

International Journal of Pattern Recognition and Artificial Intelligence
© World Scientific Publishing Company

FAST LEARNING ALGORITHM OF WAVELET NETWORK BASED ON FAST WAVELET TRANSFORM

OLFA JEMAI*

Higher Institute of Computer Sciences, Road El Jorf, Km 22.5 - 4119 Medenine, Tunisia
REGIM: REsearch Group on Intelligent Machines, University of Sfax, National Engineering
School of Sfax (ENIS)
BP 1173, Sfax, 3038, Tunisia
olfa.jemai@ieee.org

MOURAD ZAIED†, CHOKRI BEN AMAR‡
and MOHAMED ADEL ALIMIS§

REGIM: REsearch Group on Intelligent Machines, National Engineering School of Sfax (ENIS)

†mourad.zaied@ieee.org

‡Chokri.benamar@ieee.org

§adel.alimi@ieee.org

In this paper, a novel learning algorithm of wavelet networks based on the Fast Wavelet Transform (FWT) is proposed. It has many advantages compared to other algorithms, in which we solve the problem of the previous works, when the weights of the hidden layer to the output layer are determinate by applying the back propagation algorithm or by direct solution which requires to compute matrix inversion, this may be intensive computation when the learning data is too large. However, the new algorithm is realized by the iterative application of FWT to compute connection weights. Furthermore, we have extended the novel learning algorithm by using the Levenberg-Marquardt method to optimize the learning functions. The experimental results demonstrated that our model is remarkably more refreshing than some of the previously established models in terms of both speed and efficiency.

Keywords: Wavelet network, Fast wavelet transform, Learning algorithm.

1. Introduction

Neural networks have been supplied with an access to the frequency analysis with the introduction of wavelets. A wavelet network has more advantages than common networks, e.g., faster convergence, avoiding local minimum, easy decision and adaptation of the structure.^{6,37,38} Usually, there are two kinds of wavelet networks, which are constructed from different ideas.⁸ One is to take the wavelet network as a special RBF network.^{29,30} Moreover, the training of this kind of wavelet networks is

*Corresponding author. Address: 9 street of General Degaul, Gabes 6000, Tunisia.
Tel: 216-98-660360

to some extent similar to that of RBF networks. The other kind of wavelet networks is constructed from the wavelet theory.

With the development of wavelet networks, many research papers are devoted to the learning algorithms of wavelet networks. The back-propagation error may be the most popular algorithm in the learning of wavelet networks.^{29,30,20} In the course of training, the error back-propagation is often combined with orthogonal least square-backward elimination,^{29,30} which is used in the selection of network structures. At the same time, many algorithms are proposed to initialize the weights,²⁶ accelerate the convergence^{37,19} and adjust the structures of wavelet networks³⁷ when the error back-propagation is applied to the training algorithms. In addition to the error back-propagation, other algorithms were also proposed to train the wavelet networks. These algorithms include the Kalman filter,³¹ genetic algorithms^{5,21,34} and immune algorithms.²⁴ The above mentioned algorithms mostly stem from typical neural networks. They make use of the strong points of wavelets in the frequency domain. As results, they accelerate convergence, avoid local minimum and overcome overfitting to some extent.

Zhang and Benveniste,³⁹ Pati and Krishnaprasad²⁷ were the first to advance the idea of applying wavelets into neural network to build the so-called "wavelet neural network". Taking advantages of both the time-frequency zooming property of wavelets and the effective learning mechanism of neural networks, wavelet networks are becoming a powerful tool for many applications.^{14,33,28} For function learning which is the fundamental and vital problem in many fields, Zhang built an orthogonal wavelet network⁴⁰ in order to solve the redundancy and non-orthogonal problem of basis in MLP and RBF neural networks, and ultimately made use of network learning problems to solve a set of linear equations. His model provides a unique and effective expression to a given function. In addition, it preserves many of the strong points of RBF network. This model was proved to have appealing properties and to have a wide variety of applications. However, we believe that there are at least two problems as far as Zhang's work is concerned. First, Zhang et al focused on the solution of the set of linear equations in which the calculations become larger when the scale of problems gets larger.⁴⁰ This involves the inversion of a large matrix, resulting in an intensive computations. The authors suggested the use of gradient descent algorithm which is not well favored in most cases, because it is not effectively performing in real time applications and easily falls into the shortcoming of local minima. Presently the question that needs to be answered is whether there are other alternatives, or if there is a better method for avoiding huge calculations. Secondly, to the whole function, we can build the orthogonal wavelet network for non-redundant approximation similar to that proposed in⁴⁰. Furthermore, to solve the problem, we can employ algorithms for structural adaptation of the wavelet network for the elimination of unnecessary wavelons, similar to what references³³ and³² proposed or for the sparseness of large-dimensional training data. In particular, wavelets whose supports do not contain any data point are eliminated from the training process.⁴¹ However, for the part of the function, the wavelet networks

built with the methods stated above still have structural-redundancy. This means that not all neurons are needed to the part of the function. Presently, the wavelet network model successfully maintains the favorable advantages of the RBF neural network structure. Most researchers concentrate on the algorithm and the study of the wavelet network theory.^{35,36,3,16}

In this paper, we propose a new algorithm for the learning of wavelet networks constructed from multiresolution analysis (MRA) (multiresolution approximation) by application of fast wavelet transform.

The remainder of this paper is composed of four sections. Section 2 briefly reviews the theory of wavelet networks and the fast wavelet transform theory. The fast wavelet transform theory is applied to the training of a dyadic wavelet networks and the new algorithm is proposed in section 3. Also this section encompasses an extended description of the new algorithm briefly described in section 3, by optimizing selected wavelets that will be used as transfer function in the hidden layer of the network. Experiments, conducted to demonstrate the effectiveness of the optimized wavelet network and point out its advantages are described in section 4. Finally, section 5 summarizes and concludes this paper.

2. Review of wavelet transform and wavelet network

2.1. Continuous Wavelet Transform (CWT)

For a given function $f \in L^2(R)$, the continuous wavelet transform is given by:

$$\omega_{a,b} = \frac{1}{\sqrt{|a|}} \int f(t)\psi\left(\frac{t-b}{a}\right)dt = \langle \psi_{a,b}, f \rangle . \quad (1)$$

All analyzed wavelets are generated from one mother wavelet by varying the frequency parameter (a) and time or translation parameter (b). The corresponding inverse wavelet transform that rebuilds the function f from its coefficients of wavelets is:

$$f(t) = \frac{1}{C_\psi} \iint \omega_{a,b} \frac{1}{\sqrt{|a|}} \psi\left(\frac{t-b}{a}\right) \frac{dadb}{a^2} . \quad (2)$$

The coefficient C_ψ , which is equal to $2\pi \int \frac{\|\hat{\psi}(\omega)\|^2}{\|\omega\|} d\omega$, where $\hat{\psi}$ is the Fourier transform of ψ , must be different from zero and ∞ so that this inverse transform exists. This condition, always known as the admissibility condition, must be verified at the beginning of any wavelet transform.

Knowing that C_ψ is always positive, the admissibility condition can be written as follow:

$$0 < C_\psi < \infty . \quad (3)$$

2.2. Discrete Wavelet Transform (DWT)

To analyze numerical signals, the CWT is very redundant. Thus, in this case we use a set of wavelets generated by considering only sampled values of (a) and (b).

4 *O. Jemai et al.*

Here, a sampling on a time-frequency grid is adopted.

To maintain a relation between the parameters (a) and (b), we sample them in the following manner: $a = a_0^j, b = nb_0 a_0^j$ with $a_0 > 1$ and $b_0 > 0$. Thus, for analyzing a signal containing a_0^j points, we only use the family wavelets:

$$\psi(a_0^{-j}t - nb_0) \quad \text{with } j = 1, \dots, m \quad n = 1, \dots, a_0^{m-j}. \quad (4)$$

In the case of DWT, the analyzing wavelets are characterized by two new parameters: the position parameter n and the scale parameter j ($1 \leq j \leq m$, m represent the number of scales). The analyzing formula can be written by:

$$\omega_{j,n} = \frac{1}{\sqrt{a_0^j}} \sum f(t) \psi(a_0^{-j}t - nb_0). \quad (5)$$

And the inverse wavelet transform is:

$$f(t) = \frac{1}{C_\psi} \sum \sum \omega_{j,n} \frac{1}{\sqrt{a_0^j}} \psi(a_0^{-j}t - nb_0). \quad (6)$$

For the particular case, when $a_0 = 2, b_0 = 1$, the sampling is called dyadic and the resulting set of wavelets constitutes an orthogonal or a biorthogonal basis.

2.3. *MutiResolution Analysis (MRA)*

The multiresolution analysis consists of, firstly, a scaling function $\phi(x) \in L^2(\mathbb{R})$, which constitutes an orthonormal basis by varying its position on a given scale (j). The functions of every scale generate an approximation of a given signal f for analysis. Secondly, additional functions (wavelet functions) are then used to encode the difference in information between adjacent approximations.⁹

For more details, we consider $\phi(x)$ a scaling function that encounters an orthonormal basis by translation on a space V_j . The approximation of the signal f on V_j is:

$$A_j = \sum_n a_n^j \phi_{j,n}. \quad (7)$$

The a_n^j coefficients are computed by applying the scalar product:

$$a_n^j = \langle f, \phi_{j,n} \rangle. \quad (8)$$

An orthogonal complement space W_{j+1} exists and belongs to V_{j+1} , which is subjected to the following condition:

$$V_{j+1} \subset V_j \Rightarrow V_j = V_{j+1} \oplus W_{j+1}. \quad (9)$$

A set of dyadic wavelet functions derived by translations and the same dilation of one mother wavelet exists that encounters a basis of W_{j+1} .

The detail of f in W_{j+1} space is given by:

$$D_j = \sum_n d_n^j \psi_{j,n}. \quad (10)$$

The detail coefficients d_n^j are computed by the projection of the signal f on the family wavelets $\psi_{j,n}$:

$$d_n^j = \langle f, \psi_{j,n} \rangle. \quad (11)$$

Since the approximation spaces V_j are nested, the approximation signal A_j can be analyzed several times in a multiresolution scale. At a given scale j , the signal f can be written as:

$$f = \sum_k a_k^j \phi_{j,k} + \sum_i \sum_k^j d_k^i \psi_{i,k}. \quad \text{with } j \leq m \quad (12)$$

The analysis can be iterated until the last scales, and thus, f can be written as:

$$f = A_n + D_n + \dots + D_2 + D_1. \quad (13)$$

In general, when $\phi_{j,n}$ constitutes a non orthogonal basis, it is necessary to compute first the dual basis composed by the functions $\tilde{\phi}_{j,n}$ of the scaling functions $\phi_{j,n}$ to be able to calculate the a_n^j coefficients. A dual basis of a set of functions z_j can be computed by:²²

$$\tilde{z}_j = \sum_{i=1}^N (Z)_{j,i}^{-1} z_i. \quad \text{with } Z_{j,i} = \langle z_j, z_i \rangle \quad (14)$$

Using this formula, we compute the dual family of the scaling functions $\tilde{\phi}_{j,n}$ and dual family of wavelets $\tilde{\psi}_{j,n}$.

The approximation of f at the scale j and the position n is obtained by:

$$a_n^j = \langle f, \tilde{\phi}_{j,n} \rangle. \quad (15)$$

We can compute the dual family of wavelets and compute the detail coefficients using the same principle:

$$d_n^j = \langle f, \tilde{\psi}_{j,n} \rangle. \quad (16)$$

Figures 1 and 2 represent the decomposition steps of signal f using dual wavelets and scaling functions $(\tilde{\psi}_{j,n}, \tilde{\phi}_{j,n})$ and the reconstitution steps with primal wavelets and scaling functions $(\psi_{j,n}, \phi_{j,n})$:

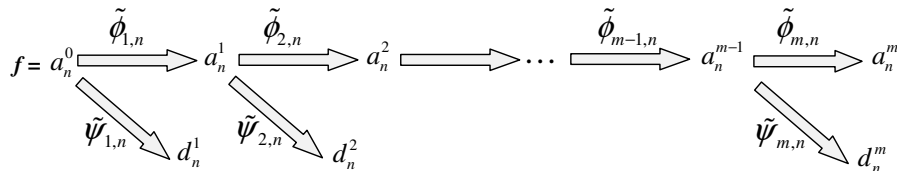


Fig. 1. Multiresolution analysis of a signal f .

6 *O. Jemai et al.*

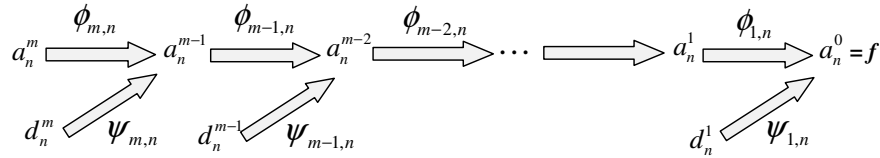


Fig. 2. Reconstitution of the signal f using all the details and the last approximation.

It remains to point out that we can use primal wavelets and scaling functions $(\psi_{j,n}, \phi_{j,n})$ to analyze a signal and the dual set $(\tilde{\psi}_{j,n}, \tilde{\phi}_{j,n})$ to compute the reconstruction.

2.4. Wavelet network

In 1992, Zhang and Benveniste introduced a new theory called "wavelet networks" using a combination of artificial neural networks based on radial basis function and wavelet decomposition. In addition, they explained how a wavelet networks can be generated and showed how they can be used for pattern matching. Zhang showed that wavelet networks are able to handle nonlinear regression of moderately large input dimension with sparse training data. Moreover, he replaced the transfer function by an admissible wavelet and demonstrated that a wavelet network preserves the property of universal approximation of the RBF networks.

A wavelet network is defined by pondering a set of wavelets dilated and translated from one mother wavelet with weight values to approximate a given signal f :

$$\tilde{f} = \sum_{i=1}^n \omega_i \psi_i. \tag{17}$$

The corresponding architecture is given in Fig. 3:

This architecture can be extended by adding dilated and translated versions of the scaling function of the corresponding wavelet used in the hidden layer of the network. In this case, the approximation of the signal is:

$$\tilde{f} = \sum_{i=1}^p \omega_i \psi_i + \sum_{j=1}^q \nu_j \phi_j. \tag{18}$$

A neural representation of this type of wavelet network is shown in Fig. 4:

In general, to learn a wavelet network, the gradient descent algorithm is adopted. As it is evident, this algorithm is not well favored in most cases because it performs poorly in real time applications and easily falls into the shortcoming of local minima.

To compute the output weight connections of the wavelet network, many studies^{22,18} proposed the use of the technique of the projection of the signal f on the dual basis of the wavelets and the scaling functions of the hidden layer. This learning

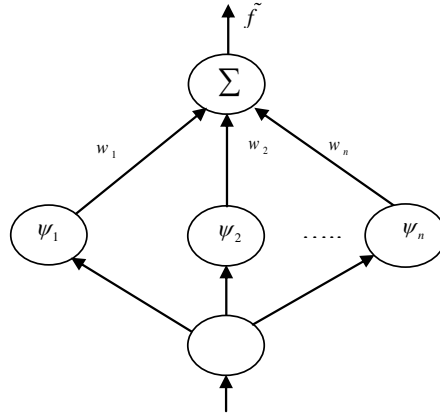


Fig. 3. Wavelet network.

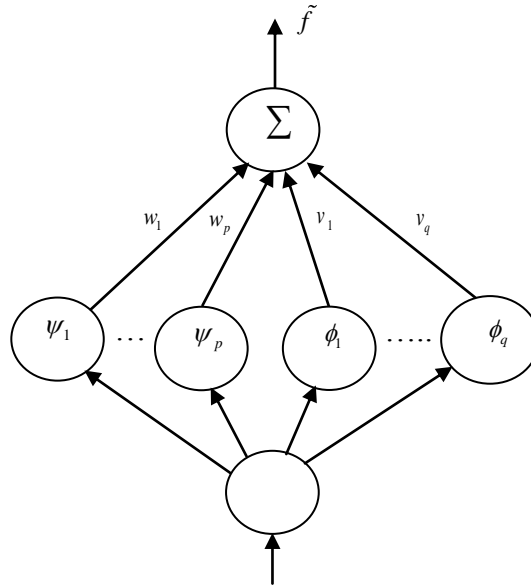


Fig. 4. Wavelet network scaling and wavelet functions based.

technique gives exact values of the weights, but it has a major shortcoming when we determine the weights of the hidden layer to the output layer which necessitates the computation of the inversion of a matrix Φ (Eq.14). The direct solution causes an intensive computation as the matrix becomes too large.

In section 3, we will propose a method that calculates rapidly the connection weights of a dyadic wavelet network. Then, it will be generalized to all types of wavelet networks. This method is based on the Fast Wavelet Transform (FWT). In what follows, we review relevant FWT theories, and then we detail the proposed learning algorithm.

2.5. Fast Wavelet Transform (FWT)

The FWT provides a simple way to handle the multiresolution analysis. In other words, with the FWT we wish to compute the approximation coefficients a_n^j (Eq. 15) and the detail coefficients d_n^j (Eq. 16) with other techniques that are simpler and easier than those based on the projection on the dual basis.

Bearing in mind that the approximation spaces V_j are nested, and in particular for V_1 and V_0 , we have $V_1 \subset V_0$, for every scaling function ϕ belonging to V_1 we get the following expression:

$$\phi(x) = \sum_n h[n] \phi_{0,n}(x). \quad \text{with } h[n] = \langle \phi(x), \phi_{0,n}(x) \rangle \quad (19)$$

Therefore, we demonstrate that the projection of f on V_{j+1} can be computed using the projection of f on V_j with the formula:⁴

$$a_n^{j+1} = \sum_k h[k] a_k^j. \quad (20)$$

The h coordinates (Eq. 20) are known as the low-pass filter or the scaling function filter.

Using equation (20) we compute the approximation coefficients of the scale $j+1$ via the approximation coefficients of the scale j and the low-pass filter h .

In addition, for the corresponding wavelet $\psi \in W_1$ of the scaling function ϕ , we have $W_1 \subset V_0$, thus, it can be decomposed on V_0 as:

$$\psi(x) = \sum_n g[n] \phi_{0,n}(x). \quad (21)$$

We can see that the detail coefficients of scale $j+1$ can be computed using the approximation coefficients of scale j .

$$d_n^{j+1} = \sum_k g[k] a_k^j. \quad (22)$$

The filter g is called the high-pass filter or the wavelet filter. Now, we can apply the MRA using only the h and g filters and their dual filters for the reconstruction and decomposition steps, respectively (see Figs. 5 and 6).

With this type of wavelet analysis, we can inverse the roles of the primal and dual filters, i.e. we can use the primal and dual filters to, respectively, decompose and reconstruct a signal.

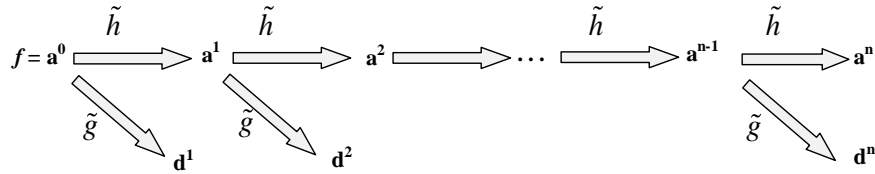


Fig. 5. Multiresolution decomposition of a signal f using dual filter bank.

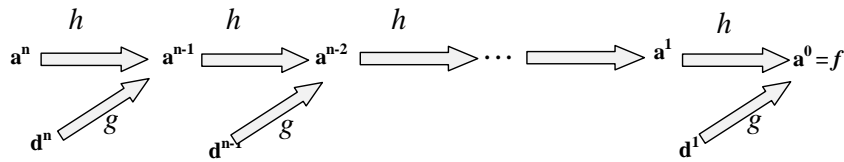


Fig. 6. Reconstitution of the signal f using primal filter bank.

In order to speed-up the calculation of the approximation and the detail coefficients, algorithms for rapid decomposition and reconstitution using filter bank are developed. Known by FWT, these algorithms decrease remarkably the time needed for the of decomposition and reconstitution steps. This figure presents the principle of FWT.

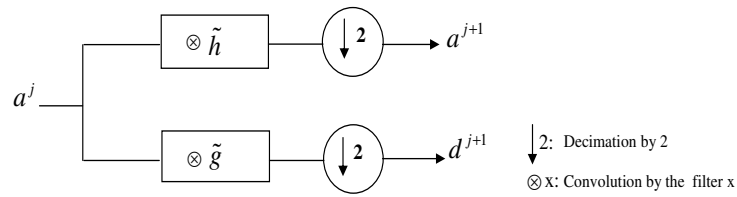


Fig. 7. Rapid decomposition with filter bank.

To get an approximation of the signal f at scale $j + 1$, the approximation a^j at scale j is convoluted by the dual filter \tilde{h} . To this end, the resulting signal is decimated by 2 (the coefficients with odd index are eliminated) to get the approximation coefficients a^{j+1} . The same steps are repeated using the dual filter \tilde{g} instead of \tilde{h} to get the details coefficients d^{j+1} .

As mentioned earlier, the approximation signal at scale $j + 1$ can also be analyzed. Therefore, we apply the same algorithm to get a^{j+2} and d^{j+2} coefficients.

The process can be iterated to analyze the signal at finer scales.

An inverse algorithm to reconstruct rapidly the approximation signal at scale j exists using the coefficients a^{j+1} and d^{j+1} . In this study, we use the FWT to achieve the decomposition process of a given signal, and the inverse wavelet transform (Eq. 12) to reconstruct it.

3. Learning wavelet network using FWT

In this section we will show how we can learn a wavelet network using only the FWT technique. Firstly, the proposed learning algorithm will be limited to wavelet network using dyadic wavelets and scaling functions as activation functions for the sake of simplicity. Secondly, in essence, the result of this proposed algorithm will be extended to use even non-dyadic wavelets and scaling functions.

3.1. Learning dyadic wavelet network using DWT basis

Looking carefully to the equation 12; if we replace, the double sum $\sum_i^j \sum_k d_k^i \psi_{i,k}$ used to compute the detail part of the signal f by only one sum, we can easily correspond it to the equation 18 and specially we can correspond the ω_i coefficients to the d_k^i ones and the ν_j coefficients to the a_k^j ones. Of course, this correspondence is available only if the considered wavelets and scaling functions are dyadic. We have seen also in the section 2.5 that the coefficients d_k^i and a_k^j can be computed by applying a series of FWT to the signal f .

Assuming that wavelets and scaling functions, used in the hidden layer of the wavelet network are dyadic, the kernel of the current problem lies in how the connection weights will be determined? Therefore, with our method, which consists on decomposing the signal by FWT, the obtained coefficients are matching the connection weights that will be used in the wavelet network.

3.1.1. Proposed learning algorithm

Step 1: Start the learning by preparing a library of candidate wavelets and scaling functions to be selected as activation functions of the wavelet network. This step includes the following items:

- (1) Choose the mother wavelet covering all the support of the signal to analyze.
- (2) Build a library formed by the wavelets and scaling functions used for computing the dyadic inverse wavelet transform.

Step 2: Compute the coefficients d_k^i and a_k^j corresponding to the wavelets and the scaling functions of the mentioned library by applying a FWT to the signal f to be learned using a dual set of wavelet and scaling function filters (\tilde{h}, \tilde{g}) .

- Step 3:** Compute the contribution of all the library activation functions ($d_k^i \psi_{i,k}$ and $a_k^j \phi_{j,k}$) in the reconstruction of the signal f which are based on the functions of the library and their corresponding weights (d_k^i and a_k^j).
- Step 4:** Set as a stop learning condition, an error E between the signal f and the output of the network.
- Step 5:** Select the activation function from the library that gives the best approximation of the signal f in the output of the network at each iteration (the learning algorithm is an incremental one). At the beginning an activation function g_1 is selected from the library, this function must minimize the error E_1 :

$$E_1 = \min_{\alpha_1 g_1} \left(\frac{1}{2} (f - \alpha_1 g_1)^2 \right). \quad (23)$$

Here g_1 is a function that can be a wavelet or a scaling function belonging to the library, the α_1 is the corresponding weight (d_k^i or a_k^j) and $\alpha_1 g_1$ is the contribution of g_1 in the reconstruction of the signal f . Since the contributions of all the library functions in the reconstruction of f are known, the selection of g_1 becomes easy. At an iteration n , an activation function g_n is selected with the same manner:

$$E_n = \min_{\alpha_n g_n} \left(\frac{1}{2} \left(f - \sum_{i=1}^{n-1} \alpha_i g_i - \alpha_n g_n \right)^2 \right). \quad (24)$$

At this step there is $(n-1)$ activation functions already selected, the minimization of the error E_n depends only on the selection of the function g_n and on the value of its corresponding weight α_n . We can observe that at an iteration n , we chose a function g_n from the library that approximate the best the residual error signal E_{n-1} of the precedent iterations.

- Step 6:** Calculate the output of the network, which is based on the hidden layer functions and the corresponding connection weights.
- Step 7:** If the error E is reached then it is the end of learning, else another function of the library is selected and we return to Step 5. The error E is calculated at the iteration n , with the formula:

$$E = \frac{1}{2} \left(f - \sum_{i=1}^n \alpha_i g_i \right)^2. \quad (25)$$

3.1.2. Creation of the library of wavelets and scaling functions

To construct the library of wavelets and scaling functions candidate to join our wavelet network, a sampling on a dyadic grid of dilation and translation parameters is preceded.

If the length of the signal f is equal to L , this sampling gives in the first scale $L/2$ wavelets. Every time that we climb a scale, the number of wavelets is divided by two. We can stop the sampling at any scale k with $k \leq m$ (Eq. 4) but we must

complete the library by the corresponding scaling functions of the last scale. The number of functions in this library is equal to L .

To have maximum coefficients characterizing a signal f in the wavelet space it's recommended to go to the last possible scale of the MRA.

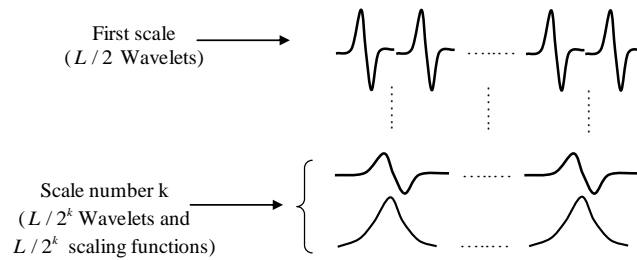


Fig. 8. The library wavelets and scaling functions after k MRA scales.

3.1.3. Calculation of the weights

To compute the weights corresponding to the activation functions that belong to the library, we analyze k (k is the number of the scales) times the signal f with the dual filters (\tilde{h}, \tilde{g}) of the scaling function $\tilde{\phi}$ and the wavelet $\tilde{\psi}$. The result is a sum of signals d^i corresponding to the weights of the wavelets of the scales i ($i = 1 \dots k$) and the weights a^k of the scaling functions of the scale k .

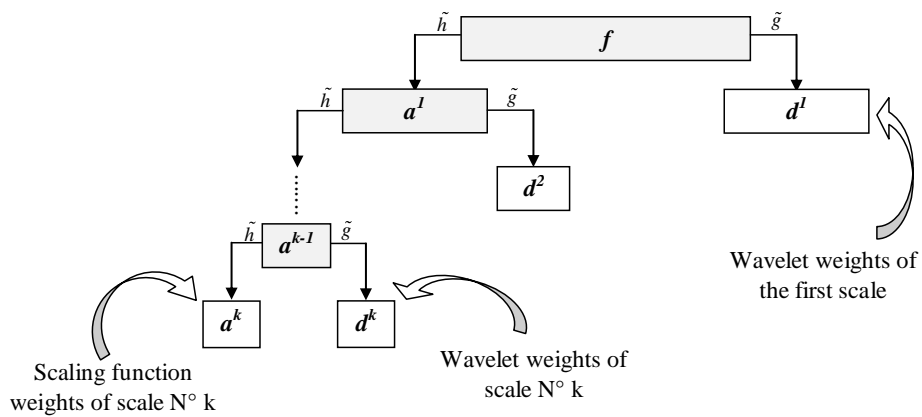


Fig. 9. The calculation process of the weights.

3.2. Extension of the proposed learning algorithm to use optimized library functions

With the proposed algorithm we can achieve an acceptable efficiency of learning, but one may ask: Can we use, as activation functions, any wavelets or any scaling functions that don't belong to the library and that approximate more the residual error signal at every iteration of the proposed learning algorithm?

This section shows how to exploit the FWT to compute the connection weights linking the psi set and the connection weights linking phi set to the output layer when these activation functions (wavelet or scaling one) do not belong to the prepared library shown in section 3.1.2. This approach can be generalized to be used with any set of activation functions. In our study in particular, we have chosen to increase the efficiency of the previously proposed learning algorithm by optimizing the dilations and translations of every selected function from the library, so we obtain functions that do not belong to the library and that do not correspond to the coefficients d_k^i and a_k^j were already computed. A newly-shaped version of the learning algorithm and a process to compute the new values of weights will be presented in details in the next section.

3.2.1. Optimized version of the learning algorithm

Step 1: Build the library of the candidate activation functions as mentioned in the precedent algorithm. Let's nominate those functions g_l with $l = 1 \cdots L$.

Step 2: : Set the stop condition of the learning as mentioned in the precedent version.

Step 3: Set a signal s equal to the signal f to be learned and initialize the output network to $\tilde{f} = 0$.

Step 4: Compute the coefficients α_l corresponding to the g_l functions of the library by applying a series of FWT to the signal s using a dual set of wavelet and scaling function filters (\tilde{h}, \tilde{g}) . s can be written :

$$s = \sum_{l=1}^L \alpha_l g_l. \quad (26)$$

Step 5: Compute the contribution of all the activation functions in the library to reconstruct the signal s . This contribution $\alpha_l g_l$ is obtained by a simple product of the functions of the library and their corresponding weights α_l .

Step 6: Select from the library a function. This function g_k (k is the number of the selected function) contributes the most to the reconstruction of the signal s .

Step 7: Optimize the dilation and the position parameters of the g_k function to better approximate the signal s and improve the accuracy of the output network.

Step 8: Compute the corresponding weight α'_k (introduced in section 3.2.3) of the optimized function g'_k of g_k doesn't belong any more to the library.

Step 9: Add the function g'_k to the hidden layer of the network that approximate f and set its corresponding weight to α'_k . The output of the network is $\tilde{f} = \tilde{f} + \alpha'_k g'_k$

Step 10: Compute the residual signal $s = f - \tilde{f}$ and return to the Step 4 if you didn't come to the end of the learning.

3.2.2. Optimization of the function nodes

To increase the efficiency of the network, we optimize the selected function g_k of the step 6 using Levenberg-Marquardt method. Indeed, the parameters of this function are used as initialization of this algorithm. The result is a function g'_k that enhances the performances of the network output and gives best approximation of the residual signal s .

At a given iteration k , to get the activation function g'_k , we optimize the dilation parameter a and the position parameter b of the function g_k by minimizing the function:

$$E_k = \min_{a,b} \left(\frac{1}{2} \left(f - \sum_{i=1}^{k-1} \alpha_i g_i - \alpha_k g_k \right)^2 \right). \quad (27)$$

3.2.3. Computing the corresponding weight of the optimized function

When we select and optimize a function from the library of the wavelet and the scaling functions to use it in the hidden wavelet network layer, we compute its corresponding weight by the following way:

(1) We have

$$s = \alpha_1 g_1 + \dots + \alpha_k g_k + \dots + \alpha_L g_L. \quad (28)$$

(2) We decompose by FWT the signal g'_k

$$g'_k = \gamma_1 g_1 + \dots + \gamma_k g_k + \dots + \gamma_L g_L. \quad (29)$$

(3) Because $\gamma_k \neq 0$ (Proof. Details are provided in Appendix) we can write

$$g_k = \frac{1}{\gamma_k} g'_k - \frac{\gamma_1}{\gamma_k} g_1 - \dots - \frac{\gamma_L}{\gamma_k} g_L. \quad (30)$$

In the equation (28) we replace g_k by the equation (30)

$$\begin{aligned} s &= \alpha_1 g_1 + \dots + \alpha_k \left(\frac{1}{\gamma_k} g'_k - \frac{\gamma_1}{\gamma_k} g_1 - \dots - \frac{\gamma_{k-1}}{\gamma_k} g_{k-1} - \frac{\gamma_{k+1}}{\gamma_k} g_{k+1} \right. \\ &\quad \left. - \dots - \frac{\gamma_L}{\gamma_k} g_L \right) + \dots + \alpha_L g_L \\ &= \left(\alpha_1 - \alpha_k \frac{\gamma_1}{\gamma_k} \right) g_1 + \dots + \frac{\alpha_k}{\gamma_k} g'_k + \dots + \left(\alpha_L - \alpha_k \frac{\gamma_L}{\gamma_k} \right) g_L. \end{aligned} \quad (31)$$

We deduce the weight α'_k of the activation function g'_k

$$\alpha'_k = \frac{\alpha_k}{\gamma_k}. \quad (32)$$

4. Simulation

4.1. Benchmark studies

The performance of the proposed wavelet network trained using Fast Wavelet Transform Learning Algorithm(FWTLA) has been illustrated by comparing with other classifiers reported in the literature using six benchmark data sets. The six benchmarks namely, Wisconsin Breast Cancer, Glass, Ionosphere, Iris flower, Pima Indian Diabetes and Wine data sets were obtained from the UCI machine learning repository.²⁵ These data represent a wide range of domains and data characteristics. The description of the properties of the data such as number of training instances, number of prediction feature values and number of classes for each data set are presented in Table 1. The data sets store pre-specified training data derived from a variety of sources: medical, biological and sociological domains. Some data sets have several or many missing values while others do not. These data also differ from each other by their size and number of predictive feature values. These data sets are as follows:²⁵

Table 1. Characteristics of benchmark data sets used in the study.

Data set	Cases	Attribute	Class	Attribute types	Missing value
W. Br. Cancer	699	9	2	Integer	Yes
Glass	214	9	7	Real	No
Ionosphere	351	34	2	Integer, Real	No
Iris	150	4	3	Real	No
Pima Indian	768	8	2	Integer, Real	No
Wine	178	13	3	Integer, Real	No

4.1.1. Wisconsin Breast Cancer data set

These data represent a set of patients with breast cancer; each patient was characterized by nine numeric attributes plus the diagnosis of the tumor as benign or malignant. The data set has 699 entries with 458 benign and 241 malignant. The nine attributes namely: clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli and mitoses.

4.1.2. Glass data set

This database consists of 214 instances. Nine attributes represent the contents of different chemical elements: refractive index, sodium, magnesium, aluminum, silicon, potassium, calcium, barium and iron were used to predict one from seven possible types of glass: building windows float processed (70 instance), building

16 *O. Jemai et al.*

windows non-float processed (76), vehicle windows float processed (Galvao), vehicle windows non-float processed (none in this database), containers (13), tableware (9) and headlamps (29 instance). From the UCI original format, first attribute (unique Id numbers) was removed in the data conversion process.

4.1.3. *Ionosphere data set*

The data set contains 351 radar data collected by a system in Goose Bay, Labrador. The system consists of a phased array of 16 high-frequency antennas with a total transmitted power on the order of 6.4 kW. The targets are free electrons in the ionosphere. It is a two-class problem where good radar returns are those showing evidence of some type of structure in the ionosphere (225 good) and bad radar returns are those that do not (126 bad). Each data contains 34 input attributes.

4.1.4. *Iris data set*

This is Fisher's famous iris data, which has been extensively studied in the statistics and machine learning literature. The data set contained 150 samples of 3 classes (50 samples in each class) of plants: iris setosa, iris versicolour and iris virginica, which were represented by 1, 2 and 3, respectively. Each samples comprised four input features: sepal length, sepal width, petal length and petal width.

4.1.5. *Pima Indian diabetes data set*

This data catalogs the presence or absence of diabetes among Pima Indian females, 21 years or older, as a function of eight numeric-valued attributes. The eight attributes namely: number of times pregnant, plasma glucose concentration (2 h in an oral glucose tolerance test), diastolic blood pressure, triceps skin fold thickness, 2 h serum insulin, body mass index, diabetes pedigree function and age. The data consist of 768 examples with 500 healthy and 268 diabetes cases. The original source of the data is the national institute of diabetes and digestive and kidney diseases.

4.1.6. *Wine data set*

The data represent results of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The data set contained 178 samples of 3 classes (59, 71 and 48 in class 1, 2 and 3, respectively) of wine. Each sample comprised 13 input features: alcohol, malic-acid, ash, alcalinity of ash, magnesium, total phenols, flavanoids, nonflavanoid phenols, proanthocyanins, colour intensity, hue, OD280/OD315 of diluted wines and proline.

4.2. *Results and discussions*

As mentioned previously, the proposed FWTLA is used to train the wavelet network to classify six benchmark data sets obtained from the UCI repository. The

optimisation of the parameters of our wavelet network (in step 7 of the extended learning algorithm version) are computed according to the Levenberg-Marquardt method.¹⁵ The following simulations will describe the results of the new wavelet network performance employing Beta super-mother wavelets² The performance comparison with other learning algorithms reported in Hoang¹⁷ is studied.

Four different categories of classifier were employed in Hoang,¹⁷ which were decision trees (C4.5, C4.5 rules, ITI, LMDT and OC1), neural networks (LVQ, Nevprop and RBF), rule-based (CN2) and instance-based (Q* and K5). Details for those machine learning classifiers can be found in.¹⁷ The results reported there covered most of the important machine learning classification approaches. Thus, it serves as a good source of comparison with the performance of the proposed FWTLA algorithm which is used to train the wavelet network. Furthermore, information about the maximal reported classification accuracy for the six benchmark data sets is taken from Department of Informatics, Nicolaus Copernicus University⁷ and used to compare with actual measured values using the proposed FWTLA algorithm. One motivation for determining comparative performance across a set of different learning algorithms is to assess whether any particular algorithm demonstrates a significant advantage over the others.

To conduct a fair comparison, the training and test data set in these experiments were equal to that presented by Hoang.¹⁷ According to Hoang,¹⁷ for each performance evaluation, 80% of the training examples were selected at random from the entire data set and the remaining 20% of the data used for testing. Ten runs, each with different training and test sets, were conducted and the results were averaged. The performance is measured by splitting the data into training and a test set, presenting the algorithm with the training set to induce a concept description (weights) and then measuring the percentage of correct predictions made by that concept description (weights) on the test set.

The performance analysis of the wavelet networks is based on the precision. The measure of the ability of the classifier to produce performance results classification is determined by classification rate. Upon completion of the evaluation procedures, the results were gathered and are shown here. Interpretation of the following results will be discussed later.

The performance of the proposed FWTLA trained wavelet network is compared with 11 classifiers listed in Hoang¹⁷ and Eklund and Hoang,^{13,12} as presented in Table 2.

Note that for fair comparison, the experiments conducted in this study followed the procedure used in Hoang¹⁷, i.e., 20% of the data samples were used for test, and the experiment was evaluated over the training and testing sets folded ten times, with the results averaged. The highest value of classification accuracy measured by Hoang in each data set is highlighted in bold-face type. Over the whole six benchmark data sets, one can notice that the proposed FWTLA algorithm trained wavelet network produces better performance than all other methods with average

Table 2. Average of the testing accuracy rates in (%) of 11 different classifiers published in Hoang¹⁷ along with the proposed Wavelet network with FWTLA for the six data sets.

Classifier	W. Br.Cancer	Glass	Ionosphere	Iris	Pima Indian	Wine
C4.5	94.25	70.23	91.56	91.60	71.02	91.09
C4.5 rules	94.68	67.96	91.82	91.58	71.55	91.90
ITI	91.14	67.49	93.65	91.25	73.16	91.09
LMDT	95.74	60.59	86.89	95.45	73.51	95.40
CN2	94.39	70.23	90.98	91.92	72.19	91.09
LVQ	94.82	60.69	88.58	92.55	71.28	68.90
OC1	93.24	57.72	88.29	93.89	50.00	87.31
Nevprop	95.05	44.08	83.80	90.34	68.52	95.41
K5	96.38	69.09	85.91	91.94	71.37	69.49
Q*	95.46	74.78	89.70	92.10	68.50	74.35
RBF	94.89	69.54	87.60	85.64	70.57	67.87
WN with FWTLA	99.85	83.48	98.42	99.66	84.09	99.94

testing accuracy of 99.85% for Wisconsin Breast Cancer, 83.48% for Glass, 98.42% for Ionosphere, 99.66% for Iris, 84.09% for Pima Indian and 99.94% for Wine data set. The maximum accuracy reported by Hoang is measured by K5 (i.e. average accuracy of 96.38%) for Wisconsin Breast Cancer, Q* (i.e. average accuracy of 74.78%) for Glass, ITI (i.e. average accuracy of 93.65%) for Ionosphere, LMDT (i.e. average accuracy of 95.45%) for Iris, LMDT (i.e. average accuracy of 73.51%) for Pima Indian and LMDT (i.e. average accuracy of 95.40%) for Wine data set. Indeed, these results show the effectiveness of the proposed wavelet network with FWTLA as compared with other classifiers published in the literature.

Moreover, information about the maximal reported classification accuracy for the six data sets is collected from ⁷ and used to compare with average value measured using the proposed FWTLA algorithm as shown in Table 3. Note that for each data set, only the first best classifier with its accuracy and reference is written in Table 3. The overall results indicate that the proposed FWTLA gives higher classification accuracies than the maximum accuracies reported on five data sets: Wisconsin Breast Cancer, Glass, Iris, Pima Indian and Wine, while for the Ionosphere, the proposed FWTLA algorithm (i.e. accuracy of 98.42%) is the second best classifier after the classification accuracy reported by the study of Nicolaus Copernicus University using 3-NN + simplex method (i.e. accuracy of 98.7%). These results show also that the proposed wavelet network with FWTLA has superior performance in comparison with other classifiers.

Table 3. Maximum accuracy in (%) of six data sets and used to compare with average values measured using the proposed WN with FWTLA.

Data set	Maximum accuracy reported			WN with FWTLA
	Reference	Method	Accuracy	
W. Br. Cancer	Duch et al. ¹¹	C-MLP2LN	99.0	99.85
Glass	Domeniconi et al. ¹⁰	Adaptive metric NN	75.2	83.48
Ionosphere	Nicolaus Copernicus University ⁷	3-NN + simplex	98.7	98.42
Iris	Duch et al. ¹¹	C-MLP2LN	98.0	99.66
Pima Indian	Statlog, Michie et al. ²³	Logdisk	77.7	84.09
Wine	Aeberhard et al. ¹	QDA	99.4	99.94

5. Conclusion

In this paper we consider the problem of constructing and training problems of wavelet networks based on fast wavelet transform. For wavelet networks, there are two main problems: The first one is how to compute the connection weights. The classical methods have different disadvantages such as time-consuming, convergence to local minima and the inversion of a huge matrix in many cases. This paper proposes a solution based on fast wavelet transform to avoid the mentioned problems. This technique not only has a hundredfold improvement in speed, but also obtains better accuracy of the connection weights. The second problem is the selection of the hidden layer wavelet functions. In fact, many approaches have been proposed to solve it, such as OLS and genetic algorithms. These previously proposed techniques selected the best wavelets from a predefined library to increase the efficiency of the wavelet network, but our selection process technique is simpler, furthermore, it avoids higher efficiency by optimizing the selected wavelets using Levenberg-Marquardt method.

The proposed learning algorithm sufficiently utilizes the time-frequency information contained in the training data, and determines iteratively the number of hidden layer nodes of the network and the associated weights by the fast wavelet transform algorithm. This algorithm largely reduces the number of hidden layer nodes of wavelet networks, and solves the problem of structure optimization of wavelet networks. Finally, we demonstrate that this algorithm is simple and efficient in classification domain.

Acknowledgment

The authors would like to acknowledge the financial support of this work by grants from the General Direction of Scientific Research and Technological Renovation (DGRSRT), Tunisia, under the ARUB program 01/UR/11/02

20 O. Jemai et al.

Appendix A. Proof of $\gamma_k \neq 0$

proof. Assume $\{g_k\}_k$ is orthonormal bases, then f is represented uniquely as

$$f = \alpha_1 g_1 + \dots + \alpha_k g_k + \dots + \alpha_L g_L. \quad (\text{A.1})$$

Since the g'_k is the optimized of g_k , based on (29), it is also obvious that f can be represented as

$$f = \alpha'_1 g_1 + \dots + \alpha'_{k-1} g_{k-1} + \alpha'_k g'_k + \alpha'_{k+1} g_{k+1} + \dots + \alpha'_L g_L. \quad (\text{A.2})$$

since $\alpha_k g_k$ contribute the most in the reconstruction of the signal f , so this means that $\alpha_k \neq 0$

let suppose that $\gamma_k = 0$. Substituting formula (29) into (A.2), it yields

$$\begin{aligned} f = & \alpha'_1 g_1 + \dots + \alpha'_{k-1} g_{k-1} + \alpha'_k (\gamma_1 g_1 + \dots + \gamma_k g_k + \dots + \gamma_L g_L) \\ & + \alpha'_{k+1} g_{k+1} + \dots + \alpha'_L g_L \end{aligned} \quad (\text{A.3})$$

since $\gamma_k g_k = 0$, formula (A.3) is represented as

$$\begin{aligned} f = & (\alpha'_1 + \alpha'_k \gamma_1) g_1 + \dots + (\alpha'_{k-1} + \alpha'_k \gamma_{k-1}) g_{k-1} + (\alpha'_{k+1} + \alpha'_k \gamma_{k+1}) g_{k+1} \\ & + \dots + (\alpha'_L + \alpha'_k \gamma_L) g_L \end{aligned} \quad (\text{A.4})$$

It follows from (A.1) and (A.4) that $\alpha_k = 0$, so this is contradictory to what is mentioned above that $\alpha_k g_k$ contribute the most in the reconstruction of the signal f which implies that $\alpha_k \neq 0$. Hence, the supposition is false which yields to conclude that $\gamma_k \neq 0$

References

1. S. Aeberhard, D. Coomans, O. Vel, "Comparison of classifiers in high dimensional settings", Technical Report Number 92-02, Department of Computer Science and Department of Mathematics and Statistics, James Cook University of North Queensland, 1992.
2. C. Ben Amar, M. Zaied and M. A. Alimi, "Beta wavelets. Synthesis and application to lossy image compression", *Journal of Advances in Engineering Software*, Elsevier Edition, Vol. 36, N7, pages 459 - 474. 2005.
3. Stephen A. Billings, Hua-Liang Wei, "A New Class of Wavelet Networks for Nonlinear System Identification". *IEEE Trans. Neural Networks*, vol.16, pp. 861-874, July 2005
4. C. S. Burrus, R. A. Gopinath, and H. Guo, *Introduction to Wavelets and Wavelet Transforms: A Primer*, Prentice Hall, Upper Saddle River, NJ, USA, 1998.
5. L. ChangGyoon, K. Kangchui, K. Eungkon, "Modeling for an adaptive wavelet network parameter learning using genetic algorithms", *in: Proceedings of the Fifteenth IASTED International Conference on Modeling and Simulation*, CA, USA, 2004, pp. 55-59.
6. L. Chunguang, W. Peng, E. Javier, "Complex-valued wavelet network", *J. Comput. System Sci.*, 67 (3) (2003) 623-632.

7. Datasets Used for Classification: Comparison of Results, Nicolaus Copernicus University, Department of Informatics, 2008, <http://www.fizyka.umk.pl/kmk/projects/datasets.html>.
8. C. Daniel, O. Umit, "Wavelet neural networks: a design perspective", in: *IEEE International Symposium on Intelligent Control-Proceedings*, 1994, pp. 376-381.
9. Daubechie I., "Ten Lectures on Wavelets", *CBMS-NSF Regional Conferences Series in Applied Mathematics*, SIAM (Capital City Press, Montpelier, Vermont), 1992.
10. C. Domeniconi, J. Peng, D. Gunopulos, "An adaptive metric for pattern classification", *Advances in Neural Information Processing Systems*, vol. 13, MIT Press, Cambridge, MA, 2000.
11. W. Duch, R. Adamczak, K. Grabczewski, "A new methodology of extraction, optimization and application of crisp and fuzzy logical rules", *IEEE Transactions on Neural Networks*, 12 (2) (2001) 277-306.
12. P. Eklund, A. Hoang, "A comparative study of public supervised classifier performance on the UCI database", *Australian Journal of Intelligent Information Processing Systems*, 9 (1) (2006) 1-39.
13. P. Eklund, A. Hoang, "A performance survey of public domain machine learning algorithms", Technical Report, School of Information Technology, Griffith University, 2002.
14. R. K. H. Galvao, Takashi Yoneyama, "A Competitive Wavelet Network for Signal Clustering". *IEEE Trans.Systems,Man,And Cybernetics-Part B: Cybernetics*, vol.34, pp.1282-1288, April 2004
15. Gler and all, "Recurrent neural networks employing Lyapunov exponents for EEG signals classification", *Expert systems with applications*, 29: 506-514, 2005.
16. Yongyong He, Fulei Chu, Binglin Zhong, "A Hierarchical Evolutionary Algorithm for Constructing and Training Wavelet Networks". *Neural Comput. Applic.*, vol.10, pp:357-366,2002.
17. A. Hoang, "Supervised classifier performance on the UCI database", Master Thesis, Department of Computer Science, University of Adelaide, Australia, 1997.
18. Changbo Hu, Rogerio Feris and Matthew Turk, "Real-time View-based Face Alignment using Active Wavelet Networks", *Proceedings of the IEEE International Workshop on Analysis and Modeling of Faces and Gestures*, p. 215, 2003.
19. X. Jinhua, W.C. Ho. Daniel, "A constructive algorithm for wavelet neural networks", *First International Conference on Natural Computation*, Changsha, China, 2005, pp. 730-739.
20. Z. Jun, G.G. Walter, Y. Miao, "Wavelet neural networks for function learning", *IEEE Trans. Signal Process.*, 43 (6) (1995) 1485-1496.
21. C. Kai, H. Yoshio, "An adaptive hybrid wavelet neural network and its application", in: *Proceedings of the 2004 IEEE International Conference on Robotics and Biomimetics*, IEEE ROBIO 2004, Shenyang, China, 2004, pp. 73-86.
22. Kruger V., "Gabor wavelet networks for object representation", Ph. D thesis, 2001.
23. D. Michie, D.J. Spiegelhalter, C.C. Taylor, *Machine Learning Neural and Statistical Classification*, Elis Horwood, London, 1994.
24. H. Min, C. Baotong, "A novel learning algorithm for wavelet neural networks", in: *First International Conference on Natural Computation*, ICNC, Changsha, China, 2005, pp. 1-7.
25. D.J. Newman, S. Hettich, C.L. Blake, C.J. Merz, D.W. Aha, UCI Repository of Machine Learning Databases, Department of Information and Computer Science, University of California, Irvine, 1998 of Conference, <http://archive.ics.uci.edu/ml/datasets.html>

22 *O. Jemai et al.*

26. Yacine OUSSAR, Grard DREYFUS , "Initialization by selection for wavelet network training", *Neurocomputing*, 34 (2002) 131-143.
27. Y. C. Pati and P. S. Krishnaprasad, "Analysis and synthesis of feed-forward neural networks using discrete affine wavelet transformations". *IEEE Trans. Neural Networks*, Vol.4, pp.73-85, Jan.1993.
28. Seda Postalcioglu, Yasar Becerikli, "Wavelet networks for nonlinear system modeling". *Neural Comput. Applic.*, vol.16, pp:433-441,2007
29. Z. Qianhua, A. Benvenist, "Wavelet network", *IEEE Trans. Signal Process.*, 13 (6) (1992) 889-898.
30. Z. Qinghua, "Using wavelet network in non parameters estimation", *IEEE Trans. Neural Networks* 8 (1997) 227-236.
31. S. Qingmei, G. Yongchao, "A stepwise updating algorithm for multiresolution wavelet neural networks", in: *Proceedings of the International Conference on Wavelet Analysis and Its Applications (WAA)*, Chongqing, China, 2003, pp. 633-638.
32. R. M. Sanner and J. J. E. Slotine, "Structurally dynamic wavelet network for adaptive control of robotic systems". *International Journal of Control*, vol. 70, pp. 405-421, 1998.
33. Celso de Sousa, Jr., Elder Moreira Hemerly, R. K. H. Galvao, "Adaptive Control for Mobile Robot Using Wavelet Networks". *IEEE Trans.Systems,Man,And Cybernetics-Part B:Cybernetics*, vol.32, pp. 493- 504, August 2002
34. K. Seong-Joo, K. Taek-Yong, S. Jae-yong, J. Hong-Tae, "Design of the scaling-wavelet neural network using genetic algorithm", in: *Proceedings of the International Joint Conference on Neural Networks*, Honolulu, HI, 2002, pp. 2174-2179.
35. K. W. Wong and A. C. S. Leung, "On-line successive synthesis of wavelet networks". *Neural Processing Letters*, vol. 7, pp. 91-100, 1998.
36. J. H. Xu and D. W. C. Ho, "A basis selection algorithm for wavelet neural networks". *Neurocomputing*, vol. 48, pp. 681-689, 2002.
37. B. Yevqeniy, L. Nataliya, P. Iryna, V. Olena, "An adaptive learning algorithm for a wavelet neural network", *Expert Systems* 22 (5) (2005) 235-240.
38. L. Ying, D. Zhidong, Z. Xi, "Approximated non-tensor pre-wavelets neural network based on compound training algorithm", *J. Inform. Comput. Sci.*, 2 (4) (2005) 715-722
39. Q. Zhang and A. Benveniste, "Wavelet network". *IEEE Trans.Neural Network*, Vol.3,pp.889-898, Nov 1992.
40. J. Zhang et al, "Wavelet neural networks for function learning". *IEEE Trans.Signal Process*, Vol.53,pp.1485-1497,1995.
41. Q. Zhang, "Using wavelet network in nonparametric estimation". *IEEE Transactions on Neural Networks*, vol. 8, pp. 227-236, 1997.