AN EFFECTIVE AND NOVEL WAVELET NEURAL NETWORK APPROACH IN CLASSIFYING TYPE 2 DIABETICS

Zarita Zainuddin, Pauline Ong**

Abstract: Designing a wavelet neural network (WNN) needs to be done judiciously in attaining the optimal generalization performance. Its prediction competence relies highly on the initial value of translation vectors. However, there is no established solution in determining the appropriate initial value for the translation vectors at this moment. In this paper, we propose a novel enhanced fuzzy c-means clustering algorithm – specifically, the modified point symmetry-based fuzzy c-means (MPSDFCM) algorithm – in initializing the translation vectors of the WNNs. The effectiveness of embedding different activation functions in WNNs will be investigated as well. The categorization effectiveness of the proposed WNNs model was then evaluated in classifying the type 2 diabetics, and was compared with the multilayer perceptrons (MLPs) and radial basis function neural networks (RBFNNs) models. Performance assessment shows that our proposed model outperforms the rest, since a 100% superior classification rate was achieved.

Key words: *Clustering, diabetes, fuzzy c-means, microarray, wavelet neural network*

Received: October 28, 2010
Revised and accepted: September 14, 2012

1. Introduction

Wavelet neural networks (WNNs) have been introduced as an alternative to MLPs; they overcome MLPs' shortcomings of multilayer structure, use of a global activation function and a slow learning algorithm [1]. WNNs as universal approximators have more compact topology than other neural networks and fast learning speed owing to the constitution of the localized wavelet activation function in the hidden layer [2]. Due to its fascinating characteristics, WNNs are gaining widespread interests in wide array of fields, from engineering to medicine, from forecasting to classification. In general, the applications of WNNs can be discriminated into two

*Zarita Zainuddin – corresponding author, Pauline Ong
School of Mathematical Sciences, Universiti Sains Malaysia, 11800 USM, Penang, Malaysia Tel.: 6046533940 Fax: 6046570910 E-mail: zarita@cs.usm.my

major areas, namely, the function approximation and classification problems. In the context of function approximation in time-series data, forecasting the global solar irradiance [3], as well as predicting the air pollutant concentration to a higher degree of accuracy [4], are some examples of the real-world applications which have made use of the effectiveness and competence of the WNNs. Apart from working effectively in analyzing the time-series data, WNNs have also shown their efficiency and precision in the complex real-world application, such as forecasting the pulp and paper properties [5], predicting the thin film thickness [6], quantifying the concentration for the phenolic compounds and also the heavy metals present in a solution for an electronic tongue employing voltammetric sensors [7, 8], and forecasting the aeroengine health condition to deduce the aeroengine performance deterioration for ensuring the flight safety [9].

The expediency of the WNNs has moreover opened up a new area in solving the classification problems, where it has been utilized successfully in the categorization of heterogeneous gene expression data, which is constructive for the early diagnosis and cancer treatment [10, 11], classification of ECG beats in the study of arrhythmias [12], power signal classification [13], as well as the digital modulation recognition problems [14]. Definitely the applications of WNNs are not limited to the aforementioned examples. In fact, its popularity increases from time to time, where the list of its applications in the real-world problem is still ongoing.

Various issues have been addressed in WNNs studies, which include adjusting the connection weights by employing different learning algorithms [15, 16], making alterations in the network architecture [17], introducing variation in the types of activation functions used in the hidden layer [18] and modifying the wavelet function parameters [19], a process that involves proper initialization of the translation and dilation vectors [20-22]. In this study, we aim to optimize the convergence characteristic and generalization ability of WNNs by emphasizing the choice of an appropriate wavelet family as the activation functions in the WNNs hidden layer and by determining the locations of the translation vectors.

WNNs update their connection weights and parameters iteratively through learning. During the learning process of a WNN, the selection of the numbers and the locations of the translation vectors are particularly crucial. An appropriate initialization of the translation parameter will do a good job of reflecting the essential attributes of the input samples, which is important for finding an optimal solution. Increasing the number of hidden nodes leads to over-fitting and computational complexity [23, 24]. Thus, assigning an appropriate number of hidden nodes simplifies the process [25].

Several approaches have been used to choose the WNNs translation vector, including an explicit expression [3, 6, 7, 20, 21, 26–28], support vector machines [22], hierarchical clustering [29], k-means clustering [30] and genetic algorithms (GAs) [31]. Although a number of promising methods that are adequate have been developed, translation vector initialization still needs more work. For example, using an explicit expression to define the translation parameter is merely a rule of thumb that is not applicable in every situation. Issues remain with algorithmic complexity, time-consuming programming execution when implementing GAs and support vector machines and the intricacy inherent in defining initial values from the branches of a binary tree in hierarchical clustering.

In this paper, we propose a novel clustering algorithm – the modified point symmetry-based fuzzy c-means (MPSDFCM) – as an alternative to the existing WNNs translation parameter initialization approaches. The proposed system was applied in classifying the type 2 diabetics and compared with the initialization methods based on k-means clustering (KM), fuzzy c-means clustering (FCM), symmetry-based k-means clustering (PSDKM), symmetry-based fuzzy c-means clustering (PSDFCM) and modified point symmetry-based k-means (MPSDKM). An experimental analysis showed that the proposed system enhanced the prediction precision and generalization ability more than the rest. We will also study the relationship between types of activation functions used in the hidden layer of WNNs and its prediction error, where the approximation capabilities of WNNs with Mexican Hat, Gaussian wavelet and Morlet activation functions respectively will be explored.

The paper is organized as follows. In Section 2, a brief introduction of WNNs is given. The description for various clustering algorithms is presented in Section 3. A brief introduction for the research background on diabetes mellitus is given in Section 4. The analyses of the experimental results and performance comparison with MLPs and RBFNNs are given in Sections 5 and 6. Finally, some conclusions are drawn in Section 7.

2. Wavelet Neural Network

2.1 Wavelet neural network architecture

Generalized from radial basis function neural networks, WNNs are in fact feed-forward neural networks with one hidden layer, radial wavelets (e.g., Mexican Hat, Morlet and Gaussian wavelet) as activation functions in the hidden nodes and a linear output layer. The structure of a WNN is illustrated in Fig. 1.

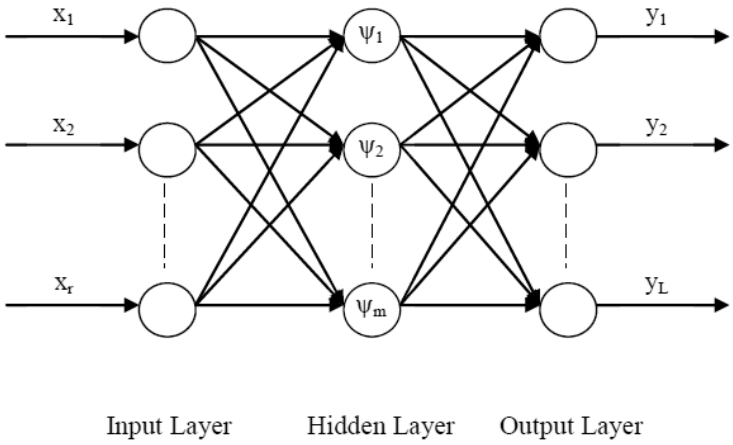

Fig. 1 *A schematic diagram for a WNN with r input nodes, m hidden nodes and L output nodes.*

The input layer receives the input variable $x = [x_1, x_2, \ldots, x_r]$, where r is the number of dimensions, and it transmits the accepted input to the hidden layer. The output for each hidden node in this layer is given as the product of the j-th multi-dimensional wavelet with r input dimensions as:

$$\psi_j(x) = \prod_{i=1}^{r} \Phi_{s_{ij},t_{ij}}(x_i), \tag{1}$$

where

$$\Phi_{s_{ij},t_{ij}}(x_i) = \Phi\left(\frac{x_i - t_{ij}}{s_{ij}}\right). \tag{2}$$

$\Phi_{s_{ij},t_{ij}}$ is the wavelet activation function of the j-th hidden node connected to the i-th input node and s_{ij} and t_{ij} are the scaling and translation vectors, respectively. The products of the hidden layer are then propagated to the output layer, where the output of the WNNs will be the linear combination of the weighted sum of the hidden layer, which is represented in the form of:

$$y_l(x) = \sum_{j=1}^{m} w_{jl}\psi_j(x) + \theta_l, l = 1, 2, \ldots, L. \tag{3}$$

The sigmoid function (logistic and hyperbolic tangent) is the commonly used activation function in an MLP. Composed of the localized wavelet activation function in the hidden layer of the WNNs, the connection weight associated with the hidden nodes can be viewed as local piecewise constant models, which leads to learning efficiency and structure transparency. Three different wavelet families were used as the activation function in this study:

1. Gaussian wavelet:
$$\Phi(x) = -x.\exp(-x^2/2) \tag{4}$$

2. Mexican Hat:
$$\Phi(x) = (n - 2x_2).\exp(-x^2), \text{ where } n = \dim(x) \tag{5}$$

3. Morlet:
$$\Phi(x) = \cos(5x).\exp(-x^2/2). \tag{6}$$

As shown in Equation (2), the influence of each wavelet activation function is determined by the scaling and translation vectors. A wavelet function may be too localized in nature if its scaling vector is set to a very small value; however, wavelet functions with large scaling values will respond in the same manner even though there exists a large variation between the input pattern and the translation vectors. On the other hand, the wavelet activation may fall out of the domain of interest if the translation vectors are not chosen appropriately. It is due to the fact that the influence of the activation function decreases proportionally to its distance from the translation vectors. The input patterns which are located far from the translation vectors fail to activate the wavelet function, hence, in turn their activations will be close to zero [21].

The nomenclature of the WNNs is summarized as follows:

x :	$x = [x_1, x_2, \ldots, x_r]$ is the input variable
y_l :	the l-th output of the WNNs
r :	number of input nodes
j :	$1, \ldots, m$, node index for the hidden layer
l :	$1, \ldots, L$, node index for the output layer
m :	number of hidden nodes
L :	number of output nodes
w_{jl} :	the connecting weights between the hidden layer and the output layer
ψ_j :	product of the j-th multi-dimensional wavelet with d input dimension
$\Phi_{s_{ij},t_{ij}}$:	the wavelet activation function of the j-th hidden node connected to the i-th input node
s_{ij}:	the scaling vector of the j-th wavelet activation function for the i-th input node
t_{ij}:	the translation vector of the j-th wavelet activation function for the i-th input node
θ_l :	$\theta_l \in R$ is the bias parameter of the l-th output node.

2.2 Learning algorithm of wavelet neural network

In the typical learning process of a neural network, learning can be divided into two stages. The first stage involves defining the activation functions used and initializing the parameters, and the second stage involves adjusting the weight vector between the hidden layer and output layer.

In this work, the wavelet families mentioned in the previous section were integrated as the activation function into WNNs-based classifiers. Initialization of the scaling vector was done accordingly to the Zhang and Benveniste (1992) method [1], while the initialization of the translation vector was done using different clustering algorithms that will be discussed in the next section.

To adjust the weight vector W in order to map the underlying relationship between the input and output space, the method of solving the pseudo-inverse was employed.

Before we begin to describe the learning algorithm of the WNNs, let us define the error signal $e_l(n)$ at the output node l at the iteration n as:

$$e_l(n) = f_l(n) - y_l(n), \tag{7}$$

where $f_l(n)$ and $y_l(n)$ are the desired output value and the actual response of output node l at iteration n, respectively.

Hence, the instantaneous value $\xi(n)$ of the sum of squared error over all neurons can be represented as:

$$\xi(n) = \frac{1}{2} \sum_{j \in C} e_j^2(n), \tag{8}$$

where C indicates all the output nodes of the network. Hence, the training of the WNNs is based on minimizing the cost function in Equation (8).

Let us represent Equation (3) as $Y = \Psi W$, where

$$Y = (y_1, y_2, \ldots, y_L)^T \tag{9}$$

$$\Psi = \begin{pmatrix} \psi(x_1, s_1, t_1) & \psi(x_1, s_2, t_2) & \ldots & \psi(x_1, s_m, t_m) \\ \psi(x_2, s_1, t_1) & \psi(x_2, s_2, t_2) & \ldots & \psi(x_2, s_m, t_m) \\ : & : & \ldots & : \\ \psi(x_d, s_1, t_1) & \psi(x_d, s_2, t_2) & \ldots & \psi(x_d, s_m, t_m) \end{pmatrix}. \tag{10}$$

and

$$W = (w_1, w_2, \ldots, w_L)^T, \tag{11}$$

where $w_j = (w_{j1}, w_{j2}, \ldots, w_{jm})^T$, $j = 1, \ldots, L$, represents the connecting weight vectors between i-th output node and the hidden nodes.

To solve the weight vector W,

$$W = \Psi^+ Y \tag{12}$$

is computed, where Ψ^+ is the pseudo-inverse defined as $\Psi^+ = (\Psi^T \Psi)^{-1} \Psi^T$.

The flow of the learning algorithm is summarized in Fig. 2.

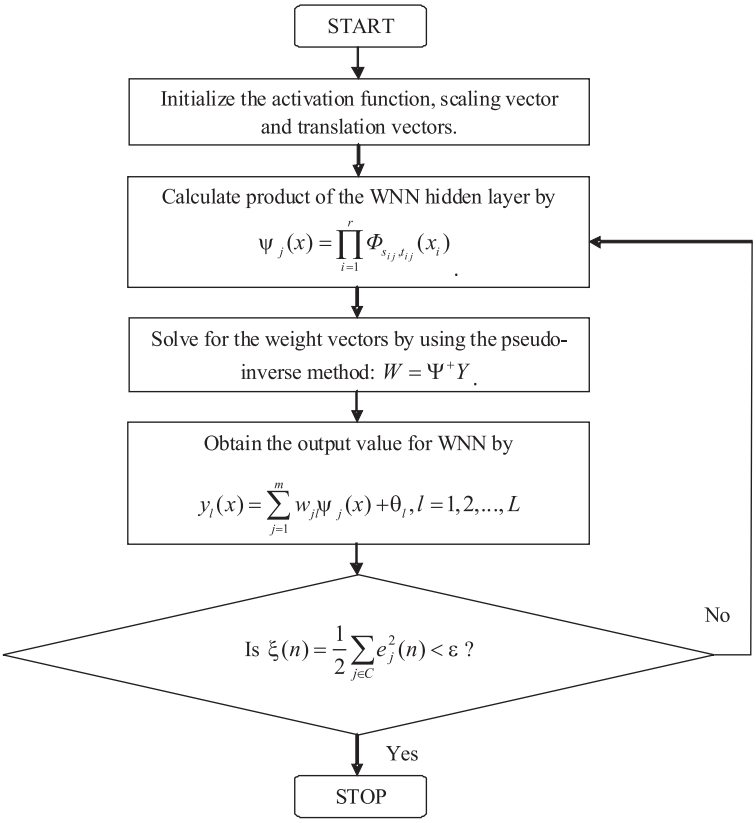

Fig. 2 *The flowchart of the learning algorithm in WNNs.*

3. Clustering Algorithms in Translation Vector Initialization

In this section, a brief introduction of the KM, FCM, PSDKM, PSDFCM and MPSDKM clustering algorithms will be given, as well as the discussion on the proposed MPSDFCM clustering method.

3.1 The conventional k-means and fuzzy c-means algorithm

The action of grouping patterns together into dissimilar clusters with respect to a similarity measure is referred as clustering. It is done in such a way that the patterns within the same cluster have a higher similarity measure than the patterns in the other clusters. The two main approaches to cluster analysis are crisp/hard clustering and fuzzy clustering, where the KM and FCM algorithms are most widely used algorithms for each approach, respectively. For the former approach, each pattern can only be categorized into one cluster, whereas for the latter, each pattern can belong to more than one cluster, with a degree of similarity that is specified by a membership function.

However, both the KM and FCM algorithms use Euclidean distance as their similarity measure, which tends to detect a hyperspherical-shaped cluster of equal size. This behavior is unfavorable when the real-world data consist of various sizes and shapes. Thus, these conventional clustering algorithms need to be enhanced.

3.2 The symmetry-based k-means and fuzzy c-means algorithm

If we notice our surroundings carefully, it can be observed that the concept of symmetry exists in almost every single area of our daily lives. A circle, car, ladder or even the crystal structures of snow exhibit symmetry. Thus, a novel similarity measure – the point symmetry distance (PSD) – proposed by Su and Chou (2001) [32] is adopted into the conventional clustering algorithms in such a way that patterns are assigned to a cluster if they have a symmetrical sense with respect to the cluster center.

Given a set of patterns p_i, where $i = 1, \ldots, N$ and a cluster center c, to measure the degree of symmetry of a data point with respect to a cluster center, the PSD is defined as:

$$D_s(p_j, c) = \min_{i=1,\ldots,N, i \neq j} \frac{||(p_j - c) + (p_i - c)||}{||(p_j - c)|| + ||(p_i - c)||}. \tag{13}$$

Su, Chou and Hsieh (2005) [33] combine the PSD with KM and FCM clustering algorithms. The resulting algorithms have been proven to work efficiently in handling data with different geometric properties. For simplicity, the resulting clustering algorithms are called the symmetry-based k-means (PSDKM) and the symmetry-based fuzzy c-means (PSDFCM), respectively. Details about the algorithms can be obtained from the respective papers.

3.3 The modified point symmetry based k-means algorithm

Upon analyzing the PSDKM and the PSDFCM algorithms, Chung and Lin (2007) [34] pointed out that there are potential problems that exist in the PSD, including the disadvantages of lack of a distance difference symmetry property, unsatisfactory handling of symmetrical inter-clusters, and lack of a closure property. In an attempt to solve these shortcomings, they have proposed a new operator – the symmetry similarity level (SSL) – which is now defined as:

$$SSL(p_i, c_k) = \max_{p_j \in c_k} \sqrt{\frac{DSL^2(p_i, c_k, p_j) + OSL^2(p_i, c_k, p_j)}{2}} \tag{14}$$

for $1 \leq k \leq K$ and $1 \leq i \leq N$, where K is the number of clusters and N is the number of data points.

As observed from Equation (14), the SSL operator consists of two components – the distance similarity level (DSL) and the orientation similarity level (OSL). The DSL operator is defined as:

$$DSL(p_i, c_k, p_j) = \begin{cases} 1 - \frac{|d_i - d_j|}{d_i}, \text{ if } & 0 \leq d_j/d_i \leq 2 \\ 0 & \text{otherwise} \end{cases}, \tag{15}$$

where $d_i = \overline{p_i c_k}$ and $d_j = \overline{p_j c_k} \cdot d_i = \overline{p_i c_k}$ and $d_j = \overline{p_j c_k}$ represent the Euclidean norm difference between pattern p_i to cluster center c_k, and Euclidean norm difference between pattern p_j to cluster center c_k, respectively.

The OSL operator is defined as:

$$OSL(p_i, c_k, p_j) = \frac{v_i.v_j}{2||v_i||||v_j||} + 0.5, \tag{16}$$

where $v_i = (c_k - p_i)$ and $v_j = (p_j - c_k)$.

From Equation (14), Chuang and Lin have integrated the SSL operator into the KM algorithm, and the resulting MPSDKM performs efficiently in detecting not only the symmetrical intra-clusters but also the symmetrical inter-clusters.

3.4 The proposed modified point symmetry-based fuzzy c-means algorithm

Inspired by the effectiveness of the SSL operator in detecting both the symmetrical intra-clusters and inter-clusters, we proposed a novel clustering algorithm – the MPSDFCM algorithm.

Given a set of data points p_i, where $i = 1, 2, \ldots, N$, by borrowing the strength of the PSDFCM and MPSDKM clustering algorithm, the proposed MPSDFCM works as follows:

Step 1: **Initialization**
Initialize the cluster centers c_i, where $i = 1, 2, ...K$ by selecting K points randomly from all the data points.

Step 2: **Coarse-Tuning**
Use the conventional FCM algorithm to update the K cluster centers.

Step 3: **Fine-Tuning 1**
Find out the set Sb_{ik} of all candidate symmetrical data points p_j for each data point p_i such that $DSL(p_i, c_k, p_j) \geq \alpha(= 0.6)$ and $OSL(p_i, c_k, p_j) \geq \beta(= 0.97)$, where $p_j \in c_k$.

Step 4: **Fine-Tuning 2**
Compute Equation (14) for each data point p_i.
If the value for $SSL(p_i, c_k*)$ is the largest and the most symmetrical point p_j, relative to c_k* belongs to Sb_{ik}, assign data point p_i to the cluster center c_k*.
Otherwise, assign data point p_i to the cluster center c_k** with the shortest distance.
Next, update the membership matrix u_{ij} based on the criterion that if $SSL(p_i, c_k*) > \theta \in [0, 1]$,

$$u_{ij} = \begin{cases} u_{ij} = 1, & \text{if} \quad j = k \\ u_{ij} = 0, & \text{if} \quad j \neq k \end{cases}$$

Otherwise, update the membership matrix u_{ij} based on the criterion that

$$u_{ij} = \frac{1}{\sum\limits_{k=1}^{c} \left(\frac{d_{ji}}{d_{ki}}\right)^{2/(m-1)}}.$$

Step 5: **Updating**
Update the new cluster center for K clusters by

$$c_k^{new} = \frac{\sum\limits_{i=1}^{n} u_{ji}^m p_i}{\sum\limits_{i=1}^{n} u_{ji}^m},$$

where $p_i \in c_k$.

Step 6: **Continuation**
If there are no pattern changes or the iterations reach a predefined maximum value, then stop. Otherwise, go to Step 3.

During the fine-tuning process in Step 3, the threshold values of 0.6 and 0.97 were imposed on the $DSL(p_i, c_k, p_j)$ and $OSL(p_i, c_k, p_j)$, respectively. This constraint was made on the assumption that the tolerance rate of the distance difference is less than 40% between the two line segments d_i and d_j, and also the tolerance rate of the angle of orientation is less than 20^o. These assumptions are considered reasonable, since if the distance difference and the angle of orientation are more than the threshold values, the data point p_j is no longer considered as symmetrical with the data point p_i.

From the assumptions, as observed in Equation (15), thus we have

$$\begin{aligned} \alpha &= 1 - \frac{|d_i - d_j|}{d_i} \\ &= 1 - 0.4 \\ &= 0.6 \end{aligned}$$

and as shown in Equation (16),

$$\begin{aligned} \beta &= \frac{v_i \cdot v_j}{2||v_i|| \cdot ||v_j||} + 0.5 \\ &= \frac{||v_i|| \cdot ||v_j|| \cos\theta}{2||v_i|| \cdot ||v_j||} + 0.5 \\ &= \frac{\cos\theta}{2} + 0.5 \\ &= \frac{\cos 20^\circ}{2} + 0.5 \\ &= 0.97 \end{aligned}$$

We proved the effectiveness of the proposed novel MPSDFCM in simulations on an artificial dataset shown below. In Fig. 3(a), there are three compact circles in the artificial dataset. The clustering results for each algorithm (FCM, PSDFCM and MPSDFCM) are shown in Fig. 3(b)–(d), respectively.

As shown in Fig. 3(c), the PSDFCM algorithm failed to correctly cluster the data points into three distinct compact circles. This shows the inability of the PSDFCM algorithm to cluster symmetrical inter-clusters. The clustering result from the proposed MPSDFCM algorithm was satisfactory (Fig. 3(d)).

Next, an artificial dataset containing two compact circles and two crossed ellipsoidal shells was used and is illustrated in Fig. 4(a). All the objects were symmetrical intra-clusters, and the two compact circles were also symmetrical inter-clusters to each other. The clustering results from applying the FCM, PSDFCM and the proposed MPSDFCM algorithms are shown in Fig. 4(b)–(d), respectively.

From Fig. 4, it can be clearly seen that only the proposed MPSDFCM was able to detect both of the symmetrical inter-clusters and intra-clusters accurately. The PSDFCM classifies the data points into two clusters only, which is not accurate. Thus, we believe that the competence and robustness of the novel MPSDFCM algorithm make it an efficient tool in the parameter initialization of WNNs, which will be proven in the experimental simulations described in Section 5.

4. Research Background

Type 2 diabetes mellitus (DM2) is the most common form of diabetes. Diabetics produce adequate amount of insulin, but the body does not respond to the insulin normally. Patients with DM2 do not depend greatly on injection of insulin dose.

Impaired glucose tolerance (IGT) is a pre-diabetic state, where the blood glucose level is higher than normal, but it is not high enough to be classified as

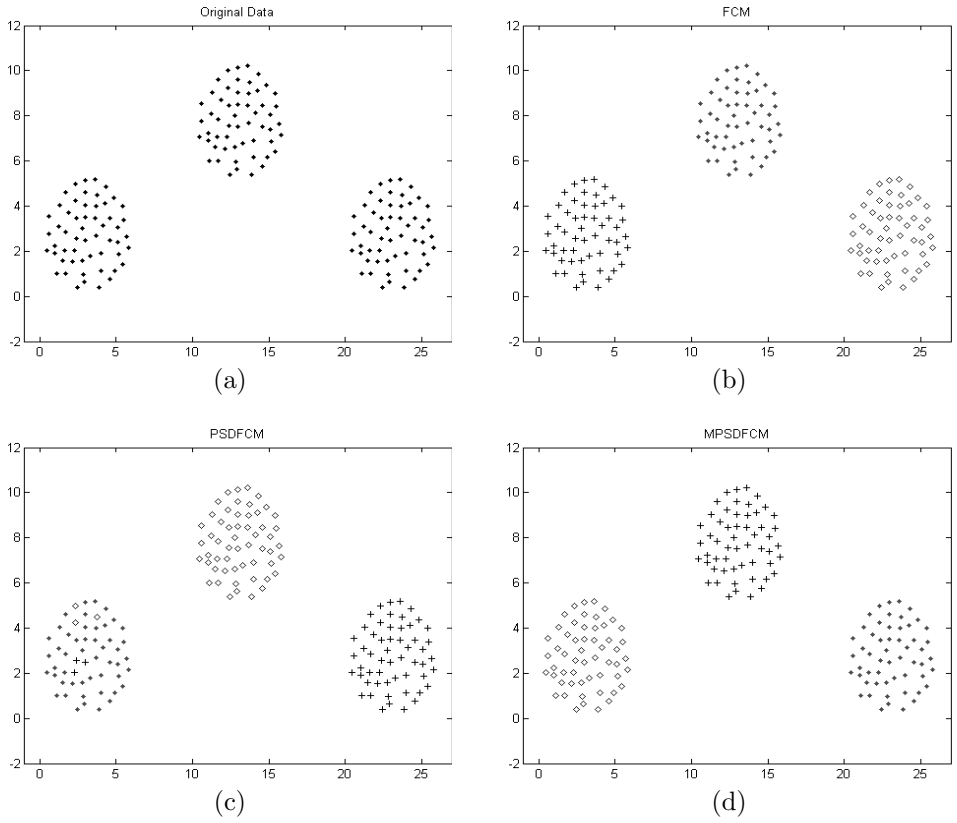

Fig. 3 *(a) Artificial dataset; (b) Clustering result by FCM; (c) Clustering result by PSDFCM; (d) Clustering results by MPSDFCM.*

diabetes. Patients with IGT are in high risk of developing diabetes, thus they are an important target group for early prevention. However, the onset of IGT is gradual and more often IGT shows mild noticeable symptoms at early stage. Since the condition of IGT is rarely diagnosed, patients with IGT may precede to DM2 several years later, when severe complications are already present. Prevalence of IGT can be detected by oral glucose tolerance test (OGTT). However, the judgment from OGTT is unreliable, since the patients might show normal or mildly elevated fasting glucose level, and the factors like having any other illness, such as a cold, lying down when the test is taken, taking medicine, smoking and drinking coffee before the OGTT will affect the accuracy of the test. Hence, a reliable alternative method, other than the gold standard OGTT, would be useful.

Development of microarray technology enables monitoring of thousands of genes expression profiles simultaneously in a single experiment. From the microarray experiment, the subtle differences between the genes expression profiles of different subtypes of heterogeneous tumours as well as different development stages of a disease can be identified. Hence, by applying microarray approach, discriminating

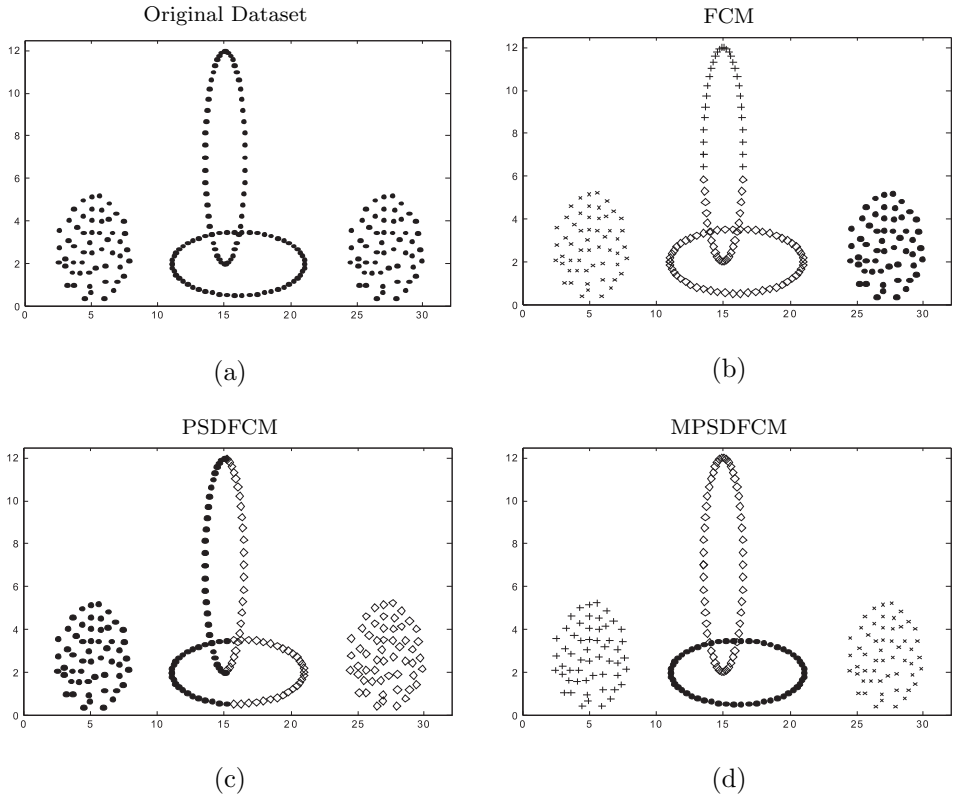

Fig. 4 *4(a) Artificial dataset; (b) Clustering result by FCM; (c) Clustering result by PSDFCM; (d) Clustering results by MPSDFCM.*

between the patients with NGT, IGT and DM2 is more reliable and accurate, since DNA microarray can discover the subtle genes expression changes characteristic of disease.

Many attempts have been done in distinguishing the DM2 patients from the healthy persons and the accuracy of the prediction continues. Statistical methods such as Bayes discriminant function, linear discriminant analysis and Fisher discriminant analysis [35], machine learning methods such as shunting inhibition artificial neural network [36], resource limited Artificial Limited System [37], ensemble neural network coupled with C4.5 rule [38], support vector machine [39] and also adaptive neuro-fuzzy inference system [40] have been shown working effectively in classifying the diabetic patients. However, up to now, development of the aforementioned computational methods and the mathematical approaches has merely been an attempt to classify between the diabetics and the non-diabetics groups. No research has been conducted to discriminate between the subgroups of NGT, IGT and DM2 from the diabetes dataset. The first attempt of implementing the proposed WNNs on the diabetes microarray dataset is encouraging, since from our previous work, the proposed system yields superior accuracy when it was

applied in classifying the heterogeneous cancer microarray dataset [10, 41]. Furthermore, studies on the application of WNNs in the context of classification have been reviewed rigorously [12–14], where the results showed that WNNs appear as a promising classification method for multi-sample data.

5. Experimental Simulations

Gene expression of 22283 genes in skeletal muscle biopsy samples from 43 male subjects (17 persons with NGT, 8 patients with IGT and 18 patients with DM2) were obtained from Array Express Database (http://www.ebi.ac.uk/microarray-as/ae/files/E-CBIL-30/E-CBIL-30.processed.zip), which is the collection of database of gene expression and other microarray data at the European Bioinformatics Institute, Cambridge. The dataset underwent the preprocessing of logarithmic transformation and quantile normalization [42], in order to reduce the distortion of the data, to remove the effects of systematic sources of variations and to transform the data into a scale suitable for analysis.

5.1 Microarray data preprocessing

Let x_{gi} denotes the spot intensity measurement of gene g-th ($g = 1, \ldots, G$) with respect to the sample i-th ($i = 1, \ldots, I$). By applying the logarithmic transformation, the transformed spot intensity measurement X_{gi} is calculated by:

$$X_{gi} = \log(x_{gi}). \tag{17}$$

Subsequently, quantile normalization is applied on the transformed spot intensity measurement X_{gi}. Let the median mock array of X_{gi} be:

$$M_g = median\{X_{g1}, X_{g2}, \ldots, X_{gI}\}. \tag{18}$$

Followed from that, the percentiles $(Q_{i0}, Q_{i1}, \ldots, Q_{i100})$ of the i-th sample and the percentiles $(Q_{M0}, Q_{M1}, \ldots, Q_{M100})$ of the median mock array are calculated. Based on the percentiles $(Q_{i0}, Q_{i1}, \ldots, Q_{i100})$, for any value of X_{gi}, the interval $[Q_{ih}, Q_{i(h+1)}]$ for which it belongs to is obtained. Thus, the normalized value X'_{gi} can be found by finding the linear interpolation between the points (Q_{Mh}, Q_{ih}) and $(Q_{M(h+1)}, Q_{i(h+1)})$.

Subsequently, the conditional T-test method was applied to select the 30 highest ranking genes as the input to the WNNs-based classifiers [42].

5.2 Multifold cross validation

Excessive training will force the WNNs to memorize the input vectors, and insufficient training will cause the WNNs will be unable to learn from the input vectors presented to them, and it will lead to poor generalization when new inputs are presented. Thus, in order to avoid these problems, multifold cross validation is used.

For the multifold cross validation, the samples are divided into k groups, where $k > 1$. One group from the samples is left out initially, where the training of the

WNNs involves the remaining samples. The training of WNNs will stop when an error goal of 0.01 is reached. Subsequently, the validation error is measured by testing on the groups left out, which are unseen during the training of WNNs. The process is repeated for k times, where i-th group will be testing samples during i-th execution, where $i = 1, 2, \ldots, k$. An average of the validation error is then calculated. In this study, a 10-fold cross validation is used.

5.3 Performance assessment

The statistical measure for the validity of the proposed WNNs is evaluated through three performance indexes, namely, the classification accuracy, sensitivity and specificity, which are formulated as:

$$accuracy = \frac{N_t}{N_{all}} \times 100\%, \tag{19}$$

$$sensitivity = \frac{TP}{TP + FN} \times 100\%, \tag{20}$$

and

$$specificity = \frac{TN}{TN + FP} \times 100\%, \tag{21}$$

where N_t is the number of testing samples that have been classified correctly, N_{all} is the total number of testing samples, TP is true positive, FN is false negative, TN is true negative and FP is false negative, respectively.

5.4 Results and discussion

The number of input nodes for WNNs is determined by the input dimension, while the number of hidden nodes is determined by the number of training samples, q, (for random initialization approach) or the number of cluster centers. In this work, the number of cluster centers is chosen to be approximately $\sqrt{q/2}$ [43]. The number of outputs is equal to 3 as there are three different subclasses for the Diabetes dataset. An error goal of 0.01 was specified as the stopping criterion and the experiments were run 10 times. The classification accuracy for the Diabetes dataset by varying the clustering algorithms and activation functions for WNNs-based classifier is presented in Tab. I. The sensitivity and specificity for each of the subclass, specifically the NGT, IGT and DM2, are presented in Tab. II to Tab. VII, respectively.

We investigate the correlation between the goodness of the types of wavelet families used and the WNNs classification capability. As shown in Tab. I, overall, when different clustering algorithms are employed in the initialization of the translation parameter, the WNNs with Gaussian wavelet outperform the models with Mexican Hat or Morlet.

Selection of the activation function in the WNNs depends on the form of the function that needs to be constructed [44]. However, the underlying function between the input-output spaces is usually unknown beforehand. Scrutinizing the problem at hand might provide hints in selecting the appropriate activation function for the WNNs. In the metaphase of disease progression, the full interaction of

Initialization Method	Wavelet Families		
	Mexican Hat	Gaussian Wavelet	Morlet
Random	93.02	95.35	93.02
KM	95.35	97.67	95.35
FCM	95.35	97.67	95.35
PSDKM	95.35	97.67	95.35
PSDFCM	97.67	97.67	95.35
MPSDKM	97.67	97.67	95.35
MPSDFCM	97.67	100	97.67

Tab. I *Performance comparison of WNNs for the Diabetes dataset (Accuracy).*

Initialization Method	Wavelet Families		
	Mexican Hat	Gaussian Wavelet	Morlet
Random	94.12	98.82	95.94
KM	97.65	98.82	95.29
FCM	92.94	95.29	92.94
PSDKM	96.47	100	96.47
PSDFCM	96.47	95.29	90.59
MPSDKM	100	100	95.29
MPSDFCM	100	100	95.29

Tab. II *Performance comparison of WNNs for the Diabetes dataset (Sensitivity: NGT).*

Initialization Method	Wavelet Families		
	Mexican Hat	Gaussian Wavelet	Morlet
Random	95.16	96.00	98.33
KM	93.85	96.92	95.38
FCM	96.92	99.23	97.59
PSDKM	94.62	96.15	96.15
PSDFCM	98.46	99.23	96.92
MPSDKM	95.15	96.15	95.38
MPSDFCM	96.15	100	99.23

Tab. III *Performance comparison of WNNs for the Diabetes dataset (Specificity: NGT).*

the gene expression profiling is a highly complex, nonlinear, stochastic and chaotic procedure. Thus, the shape of the chosen wavelet family must be able to interpret the inherent characteristics of this process. The WNNs with a Gaussian wavelet that has a great change in its shape will be a more reliable tool in capturing this type of complex interaction with abrupt variations compared to the WNNs with a Mexican Hat or Morlet wavelet that work efficiently in dealing with a sinusoidal

Initialization Method	Wavelet Families		
	Mexican Hat	Gaussian Wavelet	Morlet
Random	97.5	90	87.5
KM	97.5	100	97.5
FCM	92.5	100	95
PSDKM	97.5	100	95
PSDFCM	97.5	100	100
MPSDKM	100	100	97.5
MPSDFCM	100	100	97.5

Tab. IV *Performance comparison of WNNs for the Diabetes dataset (Sensitivity: IGT).*

Initialization Method	Wavelet Families		
	Mexican Hat	Gaussian Wavelet	Morlet
Random	98.16	100	98.82
KM	99.41	100	100
FCM	99.39	100	98.82
PSDKM	100	100	100
PSDFCM	100	99.41	99.39
MPSDKM	100	100	99.41
MPSDFCM	100	100	100

Tab. V *Performance comparison of WNNs for the Diabetes dataset (Specificity: IGT).*

Initialization Method	Wavelet Families		
	Mexican Hat	Gaussian Wavelet	Morlet
Random	90	94.44	95.56
KM	92.22	95.56	94.44
FCM	98.89	98.89	97.78
PSDKM	93.33	94.44	94.44
PSDFCM	98.89	98.89	95.56
MPSDKM	94.44	94.44	94.44
MPSDFCM	94.44	100	100

Tab. VI *Performance comparison of WNNs for the Diabetes dataset (Sensitivity: DM2).*

function with smoother variations. This might explain the superior results of the WNNs with the Gaussian wavelet compared to the rest.

Next, we evaluate the efficiency of the clustering algorithm with respect to the WNNs classification capability. Overall, employing the clustering algorithms in the parameter initialization leads to the increase in the classification capability of

Initialization Method	Wavelet Families		
	Mexican Hat	Gaussian Wavelet	Morlet
Random	95.2	96	91.2
KM	99.2	99.2	96.77
FCM	95.9	96.8	95.90
PSDKM	97.53	100	96
PSDFCM	97.57	97.57	94.37
MPSDKM	100	100	97.53
MPSDFCM	100	100	96.8

Tab. VII *Performance comparison of WNNs for the Diabetes dataset (Specificity: DM2).*

WNNs models. Inspecting the Tab. I, the WNNs-based classifiers with MPSDFCM yield the highest classification accuracy, which may be attributed to the existence of the symmetrical inter- and intra-clusters in the data. The WNNs without employing the clustering algorithm achieve the lowest classification rate of 93.02%, 95.25% and 93.02%.

When sensitivity and specificity analysis were taken to validate the performance of the proposed WNNs classifiers, the results indicate that the proposed WNNs approaches generate reasonable promising predictive aptitude. In particular, the sensitivity was within 94.12% to 100%, 87.5% to 100%, and 90% to 100%, whereas the specificity was within 93.85% to 100%, 98.16% to 100% and 91.2% to 100%, for NGT, IGT and DM2, respectively.

Particular interest is paid to the sensitivity analysis for IGT subgroup, since it is hardly diagnosed at the very initial stage. A classifier with high sensitivity would lead to the decrease of false negatives, where it implies that most probably the occurrence of IGT can be detected correctly. From Tab. IV, it can be observed that, except for the WNNs with random initialization more frequently, WNNs with Gaussian wavelet activation function have higher sensitivity than WNNs with Mexican Hat and Morlet activation functions. Thus, this indicates that whenever an IGT sample exists, most likely it will be classified correctly with an almost 100% confidence by WNNs with Gaussian wavelet. Specifically, predictive competence of WNNs with Gaussian wavelet was further improved, when the MPSDFCM initialization approach was taken into account, since it surpassed the rest in terms of classification accuracy, sensitivity and specificity, which can be observed in Tab. I to Tab. VII.

Finally, to validate the statistical significance of the proposed WNNs with Gaussian wavelet and MPSDFCM initialization algorithm, the standard normal statistic z-test was applied [45]. The null hypothesis is to postulate that the number of samples being classified correctly does not exceed the number of samples that would be classified correctly by chance.

Let N be the total number of sample, n_j be the dimension of the j-th subgroup, e be the overall chance frequency and o be the number of samples classified correctly.

The z statistical test is given as:

$$z = \frac{o - e}{\sqrt{e(N-e)/N}}, \tag{22}$$

where

$$e = \sum_{j=1}^{3} q_j n_j, \tag{23}$$

and

$$q_j = \frac{n_j}{N}. \tag{24}$$

In this case, we have $o = 43, e \approx 15.74, z \approx 8.63$ and the p-value ≈ 0. Thus, the null hypothesis was rejected at the confidence level of 5% and the predictive competence of the proposed model was authenticated from a statistical point of view.

6. Performance Comparisons

From the experimental results shown in Tab. I to Tab. VII, the highest classification accuracy, sensitivity and specificity achieved by WNNs with the integration of Gaussian wavelet and MPSDFCM is used to compare against the predictive capability of other neural network models. Two popular neural network models, namely, MLPs and RBFNNs are chosen for the comparison purpose.

For the network architecture of MLPs, a 38-3-3 network structure with gradient descent backpropagation learning algorithm was used, where the number of hidden nodes was determined by using the Blum's "rules of thumb" [46] and the learning rate was set to 0.01. The training continued until 100,000 epochs was reached within the error goal of 0.01. For the RBFNNs, a one-hidden layer network model with gradient descent technique was used, where the spread value was initialized as 1, and the training of RBFNNs stopped when it met the error goal of 0.01. The 10 runs simulations were completed for both of the MLPs and RBFNNs, and then the average value for the performance indexes was calculated.

The performance of MLPs, RBFNNs and WNNs in terms of classification accuracy, sensitivity and specificity is presented in Tab. VIII to Tab. X.

As shown in Tab. VIII, the highest classification rate on the diabetes dataset is achieved by the proposed WNNs (100%), followed by RBFNNs (90.70%) and

Neural Network Model	Classification Rate (%)	No. of Misclassify Sample
MLPs	88.37	5
RBFNNs	90.70	4
WNNs	100	0

Tab. VIII *Performance comparison of MLPs, RBFNNs and WNNs for the Diabetes dataset (Accuracy).*

Subclass	Classifier		
	MLPs	RBFNNs	WNNs
NGT	94.71	94.71	100
IGT	88.75	78.75	100
DM2	88.89	86.67	100

Tab. IX *Performance comparison of MLPs, RBFNNs and WNNs for the Diabetes dataset (Sensitivity).*

Subclass	Classifier		
	MLPs	RBFNNs	WNNs
NGT	88.08	84.23	100
IGT	91.71	90.57	100
DM2	92	89.60	100

Tab. X *Performance comparison of MLPs, RBFNNs and WNNs for the Diabetes dataset (Specificity).*

MLPs (88.37%). Examining from the aspect of sensitivity and specificity, it can be observed that the WNNs clearly outperformed the rest in distinguishing between the classes of NGT, IGT and DM2 with less false alarm and high certainty, as shown in Tab. IX and Tab. X. With regards to the sensitivity analysis for IGT, as shown in Tab. IX, RBFNNs have the lowest predictive power in identifying the occurrence of IGT, since they merely achieve sensitivity of 75.85%.

MLPs performed the worst; five samples are classified incorrectly. This unsatisfactory performance is probably due to the behavior of the MLPs. The sigmoid function used as the activation function in the MLPs is a globalized activation function. Compared with the localized wavelet function in the hidden layer of WNNs which lead to learning efficiency, the learning of the MLPs is time-consuming. In addition, the backpropagation learning algorithm used by MLPs tends to get trapped in local minima. When the MLPs are unable to converge to a global minimum, the prediction accuracy of the MLPs deteriorates.

Comparing the classification accuracy achieved by the proposed system and RBFNNs, the former seems to be a more appropriate classifier for the diabetes dataset. This might be caused by the activation function used in the hidden layer of RBFNNs and WNNs. Gaussian function is used in the hidden nodes of RBFNNs. However, a periodic function and an exponential function are approximated better by using a WNN with an oscillating wavelet activation function and by a Gaussian activation function respectively. Picturing the gene expression profiles of DM2 is a highly complex and stochastic process. The behavior of the oscillating wavelet activation function captures this chaotic process better, compared with Gaussian function.

7. Conclusions

The full gene picture of the gene expression profiling in DM2 is a pretty complex procedure, and it varies drastically from patients to patients even at the similar disease stage. There are many detectable and undetectable factors that are contributed to the evolving of the disease; the interactions between these factors are highly complex, nonlinear, stochastic and chaotic. From the experimental results, WNNs seem to be a suitable tool in capturing the input-output behavior of the gene expression profiling without knowing the involved explicit internal process; where the percentage of correct classifications ranged from 93.02% to 100%. The developed MPSDFCM algorithm outperformed the other WNNs at overall, regardless of random initialization or the use of KM, FCM, PSDKM, PSDFCM of MPSDKM algorithms. Thus, the beneficial potential of this proposed method is promising. The performance of the WNNs with different wavelet families as the activation function in the hidden layer for classification were tested and compared; there was no significant variation and all of the models performed well.

When a performance assessment was made between the proposed approach and the popular neural network models, i.e. MLPs and RBFNNs, the recognition rate of the proposed method outperformed the rest. Thus, the categorization effectiveness of the proposed WNNs models is promising. Generally, the WNNs with Gaussian wavelet and MPSDFCM seem to outperform the rest.

In future works, the pathway between clusters of genes should also be considered, since genes with low significant values from conditional T-test might have strong discriminative power when they couple together. Therefore, not only genes at the highest ranking are selected.

References

[1] Zhang Q. G., Benveniste A.: Wavelet Networks. IEEE Trans. Neural Network, **3**, 1992, pp. 889-898.

[2] Moddy J., Darken C. J.: Fast learning in network of locally-tuned processing units. Neural Comput., **1**, 1989, pp. 281-294.

[3] Cao J., Lin X.: Application of the diagonal recurrent wavelet neural network to solar irradiation forecast assisted with fuzzy technique. Eng. Appl. Artif. Intel., **21**, 2008, pp. 1255-1263.

[4] Zainuddin Z., Ong P.: Modified wavelet neural network in function approximation and its application in prediction of time-series pollution data. Appl. Soft Comput., **11**, 2011, pp. 4866-4874.

[5] Zainuddin Z., Wan Daud W. R., Ong P., Shafie A.: Wavelet Neural Networks Applied to Pulping of Oil Palm Fronds. Bioresource Technol., **102**, 2011, pp. 10978-10986.

[6] Cui W. Z., Zhu C. C., Zhao H. P.: Prediction of thin film thickness of field emission using wavelet neural networks. Thin Solid Films, **473**, 2005, pp. 224-229.

[7] Gutierrez J. M., Gutes A., Cespedes F., Valle M., Munoz R.: Wavelet neural networks to resolve the overlapping signal in the voltammetric determination of phenolic compounds. Talanta, **76**, 2008, pp. 373-381.

[8] Gutierrez J. M., Moreno-Baron L., Valle, M., Leija L., Munoz R.: Wavelet Neural Network as a Multivariate Calibration Method in Voltammetric Electronic Tongues. Neural Network World, **19**, 2009, pp. 53-64.

[9] Zhong S. S., Li Y., Ding G., Lin L.: Continuous wavelet process neural network and its application. Neural Network World, **17**, 2007, pp. 483-495.

[10] Zainuddin Z., Ong P.: Improved wavelet neural networks for early cancer diagnosis using clustering algorithms. Int. J. Comput. Intell., **6**, 2010, pp. 33-39.

[11] Zainuddin Z., Ong P.: Improved Wavelet Neural Network for Early Diagnosis of Cancer Patients Using Microarray Gene Expression Data. In: Proc. of the IEEE Int. Joint Conf. on Neural Network, IEEE Press, NY, pp. 2663-2670.

[12] Ubeyli E. D.: Wavelet/probabilistic neural networks for ECG beats classification. Neural Network World, **17**, 2007, pp. 255-267.

[13] Biswal B., Dash P. K., Panigrahi B. K., Reddy J. B. V.: Power signal classification using dynamic wavelet network. Appl. Soft Comput., **9**, 2009, pp. 118-125.

[14] Avci E., Avci D.: The performance comparison of discrete wavelet neural network and discrete wavelet adaptive network based fuzzy inference system for digital modulation recognition. Expert Syst. Appl., **35**, 2008, pp. 90-101.

[15] Chen Y. H., Yang B., Dong J. W.: Time-series prediction using a local linear wavelet neural network. Neurocomputing, **69**, 2006, pp. 449-465.

[16] Lin C. J., Tsai H. M.: FPGA implementation of a wavelet neural network with particle swarm optimization learning. Math. Comput. Model., **47**, 2008, pp. 982-996.

[17] Li H., Wan B.: Multi-Input-Layer Wavelet Neural Network and its Application. In: Proc. of the 5^{th} Int. Conf. Comput. Intell. Multimedia Appl., IEEE Computer Society, NY, pp. 468.

[18] Mohd Idris M. F., Dahlan Z. A., Jusoff H. K.: The performance of two mother wavelets in function approximation. J. Math. Res., **1**, 2009, pp. 135-143.

[19] Bellil W., Ben Amar C., Alimi A. M.: Comparison between Beta Wavelets Neural Networks, RBF Neural Networks and Polynomial Approximation for 1D, 2D Functions Approximation. Trans. Eng. Comput. Technol., **13**, 2006, pp. 102-107.

[20] Lin C.-J.: Nonlinear systems control using self-constructing wavelet networks. Appl. Soft Comput., **9**, 2009, pp. 71-79.

[21] Oussar Y., Dreyfus G.: Initialization by selection for wavelet network training. Neurocomputing, **34**, 2000, pp. 131-143.

[22] Zhang X. G., Kuang Y. Z., Geng D. H.: Fuzzy wavelet neural networks based on SVM. J. East Chin. Univ. Sci. Technol., **32**, 2006, pp. 1351-1354.

[23] Tzeng S. T.: Design of fuzzy wavelet neural networks using the GA approach for function approximation and system identification. Fuzzy Set Syst., **161**, 2010, pp. 2585-2596.

[24] Su H. Z., Wu Z. R., Wen Z. P.: Identification model for dam behavior based on wavelet network. Comput-Aided Civ. Inf., **22**, 2007, pp. 438-448.

[25] Ling S. H., Iu H. H. C., Leung F. H. F., Chan K. Y.: Improved hybrid particle swarm optimized wavelet neural network for Modeling the development of Fluid Dispensing for Electronic Packaging. IEEE Trans. Ind. Electron., **55**, 2008, pp. 3447-3460.

[26] Cao J. W., Lin Z. P., Huang G. B.: Composite function wavelet neural networks with extreme learning machine. Neurocomputing, **73**, 2010, pp. 1405-1416.

[27] Srivastava S., Singh M., Hanmandlu M., Jha A. N.: New fuzzy wavelet neural networks for system identification and control. Appl. Soft Comput., **6**, 2005, pp. 1-17.

[28] Zhang D. H., Bi Y. Q., Bi Y. B., Sun Y. T.: Design and initialization algorithm based on modulus maxima of wavelet transform for wavelet neural network. In: Prof. of the 2004 Int. Conf. Power Syst. Technol., IEEE Press, NY, pp. 897-901.

[29] Wei M., Jin S., Wang L., Zhou Y.: Defect Characteristic Prediction of Pipeline by Means of Wavelet Neural Network Based on the Hierarchical Clustering Algorithm. In: Prof. of the Biennial Int. Pipeline Conf., ASME Press, NY, pp. 921-924.

[30] Hwang K., Mandayam S., Udpa S. S., Udpa L., Lord W., Atzal M.: Characterization of gas pipeline inspection signals using wavelet basis function neural networks. NDT&E Int., **33**, 2000, pp. 531-545.

[31] Kim S. J., Kim Y. T., Seo J. Y., Jeon H. T.: Design of the scaling-wavelet neural network using genetic algorithm. In: Proc. of the Int. Joint Conf. on Neural Networks, IEEE Press, NY, pp. 2174-2179.

[32] Su M. S., Chou C. H.: A modified version of the K-means algorithm with a distance based on cluster symmetry. IEEE Trans. Pattern Anal., **23**, 2001, pp. 674-680.

[33] Su M. C., Chou C. H., Hsieh C. C.: Fuzzy C-means algorithm with a point symmetry distance. Int. J. Fuzzy Syst., **7**, 2005, pp. 175-181.

[34] Chung K. L., Lin J. S.: Faster and more robust point symmetry-based K-means algorithm. Pattern Recogn., **40**, 2007, pp. 410-422.

[35] Kahramanli H., Allahverdi N.: Design of a hybrid system for the diabetes and heart diseases. Expert Syst. Appl., **35**, 2008, pp. 82-89.

[36] Arulampalam G., Bouzerdoum A.: Application of Shunting Inhibitory Artificial Neural Networks to medical diagnosis. In: Proc. of the Seventh Australian and New Zealand Intell. Inform. Syst. Conf., IEEE Press, NY, pp. 89-94.

[37] Watkins A., Boggess L.: A new classifier based on resource limited artificial immune systems. IEEE Comput. Evol. Comput., 2002, pp. 1546-1551.

[38] Zhou Z. H., Jiang Y.: Medical diagnosis with C4.5 rule preceded by artificial neural network ensemble. IEEE Trans. Inf. Technol. Biomed., **7**, 2003, pp. 37-42.

[39] Polat K., Salih G., Arslan A.: A cascade learning system for classification of diabetes disease: Generalized Discriminant Analysis and Least Square Support Vector Machine. Expert Syst. Appl., **34**, 2008, pp. 482-487.

[40] Othman M. F., Yau T. M. S.: Neuro fuzzy classification and detection technique for bioinformatics problems. In: Prof. of the First Asia Int. Conf. on Model. Simul., IEEE Press, NY, pp. 375-380.

[41] Zainuddin Z., Ong P.: Reliable multiclass cancer classification of microarray gene expression profiles using an improved wavelet neural network. Expert Syst. Appl., **38**, 2011, pp. 13711-13722.

[42] Amaratunga D., Cabrera J.: Exploration and analysis of DNA microarray and protein array data. New Jersey: John Wiley & Sons, 2004.

[43] Kanti V. M., Kent J. T., Bibby J. M.: Multivariate analysis. London: Academic Press, 2000.

[44] Ramuhalli P., Udpa L., Udpa S. S.: Electromagnetic NDE signal inversion by function-approximation neural networks. IEEE Trans. Magn., **38**, 2002, pp. 3633-3642.

[45] Braglia M., Carmignani G., Frosolini M., Zammori F.: Data classification and MTBF prediction with a multivariate analysis approach. Reliab. Eng. Syst. Saf., **97**, 2012, pp. 27-35.

[46] Blum A.: Neural networks in C++. New York: John Wiley & Sons, 1992.