Performance analysis of q-valued dimensions Parametric Vector Neural Network classifier

Nwe Nwe Soe, Win Htay
Computer University, Thaton
nwenwesoe.mlm@gmail.com, wynnhtay@gmail.com

Abstract

Neural network ensemble techniques have been shown to be very accurate classification techniques. However, in some real-life applications, a number of classifiers required to achieve a reasonable accuracy is enormously large and hence very space consuming. This paper introduces special neural method, Parametric Vector Neural Network (VNN), which has great associative memory and high performance. Parametric VNN analyzed using various size of database having randomly created patterns, noise levels, and fixed q-dimensions. The result shows that it has capacity much greater than conventional Neural Networks. Once T matrix is created for the stored patterns in Database, most similar pattern with the input one can be achieved easily by just multiplying two matrices. The resulting associative memory can recognize highly noisy and correlate input patterns.

Keywords: *Vector Neural Network (VNN), q- valued dimensions, Neural Network Classifier*

1. Introduction

To use a new and special Neural Network as a classifier in the process of pattern recognition, thorough analysis of the Parametric Vector Neural Network is needed. It is, in fact, a special Neural Network which is described in the following papers [1-4]. Parametric vector models of the associative memory are desirable both with regard to the storage capacity and noise immunity. Furthermore, such models were not widely used up to now. In the publication of the paper [3], the algorithm of mapping of binary patterns into *q*-valued ones was proposed. It was shown in that such mapping allows one to use Vector Neural Network for storing and processing of signals of any type and any dimension. In this paper, performance analysis is carried out to make sure the effectiveness and advantages of the network. In [1-3], it is said that PVNN is suitable to the distorted input pattern to get high accuracy.

The frequency-phase modulation is more convenient for optical processing of signals. It allows us to back down an artificial adaptation of an optical network to amplitude modulated signals [4-9]. Signals with q different frequencies can propagate along one interconnection. In this paper q-size is considered as a constant number 10.

2. Theoretical Background of the VNN

This section provides the main characteristics of the ensembles that will be analyzed in the present work.

2.1. Vector Neural Network (VNN)

Authors have been carrying out research on hand written character recognition from Palm leaf manuscript. After segmentation and object extraction, features must be extracted from segmented object. Unfortunately, for the time of being, real features cannot still be used. To test or analyze PVNN, features are generated randomly and are stored them in Pattern Database.

Let us consider a recognition problem of an input feature vector of a segmented image having N features. In pattern database, there will be M patterns $\{X_\mu\}$.

$$X_\mu = \{\vec{x}_{\mu 1}, \vec{x}_{\mu 2}, \ldots\ldots, \vec{x}_{\mu N}\}, \ \mu = \overline{1, M}$$

$\vec{x}_{\mu i}$, $i = \overline{1, N}$ - unit vector of i[th] neuron directing to one direction of q-dimension space [11-15]. An input vector must be most similar to the one in the pattern database. This process can be seen as in the following Figure 1.

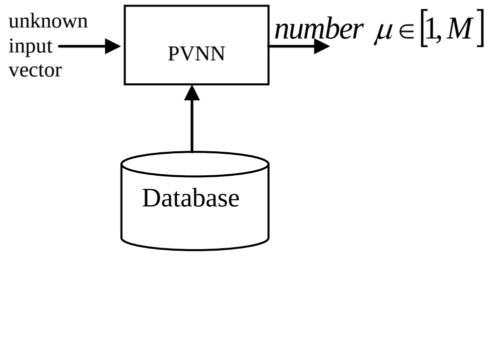

Figure 1. Diagram of pattern recognition using PVNN

This q-dimensional Parametric Vector Neural Network (VNN) can be considered as a two-layer Neural Network. Each neuron $\vec{x}_{\mu i}$ is coded to q-dimensional space before input to the network. All the input neuron is connected each output neuron of PVNN. N-size output vector is considered to get the exact number of pattern which is the most similar to the input. Each $\vec{x}_{\mu}$ must be in $Y_{\mu} = (\vec{y}_{\mu 1}, \vec{y}_{\mu 2}, ..., \vec{y}_{\mu n})$. N is the output neuron count and in this paper the value of n is 4 which means maximum number of pattern is 10000 (from 0 to 9999). Y_{μ} is the output number of the PVNN. μ is decoded from the unit output vector $\vec{y}_{\mu 1}, \vec{y}_{\mu 2}, ..., \vec{y}_{\mu n}$.

Synaptic mutual matrix T is calculated as the following formula-

$$T_{ij} = \sum_{\mu=1}^{M} \vec{y}_{\mu i}\, \vec{x}_{\mu j}^{T},\ i = \overline{1, n},\ j = \overline{1, N}$$

where $\vec{x}_{\mu j}^{T}$ is the transpose matrix of $\vec{x}_{\mu j}$.

Synaptic mutual matrix T_{ij} between i^{th} and j^{th} neurons are defined by qxq matrix.

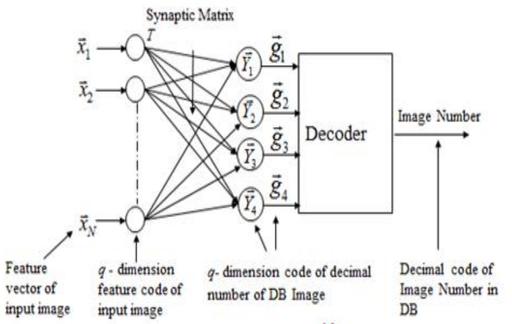

Figure 2. Identification of PVNN

After that, using pattern number in DB and coded features of that image, Synaptic matrix T is created. Once it is created, it will be used to classify the input pattern determining which image number in DB is the most similar to that input one. Output neuron number is depending on the DB size (training pattern count). In fact, output neuron count is actually digit count of the total training images. In the above Figure 2, the Parametric Vector Neural Network is demonstrated how it works on the four output neurons. This type of Neural Network is very fast and suitable for pattern recognition process.

Let us consider that the $Z = (\vec{z}_1, \vec{z}_2, ..., \vec{z}_N)$ is the input feature vector to the PVNN.

$$\vec{h}_i = \sum_{j=1}^{N} \vec{T}_{ij}\, \vec{z}_j,\ i = \overline{1, 4},\ N\text{=feature count}$$

The $(q\times q)$-matrix T_{ij} described in above equation is simply multiply by the input vector. This matrix affects the vector $z_j \in R^q$, converting it in a linear combination of basis vectors. This combination is an analog of the packet of quasi-monochromatic pulses that come from the j^{th} neuron to the i^{th} one after transformation in the inter connection. After decoding the h_i, a decimal number is obtained. This number is the database pattern number which is most similar to the input one [16-19].

3. Generating Pattern Database

The output of this paper is used to determine whether PVNN should be used or not for the features which will be extracted. Up to now real features are still not in hand and pattern database need to be generated with random features.

3.1 Creation of Database for q-valued dimensions with Parametric VNN

To analyze the PVNN, pattern database is created with the randomly generated features. The creating database is performed as shown in Figure 3.

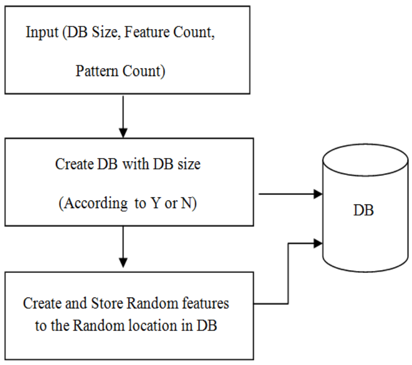

Figure 3. Creation of Database for Random features

3.2 Performance Analysis

The performance analysis of PVNN, an analysis tool system is created using C# programming environment. The user can choose the desired database (DB) size, features and patterns count. After storing all the patterns in DB, Synaptic matrix is calculated according to the DB size, features, patterns count and dimension size.

This matrix is stored for further use in calculation to find similar pattern. The following Figure 4 shows the calculation Synaptic T matrix and store procedure.

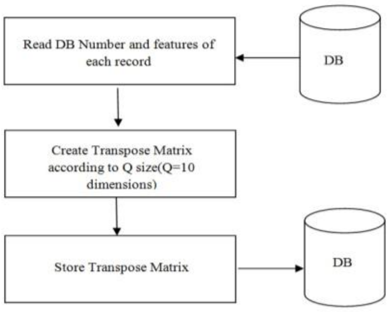

Figure 4. Creation Synaptic Matrix

This paper is intended not to compare with other neural classifiers but to analyze its own parameters such as q-size, database size. Parameters are considered how to used, how much should be used, and to know the tolerance of noise level.

3.3 Testing Recognition Power

To test the recognition power of the T matrix, one of the patterns in DB is used as an input one. Desired noise level is added to that input feature vector. And then read Synaptic matrix from the DB and just multiply by the q-value coded input feature vector to get the output number. This system will easily retrieve the decimal number from DB. We can check whether this number is similar to input one or not is illustrated in the following Figure 5.

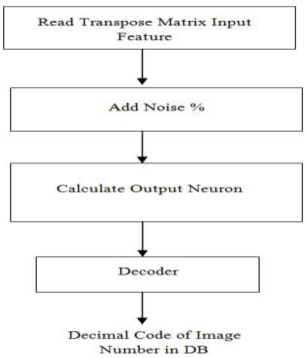

Figure 5. Calculate the decimal code of image number in DB

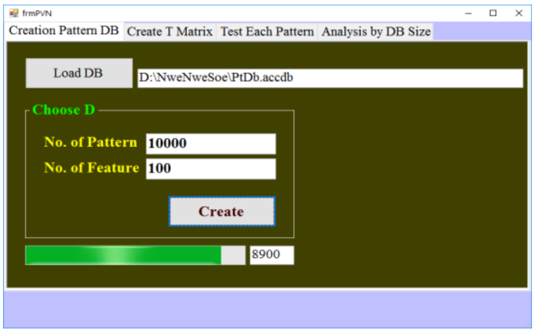

Figure 6. Creating Pattern Database

The above diagram presented in Figure 6 shows the testing process for one selected input. In this program, the four main sub menus included and they are Creation Pattern DB, Create T matrix, Test Each Pattern and Analysis by DB size. It is very clear that the speed of calculation (recognition) process is very high, because it needs only multiplication of two matrices and no need to consider the pattern database. The empirical results show that the PVNN gets the high accuracy.

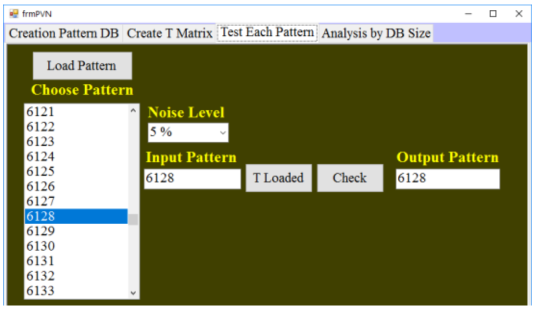

Figure 7. Testing one by one by Noise level %

In the above Figure 7 the user can test and calculate by choosing pattern and noise level percentage and then T loaded and check buttons are clicked. Finally, the results are shown in output pattern.

3.4 Analysis by Pattern Database Size and Noise level

Accuracy mainly depends upon the noise level and database size, Pattern count. In this work, PVNN classifier is analyzed its robustness both with noise level and database size. In the first test, various noise levels (5%, 10%, 15% and 20%) are added to the input feature. The result with noise % level and Accuracy is shown in the following Figure 8.

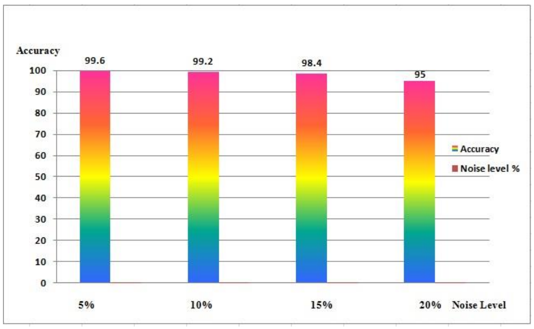

Figure 8. Comparison of Parametric VNN classifier with noise 5%, 10%, 15% and 20% and Accuracies are 99.6, 99.2, and 98.4.

The result shows that the Parametric Vector Neural Network is enough robust to the noisy input feature. For the fix noise level, the performance analysis is carried out on the pattern count in the database. The result is getting worse when database size is greater in Figure 9.

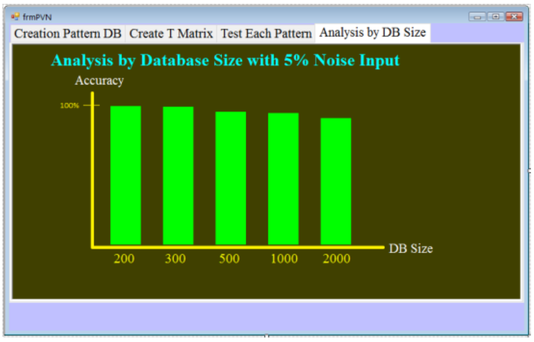

Figure 9. Experimental result by database size with the fix noise level.

Table 1. Performance Analysis by Database Size and Noise level Table calculates Accuracy

Analysis by Database size and noise level				
DB Size	**Accuracy by Noise Level**			
	5%	**10%**	**15%**	**20%**
500	99.6	99.2	98.4	95
1000	99.4	99.1	96.1	90.3
2000	99.2	99.0	95.3	86.7
3000	99.0	98.3	93.4	82.4
10000	98.4	96.2	92.1	80.3

In Table1 compares the performance analysis by Database size and Noise level calculates accuracy by using PVNN classifier. We can observe that our PVNN classifier performs significantly and consistently better than other methods in comparison. The accuracy of the methods, where the modest feature PVNN classifier approach, with corresponding savings in computational time and memory storage by using DB size and noise level %.

4. Benefit and Limitation

High associated memory and very fast processing time is obvious benefits. But when the features are stored sequentially in DB, it frequently occur false recognition. That is why, database record size is needed to predefine and each record is stored at the random places in Database. To get high recognition power, q-size is needed to increase.

5. Conclusion

In this paper a different type of neural classifier is introduced to enhance the performance analysis of q-dimensional PVNN classifier. It just needs to create a Synaptic matrix once and after that just multiplication is needed to get result. Experimental results show that the proposed neural classifier can effectively achieve high accuracy and very fast. It can be used as the classifier for the noisy input because of good associated memory. According to the empirical and analytical result, we can use it for handwritten palm leaf manuscripts recognition.

References

[1]. B. V. Kryzhanovsky and A. L. Mikaelyan. “An associative memory capable of recognizing strongly correlated patterns”, Doklady Mathematics, Vol.67, No.3, Russian, 2003, pp.455-459.

[2]. B.V. Kryzhanovsky and L.B.Litinskii and A.Fonarev, Optical Neural Network Based on the Parametrical Four Wave Mixing Process”, Proc. of ICONIP, Vol.4, Singapore, 2002, pp.1704-1707.

[3]. B.V. Kryzhanovsky, L.B. Litinskii, “Vector models of associative memory Automaton and remote control”, Vol. 64, Russian, 2003, pp.1782-1793.

[4]. B.V. Kryzhanovsky, L.B. Litinskii and A.Fonarev, “Parametrical neural network based on the four-wave mixing process”, Nuclear Instruments and Methods in Physics Research, Vol. 502, No.2-3, Russian, 2003, pp. 517 – 519.

[5]. Boris V Kryzhanovsky, Andery L. Mikaelyan. “Vector neural net identifying many strongly distorted and correlated patterns”, International conference on Information Optics and Photonics Technology Photonics Asia-2004, Vol. 5642, Beijing-2004., pp. 124-133.

[6]. Noest, J, “Discrete-state phasor neural networks”, Phys. Rev. A 38, 1988, pp.2196–2199.

[7]. Kanter, I, “Potts-glass models of neural networks”, Phys. Rev. A 37, 1988, pp.2739–2742.

[8]. Cook, J, “The mean-field theory of a Q-state neural network model”, J. Phys. A 22, 1989, pp. 2000–2012.

[9]. Vogt, H., Zippelius, A, “Invariant Recognition in Potts Glass Neural Networks”, J. Phys.A 25, 1992, pp.2209–2226.

[10]. Kryzhanovsky, B.V, Mikaelian, A, “On the Recognition Ability of a Neural Network on Neurons with Parametric Transformation of Frequencies”, Doklady Mathematics 65(2), 2002, pp.286–288.

[11]. B.V.Kryzhanovsky, V.M.Kryzhanovsky, A.L. Mikaelian and A.Fonarev. “Parametric dynamic neural network recognition power”. Optical Memory & Neural Networks, Vol. 10, No.4, 2001, pp.211-218.

[12]. Sung-Bae Cho, “Neural-Network Classifiers for Recognizing Totally Unconstrained Handwritten Numerals”, IEEE Transactions On Neural Networks, Vol. 8, No. 1, January, 1997, pp. 43-53.

[13]. Hansen, L., and Salamon, P., “Neural Network Ensembles”, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1990, pp. 993-1001.

[14]. Krogh, A., Vedelsby, J., “Neural Network Ensembles, Cross Validation and Active Learning”, in Tesauro, G., Touretzky, D., and Leen, T., (Eds.), Advances in Neural Information Processing Systems, Vol. 7, MIT Press, 1995.

[15]. Hashem, S., Schmeiser, B. and Yih, Y., “Optimal Linear Combination of Neural Networks: An Overview”, Proceedings of the IEEE International Conference on Neural Networks, 1994.

[16]. Maclin, R. and Shavlik, J., “Combining the Predictions of Multiple Classifiers: Using Competitive Learning to Initialize Neural Networks”, Proceedings of the 14th International Joint Conference on Artificial Intelligence, 1995.

[17]. Funahashi, K. “On the approximate realization of continuous mapping by neural networks”, Neural Networks 2, 1989, pp. 183–192.

[18]. Hartono, P., Hashimoto, S. “Ensemble of linear perceptrons with confidence level output”, In: Proc. 4th Intl. Conf. on Hybrid Intelligent Systems. Kitakyushu, Japan, 2004, pp. 186–191.

[19]. Hornik, K., Stinchcombe, M., White, H., “Multilayer feedforward networks are universal approximators”, Neural Networks 2, 1989, pp.359–366.