training data, the results indicate the potential of ANNs for fast, powerful and good coding and compression system.

5. Conclusions

The approach introduced in this paper consists of a simple and new technique based on ANNs for patterns coding and compression. Performed tests demonstrate acceptable results, with low computational cost and ease of implementation.

Although there is no method for determining the best number of PEs to include in the hidden layer. However the number of PEs used in this work have been sufficient for such amount of input data. More PEs in the hidden layer would result in better performance.

The increase of hidden nodes significantly increases the computation time in a fully-connected feed-forward network with longer considrable improvements.

The proposed method reduces the preprocessing for the blocking of the input patterns.

While crisp values of (1) and (-1) were used during training, values between (-1) and (1) could result at the output of the ANN, no strategy is required to deal with such values.

To reconstruct the original patterns, this requires the output weights matrix and the output nodes functions which are unknown and not available in the encoded sent vectors.

To improve this situation, additional hidden layers ANN models can be created that produced multi weights matrices for decoding the coded information.

The training data set contained only 45 patterns. It may be necessary to collect more data to increase the size of the training data set and ensure a whole coding system.

Another noteworthy result from this study is that the time needed to train an ANN model is less than 15 minutes; and the testing time is usually less than one second per input data. This information is of paramount importance for real time application problems where one may have only few seconds to make informed decisions about the original sending message.

The MLP networks takes longer training time because they use iterative training algorithm such as Back-propagation, but shorter work time (after training) because of simple dot-product calculations.

References

[1]. R. Martínez-Noriega, H. Kang, B. Kurkoski, K. Yamaguchi, K. Kobayashi, M. Nakano, ***" Increasing Robustness of Audio Watermarking DM using ATHC Codes",*** Conference in technical cooperation with IEEE and IEEE Computer Society, MCIS, 2006.

[2]. Nedeljko Cvejic, ***"Algorithms for audio watermarking and steganography"***, University of Oulu, Finland, 2004.

[3]. Hamza Özer, Bülent Sankur, Nasir Memon, Ismail Avcıbas, ***"Detection of audio covert channels using statistical footprints of hidden messages"***, Digital Signal Processing, Vol. 16, pp. 389–401, Elsevier Inc, December, 2006.

[4]. Guillermo A. Francia, Tyler S. Gomez, ***" Steganography Obliterator: An Attack on the Least Significant Bits"***, InfoSecCD Conference'06, Kennesaw, GA, USA, September 22-23, 2006.

[5]. Ismail Avcıbas¸ Mehdi Kharrazi, Nasir Memon, Bulent Sankur, ***"Image Steganalysis with Binary Similarity Measures"***, EURASIP Journal on Applied Signal Processing, Vol. 17, pp.2749–2757, 2005.

[6]. Pascal Fleury, Olivier Egger, "***Neural Network Based Image Coding Quality Prediction***", Signal Processing Laboratory, Swiss Federal Institute of Technology, Lausanne, Switzerland, 1997.

[7]. Haykin Simon, ***"Neural Network–a comprehensive foundation"***, Prentice Hall, 1999.

[8]. Kinnebrock W., ***"Neural Networks-Fundamental, Applications, and Examples"***, Oldenburg, 1995.

[9]. Patterson Dan W., ***"Artificial Neural Networks, Theory and Application"***, Prentice Hall, 1996.

[10]. Yilmaz Yoru, T. Hikmet Karakoc, Arif Hepbasli, ***"Application of Artificial Neural Network (ANN) Method to Exergetic Analysis of Gas Turbins"***, Int. Symp. on Heat Transfer in Gas Turbine Systems, Antalya, Turkey, 9*14 August 2009.

[11]. G. Sadashivappa and K. V. S. AnandaBabu, ***"Evaluation of Wavelet Filters for Image Compression"***, World Academy of Science, Engineering and Technology, Vol. 51, 2009, pp: 131-137.

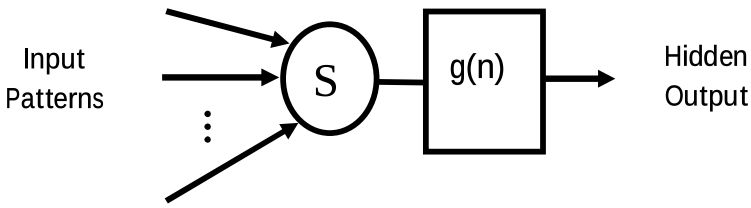


Figure (1) the architecture of artificial neuron

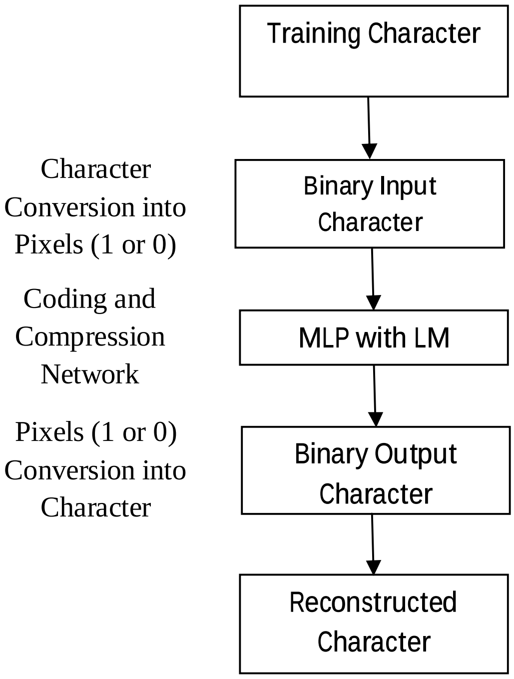


Figure (2) A System for character coding and compression

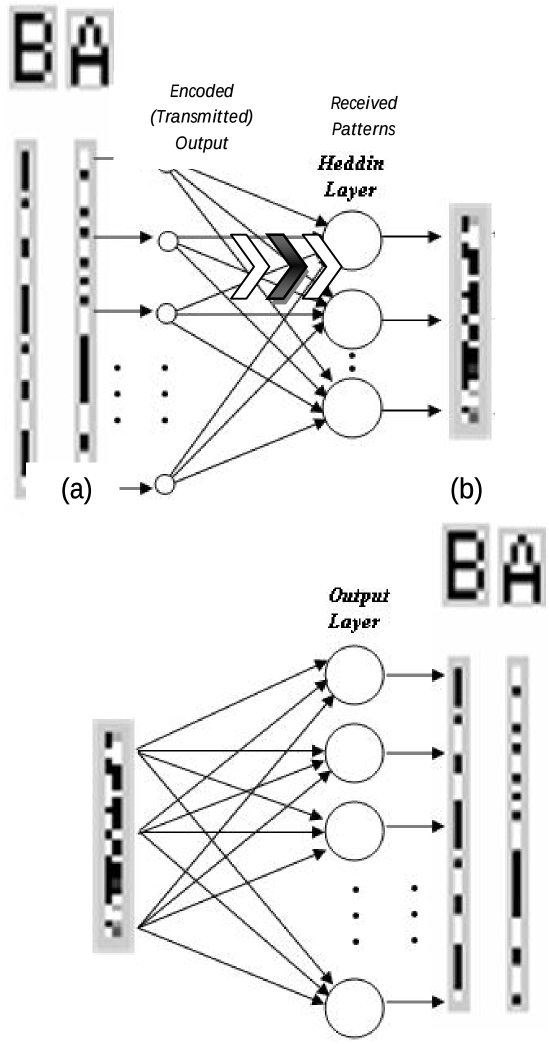
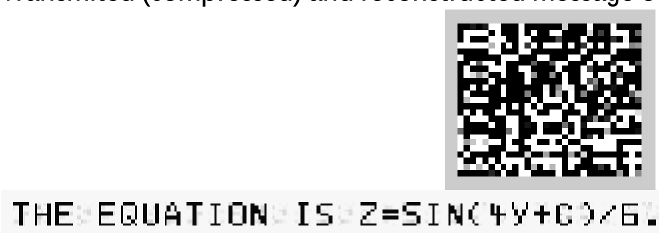


Figure (3) The architecture and process of the proposed ANN for (a) patterns coding and compression (b) patterns decoding and decompression

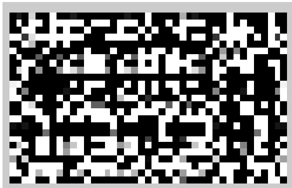
(a) (b)

Figure (4) After training of ANN for patterns coding and compression (a) Hidden layer output (coded patterns) (b) Actual output of ANN.

Transmited (compressed) and reconstructed message of test example 1:

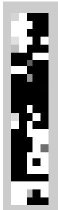


Transmited (compressed) and reconstructed message of test example 2:



SEND TO ME ALL THE AVAILABLE INFORMATIONS

Transmited (compressed) and reconstructed message of test example 3:



DO IT

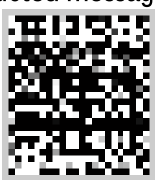
Transmited (compressed) and reconstructed message of test example 4:

THERE ARE SOME ERRORS IN YOUR DATA

Transmited (compressed) and reconstructed message of test example 5:

OK

Transmited (compressed) and reconstructed message of test example 6:



83625 USD FOR EACH ONE

Figure (5) Coded and reconstructed examples for t