Principle of Neural Network and Its Main Types: Review

Abdel-Nasser Sharkawy

Principle of Neural Network and Its Main Types: Review

Abdel-Nasser Sharkawy[*]

Mechatronics Engineering, Mechanical Engineering Department, Faculty of Engineering, South Valley University, Qena, 83523, Egypt

Abstract: In this paper, an overview of the artificial neural networks is presented. Their main and popular types such as the multilayer feedforward neural network (MLFFNN), the recurrent neural network (RNN), and the radial basis function (RBF) are investigated. Furthermore, the main advantages and disadvantages of each type are included as well as the training process.

Keywords: Neural Network; Multilayer Feedforward Neural Network; Recurrent Neural Network; Radial Basis Function; Training; Advantages and Disadvantages.

1. INTRODUCTION

The neural network (NN) [1] is defined as the massively parallel distributed processor which consists of simple processing units that has a natural propensity to store the experiential knowledge and making it available to use.

The NNs are widely used by many researchers in many different applications such as robotics [2-7], speech recognition [8, 9], human face recognition [10, 11], medical applications [12-14], manufacturing [15, 16], and economics [17, 18].

In robotics applications, the MLFFNN proposed in [2-4] was used to detect the occurred collisions during the collaboration between the human operator and the robot. In [5], experiments were presented on the learning of object handling behaviours by a small humanoid robot using the RNN with parametric bias. In [6], a variable admittance control in human-robot cooperation was proposed based on online training of MLFFNN. In this admittance controller, the variable damping was adjusted. In [7], the virtual inertia of the robot admittance controller based also on MLFFNN.

Speech recognition was also proposed based on NN. In [8], convolutional NN-based acoustic model for raw speech signal was discussed. In this case, the relation between raw speech signal and phones in a data-driven manner was established. In [9], the convolutional NN-based approach was proposed to large vocabulary speech recognition task. This approach was compared against the conventional NN-based approach on Wall Street Journal corpus.

Human face recognition was also proposed based on NN. In [10], the MLFFNN was presented for the development of a human face recognition system. In this case, the network was trained with a set of face images until it is in a learned state. The network could classify the face input into its class. In [11], a novel face recognition method was proposed based on multi-features using a neural networks committee machine. The committee consists of several independent neural networks trained by different image blocks of the original images in different feature domains.

NNs are widely used with medical applications and diagnosis [12]. In [13], the capability of the Rank M-Type RBF neural network was presented in medical image processing applications. In [14], a comparative chest diseases diagnosis was realized by using multilayer, probabilistic, learning vector quantization, and generalized regression neural networks. The chest diseases dataset was prepared by using patient's epicrisis reports from a chest diseases hospital's database.

NNs are also used in various areas of manufacturing. In [15], methods for training neural networks was proposed to model complex manufacturing processes. In this work [15], a case study of a complex forming process was used to demonstrate a real implementation case in industry. In [16], the RNN was used for the design and the implementation of a sequential controller for a flexible manufacturing system.

Economics' applications also used the artificial NNs. In [17], an approach for the process of forecasting the financial data was presented using the RBF. In [18], the consumer behavior was identified using the artificial neural networks based on the information obtained from traditional surveys. The results proved that the neural networks have a good discriminatory power,

*Address correspondence to this author at the Mechatronics Engineering, Mechanical Engineering Department, Faculty of Engineering, South Valley University, Qena, 83523, Egypt; E-mail: eng.abdelnassersharkawy@gmail.com

generally providing better results compared with traditional discriminant analysis.

From this discussion, we can conclude that there are different types of NNs. In addition, many applications use the NN.

This paper presents the main popular types of the neural networks such as the multilayer feedforward neural network (MLFFNN), recurrent neural network (RNN), and the radial basis function (RBF). The advantages and disadvantages of each type are included as well as the training process. In fact, the advantages and disadvantages of each NN type are presented in this paper to help researchers to choose/select the suitable NN type in their work.

The rest of this manuscript is as follows. Section 2 presents the meaning of the neuron. Section 3 includes the examples of the activation functions. In section 4, the general properties of the NNs are presented. Section 5 presents the MLFFNN architecture, advent-ages, disadvantages, and training. Section 6 presents the RNN architecture, advantages, disadvantages, and training. In section 7, the RBF is presented. Section 8 presents comparison between other types of the NNs. In final, section 9 summarizes the main important points of this paper.

2. THE NEURON MODELS

The neuron is the information-processing unit [1], which is fundamental to the neural network operation. The block diagram showing the model of the neuron is presented in Fig. (**1**). This figure forms the basis for designing a large neural networks' family.
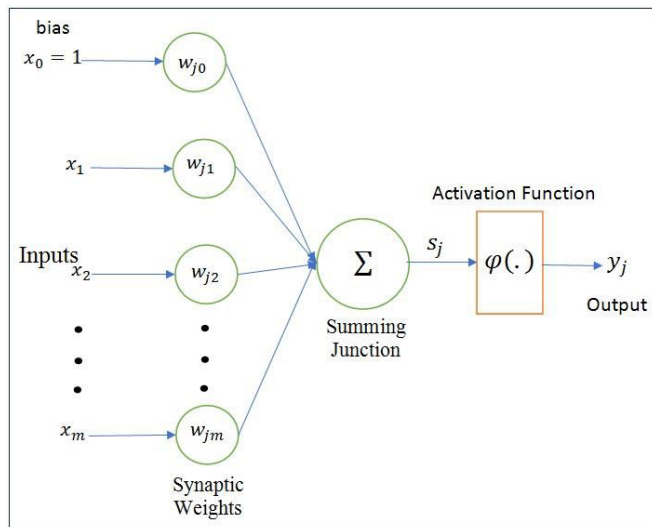

Figure 1: Nonlinear model of a neuron.

In mathematical analysis.

$$s_j = \sum_{m=0}^{m} w_{jm} x_m \tag{1}$$

and

$$y_j = \varphi(s_j) \tag{2}$$

where $x_1, x_2, ..., x_m$ are the input signals; $x_0 = 1$ is the bias. $w_{j0}, w_{j1}, ..., w_{jm}$ are the respective synaptic weights of neuron j. $\varphi(.)$ is the activation function; and y_j is the output signal of the neuron.

3. ACTIVATION FUNCTION

The activation function expressed by $\varphi(s)$ defines the neuron output in terms of the induced local field s. In follows, three types of the activation functions are defined:

- ***The Threshold Function***

In this type of activation function,

$$\varphi(s) = \begin{cases} 1 \ if \ s \geq 0 \\ 0 \ if \ s < 0 \end{cases} \tag{3}$$

- ***The Sigmoid Function***

The example of the sigmoid function is the **"*logistic function*",** and it is defined by

$$\varphi(s) = \frac{1}{1+\exp(-as)} \tag{4}$$

where a is the slope parameter of the sigmoid function.

The ***hyperbolic tangent function*** is also used, and it is defined by

$$\varphi(s) = \tanh(s) \tag{5}$$

- ***The Squash Function***

This type of activation function is defined as

$$\varphi(s) = \frac{s}{1+|s|} \tag{6}$$

4. GENERAL PROPERTIES OF THE NNs

NNs present a lot of advantages which make them widely used in many different applications. These advantages are summarized as follows.

- NN is a powerful tool for the non-linear systems identification [1, 19].

- It can adapt itself by changing the network parameters in a surrounding environment and

can easily handle imprecise, fuzzy, noisy, and probabilistic information [19].

- The NN derives its computing power through its massively parallel distributed structure and its ability to learn and, therefore, to generalize. It was proven that the neural network could approximate any complex (large-scale) linear or non-linear function given appropriate training data [1, 20].

- It also offers useful properties and capabilities such as nonlinearity, input–output mapping and adaptivity, function approximation and generalization [1].

- NN has a kind of universality [21] e.g. approximation of smooth batch data containing the input, output and possibly gradient information of a function [22], and approximating the derivatives of a function [23].

- NN has a vital role in the identification of the dynamic systems and fault detection since it not only can be used to detect the occurrence of the fault, but it also provides a postfault model of the robotic manipulator. This postfault model can be effectively used to isolate and identify the fault and, if possible, for accommodation of the failure [24].

In the following three sections, we concentrate on the three main types of the neural networks: MLFNN, RNN, and RBF.

5. MULTILAYER FEEDFORWARD NN (MLFFNN)

Feedforward NNs are the artificial NNs in which the connections between units do not form a cycle. Feedforward NNs were the first type of artificial NN invented. In addition, they are simpler than their counterpart, the RNNs. They are called Feedforward because information only travels forward in the network (no loops), first through the input nodes, then through the hidden nodes (if present), and in final, through the output nodes. The Feedforward NN divides into two main types as follows.

- ***Single-Layer Feedforward NN (SLFFNN)***

The Single layer is referring to the output layer of the computation nodes (neurons). The input layer of source nodes is not counted because no computation is performed there. The SLFFNN is shown in Fig. (**2**).
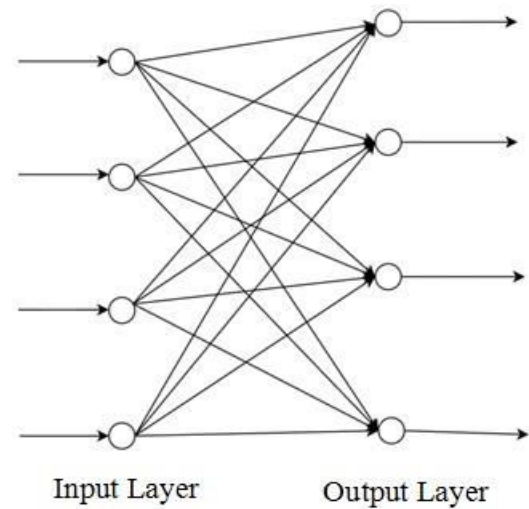

Figure 2: Single layer feedforward neural network.

- ***Multilayer Feedforward NN (MLFFNN)***

The architecture of this NN has one or more hidden layers. Fig. (**3**) shows fully connected feedforward network with one hidden layer and one output layer.
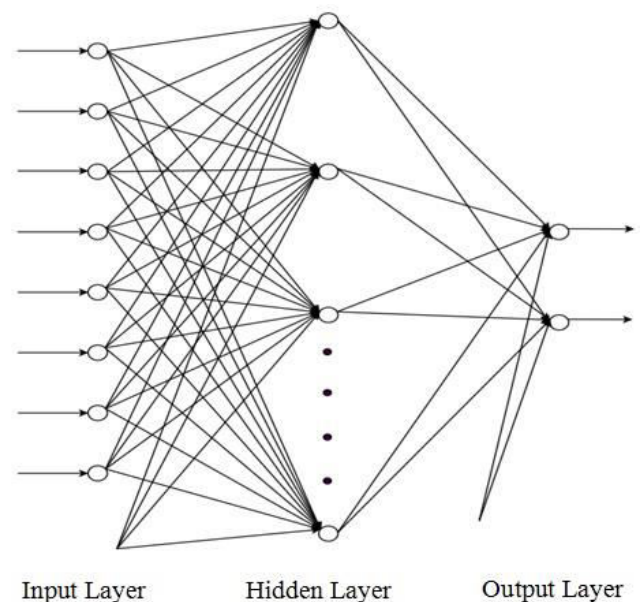
Figure 3: Multilayer feedforward neural network.

Discussion:

Multilayer feedforward NN was used with our research interests in robotics. For example, in [25], the MLFFNN was designed and trained for human-robot collisions detection during the collaboration between the robot and human. In this work, two NN architectures were proposed. The first architecture was composed of one hidden layer. This architecture was designed using the intrinsic joint position and torque

sensors from the manipulator. As known, that most of the robots do not have joint torque sensors and only the collaborative robots have these joint torque sensors. Therefore, this NN architecture can be applied only for the collaborative robots.

The second NN architecture in [25] was composed of two hidden layers. The main benefits from using two hidden layers are the two hidden layers can achieve better result than using a single layer [26], and also

they give a better generalization capability than a single hidden layer in many cases [27]. This architecture was designed using only the intrinsic joint position sensors from the manipulator as it is known that most of the robots have joint position sensors. Therefore, using the two hidden layers in this architecture helped us to implement a method that can be applied to any robot. Both NN architectures are presented in Figs. (**4** and **5**), respectively [25].
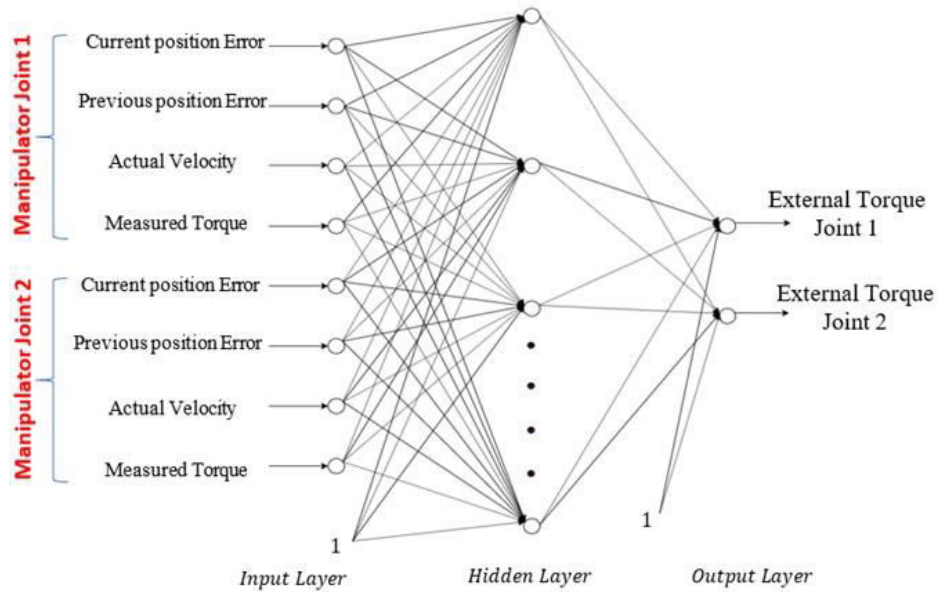

Figure 4: The MLFFNN architecture using the joints' positions and torques sensors of the manipulator. This Picture was taken from ref. [25].
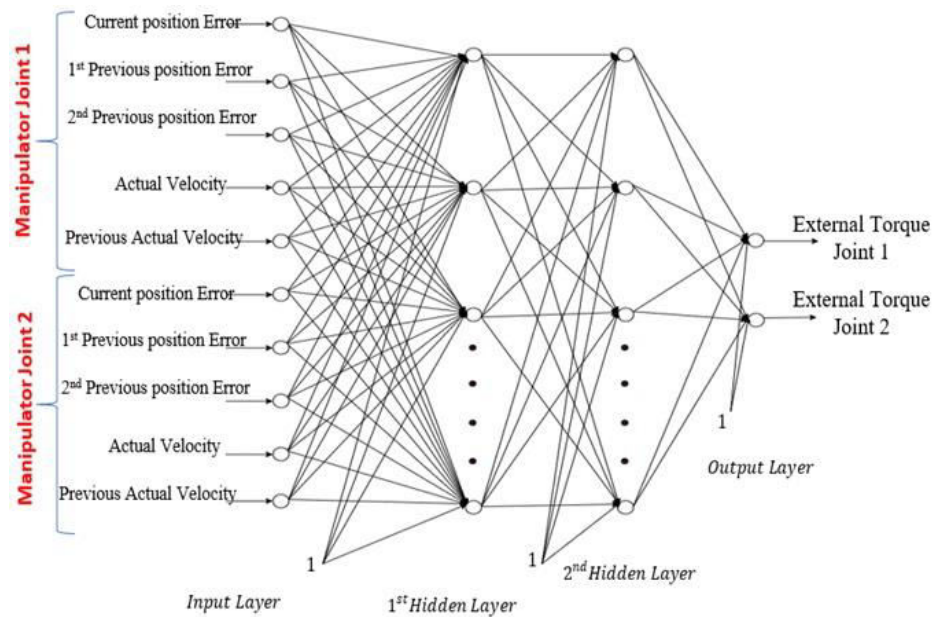

Figure 5: The MLFFNN architecture using only the joints' positions sensors of the manipulator. This Picture was taken from ref. [25].

Difference between MLFFNN and Multilayer Perceptron:

The multilayer perceptron is the special case of the feedforward NN, where every layer is a fully connected layer. In some definitions, the number of nodes in each layer is the same. In addition, in many definitions the activation function across hidden layers is the same. Fig. (**6**) [28] shows what this means.
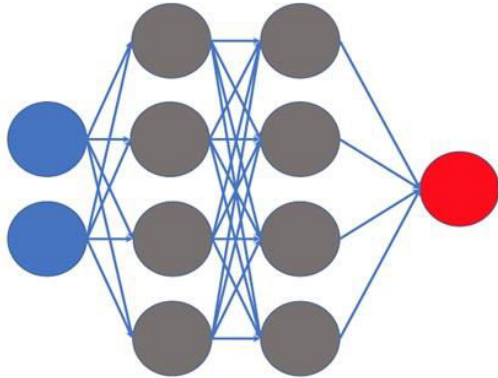

Figure 6: The multilayer perceptron. The picture is taken from ref. [28].

5.1. Advantages and Disadvantages

The main advantage of the MLFFNNs is that they can be used for the difficult to complex problems. However, they need sometimes long training time. In detail, the main advantages and disadvantages of the MLFFNNs are summarized in Table **1**.

5.2. MLFFNN Training

Consider y_j the function signal produced at the output of neuron j in the output layer by the stimulus $\mathbf{x}(n)$ applied to the input layer. Therefore, the error signal produced at the output of neuron j is

$$e_j(n) = d_j(n) - y_j(n) \tag{7}$$

where $d_j(n)$ is the ith element of the desired-response vector $\mathbf{d}(n)$.

The instantaneous error energy of neuron j is known by [1]

$$E_j(n) = \frac{1}{2}e_j^2(n) \tag{8}$$

Summing the error-energy contributions of all the neurons in the output layer, the total instantaneous error energy of the whole network is expressed as

$$E(n) = \sum_{j \in C} E_j(n) = \frac{1}{2}\sum_{j \in C} e_j^2(n) \tag{9}$$

where the set C includes all the neurons in the output layer. With the training sample consisting of N examples, the error energy averaged over the training sample is defined by

$$E_{av}(N) = \frac{1}{N}\sum_{n=1}^{N} E(n) = \frac{1}{2N}\sum_{n=1}^{N}\sum_{j \in C} e_j^2(n) \tag{10}$$

Two different methods are presented for training the MLFFNN: batch learning and online learning.

- ### Batch Learning

In the batch method of supervised learning, the adjustments to the synaptic weights of the MLFFNN are performed after the presentation of all the N examples in the training sample that constitute one epoch of training. In other words, the cost function for batch learning is defined by the average error energy E_{av}. Adjustments to the synaptic weights of the MLFFNN are made on an epoch-by-epoch basis.

- ### Online Learning

In the online method of supervised learning, the adjustments to the synaptic weights of the MLFFNN

Table 1:   The Main Advantages and Disadvantages of the MLFFNN

Type of NN	Advantages	Disadvantages
MLFFNN	▪ MLFFNN is one of the popular ANN that has been used for many ANN applications [29, 30]. ▪ It has a very simple structure compared with other types of artificial neural networks [1, 31]. ▪ MFFNN gives good performance in many cases [32]. ▪ MFFNN is a robust neural network that can be applied easily and successfully in various problem domains [33-35]. ▪ The on-line training of the MLFFNN was used for tuning the control parameters [6, 7, 36-38]. ▪ For function approximation problems, they are preferred as a general model for surfaces without regular peaks and valley [39]. ▪ For classification problems, they can get better classification results with much more efficient networks than RBF networks [32, 39].	This network type requires a large number of pairs of input and target for the training process [32, 40].

Notes: To illustrate the properties presented in Table **1**, in our previous paper [2], the MLFFNN is used for human-robot collisions detection, which is an important application for the safety of the human-robot cooperation. In that paper, the used MLFFNN gave very good performance and very low mean squared error (MSE) (0.040644) after 932 iterations. However, it needed large number of pairs of input and target (56358).

are performed on an example-by-example basis. The cost function to be minimized is therefore, the total instantaneous error energy $E(n)$. The on-line training of the MLFFNN was used for tuning the control parameters as presented in references [6, 7, 36-38].

5.2.1. Learning Algorithms

In this paper, the error back propagation algorithm and the Levenberg-Marquardt algorithm are discussed for training the MLFFN.

Levenberg-Marquardt (LM) algorithm is a type of the second-order optimization techniques that have a strong theoretical basis and provide significantly fast convergence, and it is considered as an approximation to Newton's Method [41, 42]. LM learning is used because it can trade-off between the fast learning speed of the classical Newton's method and the guaranteed convergence of the gradient descent [41, 43]. LM algorithm always suits to large data sets, and converges in less iterations and in shorter time than the other training algorithms. In this algorithm, the optimum weight adjustment $\Delta \boldsymbol{w}$ applied to the parameter vector $\boldsymbol{w}$ is defined by

$$\Delta \boldsymbol{w} = [\mathbf{H} + \lambda \mathbf{I}]^{-1} \boldsymbol{g} \qquad (11)$$

where, $\mathbf{H}$ is Hessian of the second order function and $\boldsymbol{g}$ is the gradient vector of the second order function. $\mathbf{I}$ is the identity matrix of the same dimensions as $\mathbf{H}$ and λ is a regularizing or loading parameter that forces the sum matrix $(\mathbf{H} + \lambda \mathbf{I})$ to be positive definite and safely well-conditioned throughout the computation.

Error backpropagation algorithm has the simplicity of implementation [44], and accelerating convergence to a minimum of the error function if the optimal values of its parameters, learning and momentum rates, are obtained [1, 45]. In this algorithm, the correction weight Δw_{jm} applied to the synaptic weight connecting neuron m to neuron j is defined by the delta rule as:

$$\Delta w_{jm} = \eta \times \delta_j(n) \times x_m(n) \qquad (12)$$

where, η is the learning rate parameter, $\delta_j(n)$ is the local gradient, and $x_m(n)$ is the input signal of the neuron j.

5.3. MLFFNN Generalization

The generalization ability is the most important factor for the NN [1]. This property shows the effectiveness of the trained NN in different conditions and tasks. In [46], the generalization ability to unseen objects and backgrounds has been largely improved, and both the attentional model and the auxiliary classification task were necessary to get this improvement. In [47], the MLFFNN was proposed for the human-robot collision detections and the collided link identification. In that work, the training of the MLFFNN happened using a robot joint motion with limited range of the joins' angles. The effectiveness and the generalization ability of their system was checked outside of the training range joints motion and under different conditions. The method in these cases presented high effectiveness and generalization. This case is presented in Table **2** [47].

As seen from the table that the trained MLFFNN achieve a very good percentage (75%) to detect the collisions between the human and the robot, outside the training range. In fact, this proves the generalization ability and the effectiveness of the trained MLFFNN using different conditions than the ones used for the training process.

In [6], the training of the MLFFNN happened during the robot end-effector movement along straight-line segment. The effectiveness and the generalization ability were investigated using movements along different lengths of the straight-line segments than the one used for the training. The trained MLFFNN presented in these cases highly desired performance. This work was extended and widely presented in [7]. In [7], the generalization ability was presented using three different cases than the ones used for the training process, as follows:

- Different lengths of movements were tested.

- A load was connected to KUKA robot end-effector for emulating real industrial cases since the robot could assist to carry different loads.

- The generalization ability was checked along a different axis (y-axis).

Table 2: Case Presents the Generalization Ability of the Trained MLFFNN. These Results are Taken from Ref. [47]

The Overall Effectiveness of the Trained MLFFNN Inside the Training Range	The Overall Effectiveness of the Trained MLFFNN Outside the Training Range
86.6 %	75.0 %

The generalization ability and the effectiveness of the trained MLFFNN using these three different conditions was high.

From this discussion, we can conclude that the trained NN can be generalised under different cases and conditions.

Note: From our study on the MLFFNN and also using it in our previous work, we can conclude that the MLFFNN can be used for many applications and it is a robust NN. The MLFFNN is easy to be implemented and programmed and it can give excellent results.

6. RECURRENT NN (RNN)

The recurrent NNs [48, 49] are the neural models which are able to take some context into account in their decision function. The RNN distinguishes itself from a feedforward neural network in that it has at least one feedback loop, as shown in Fig. (**7**).
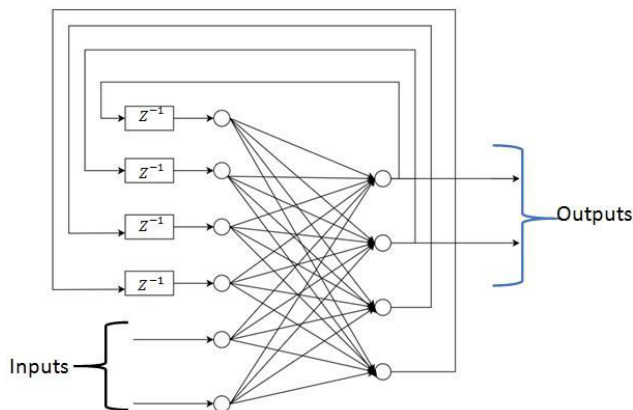

Figure 7: Recurrent network with hidden neurons.

The two main variants proposed in the literature about the RNN, also called "simple" RNNs, are : the Elman [49] and the Jordan [48] RNN models. Both the Elman and Jordan neural networks consist of an input layer, a hidden layer, a delay layer, and an output layer. The delay neurons of an Elman neural network are fed from the hidden layer, whereas the delay neurons of a Jordan neural network are fed from the output layer. The structure of the Elman and the Jordan NN is presented in Fig. (**8**) [50].

Discussion:

In this discussion, we present an example which illustrates the importance of the Elman and Jordan RNN. In [50], a seasonal autoregressive integrated moving average (ARIMA) model and the two recurrent neural networks; the Elman and Jordan NNs, were

established for conducting short-term prediction of human brucellosis cases in mainland China.
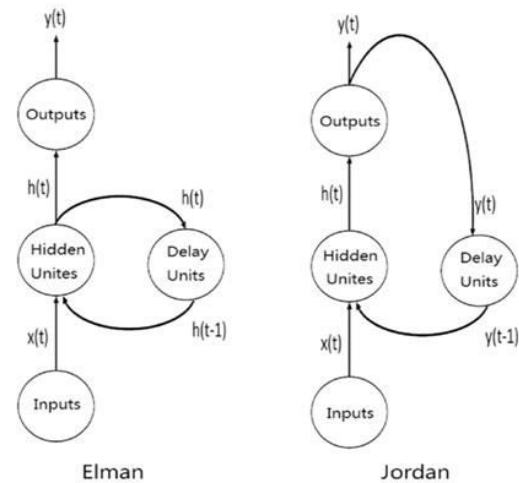

Figure 8: The structure of the Jordan and the Elman NNs. This picture is taken from ref. [50].

In this study, three performance indexes which are the root mean squared error (RMSE), the mean absolute error (MAE) and the mean absolute percentage error (MAPE), were used to assess the fitting and forecasting accuracy of the three models. The MAPE of Elman and Jordan neural networks were almost the same as the MAPE of the seasonal ARIMA model in the training set, whereas the RMSE and the MAE of Elman and Jordan neural networks were lower than those of the ARIMA model. The RMSE and MAE of the Elman neural network were the lowest, whereas the MAPE of the Elman neural network was the highest in the training set. The most likely reason was that the Elman neural network gained better fitting accuracy for large values, but gained poorer fitting accuracy for small values in this study. Importantly, the Elman and Jordan neural networks achieved much higher forecasting accuracy in the test set. The RMSE, MAE, and MAPE of the Elman and Jordan neural networks were far lower than those of the seasonal ARIMA model. Therefore, Elman and Jordan Recurrent Neural Networks are more appropriate than the seasonal ARIMA model for forecasting nonlinear time series data, such as human brucellosis.

6.1. Advantages and Disadvantages

The RNN remembers each and every information through time. It is useful in time series prediction due to the feature to remember previous inputs as well. This is called Long Short-Term Memory. In detail, the advantages and disadvantages of the RNN are presented in Table **3**.

Table 3:   The Main Advantages and Disadvantages of the RNN

Type of NN	Advantages	Disadvantages
RNN	- The RNN is the dynamic neural network [1, 30] and is computationally powerful and can be used in many temporal processing models and applications [51]. - RNNs possess the so-called universal approximation property [52], that is, they are capable of approximating arbitrary nonlinear dynamical systems (under loose regularity conditions) with arbitrary precision, by realizing complex mappings from input sequences to output sequences [53]. - The RNN size is significantly compact compared with feedforward networks for the same approximation accuracy of dynamic systems [51]. - Recurrent networks can be used as associative memories to build attractors from input-output association [51]. - The recurrent neural network is widely used in human-robot interaction (HRI) tasks as presented in [54-57].	- Although, the recurrent network is a simple and powerful model, in practice, it is hard to train properly [58]. - Among the main reasons why this model is so unwieldy are the vanishing gradient and exploding gradient problems described in [59]. - Due to the nonlinear nature of unit activation output characteristics and the weight adjustment strategies, the network stability is often difficult to ascertain [30]. - It cannot process very long sequences if using tanh or relu as an activation function.

6.2. RNN Training

We describe here in this subsection two modes of training a recurrent network as follows.

Epochwise training: For a given epoch, the RNN uses a temporal sequence of input–target response pairs and starts running from some initial state until it reaches a new state, at which point the training is stopped and the network is reset to an initial state for the next epoch. The initial state does not have to be the same for each epoch of training.

Continuous training: This second method of training is suitable for situations where there are no reset states available or on-line learning is required. The distinguishing feature of continuous training is that the network learns while performing signal processing. Simply put, the learning process never stops.

Two different learning algorithms are used with the RNNs as follows.

- **The back-propagation-through-time (BPTT) algorithm** for training the RNN is an extension of the standard back-propagation algorithm. It is derived by unfolding the temporal operation of the network into a layered feedforward network, the topology of which grows by one layer at every time-step.

- **The real-time recurrent learning (RTRL) algorithm:** This algorithm is suitable for online continuous training.

Note: As seen from the discussion of the RNN that it is complex compared with the MLFFNN. The mathematical computation of the RNN is also more complex compared with MLFFNN. In our research area, we suggest the RNN to be applied for detecting the human-robot collision detection and compared with the MLFFNN. In addition, the RNN can be suggested for the variable admittance control in human-robot collaboration. In general, comparing different types of the NNs to the same dataset is important for searching for the best desired performance.

7. RADIAL BASIS FUNCTION (RBF)

The RBF network is the artificial NN which uses radial basis functions as activation functions. The output of the network is a linear combination of radial basis functions of the inputs and neuron parameters. RBF networks have many uses as follows: 1) function approximation, 2) time series prediction, 3) classification, and 4) system control. The RBF, as presented in Fig. (**9**), composes from the following three layers [1]:

- *Input layer:* which consists of m_o source nodes, where m_o is the dimensionality of the input vector x.

- *Hidden layer:* which consists of the same number of computation units as the size of the training sample, N; each unit is mathematically described by a radial basis function.

$$\varphi_j(x) = \varphi\big(\|\mathbf{x} - \mathbf{x}_j\|\big) \qquad (13)$$

where, the jth input data point $\mathbf{x}_j$ defines the center of the radial-basis function, and the vector $\mathbf{x}$ is the signal (pattern) applied to the input layer. Thus, unlike a MLFFNN, the links connecting the source nodes to the hidden units are direct connections with no weights.

- *Output layer:* which consists of a single computational unit. Clearly, there is no restriction on the size of the output layer, except to say that typically the size of the output layer is much smaller than that of the hidden layer.
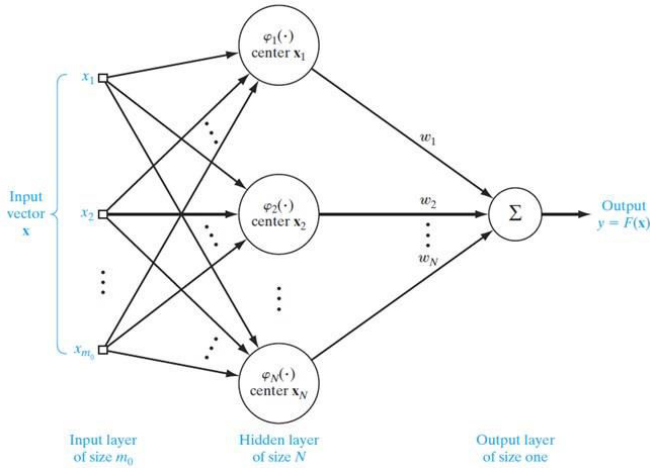

Figure 9: The architecture of the RBF network. This picture is taken from ref. [1].

Discussion:

In this discussion, we present an example/application for the RBF. In [60], diabetes database was used for empirical comparisons between the RBF neural network and the multilayer perceptron (MLP) network. Their results illustrated that the RBF outperforms the MLP model both in the test set and the external set. Their study indicated the good predictive capabilities of RBF neural network. In addition, the time taken by RBF is less than that of the MLP in their application. Although application of RBF network is limited in biomedicine, many comparative studies of MLP and statistical methods are illustrated using a wide range of databases [61-66]. The limitation of the RBF neural network is that

it is more sensitive to dimensionality and has greater difficulties if the number of units is large.

7.1. Advantages and Disadvantages

The RBF networks have the advantages of easy design, good generalization, strong tolerance to input noise, and the online learning ability. In addition, these properties of the RBF networks make them very suitable for designing flexible control systems. The advantages and disadvantages of the RBF are listed in Table **4**.

Note: As seen from the discussion of the RBF that it is an easy design and simple. In our research area, we also suggest the RBF to be applied for implementing a method for safe human-robot collaboration and compared with the MLFFNN and the RNN. In addition, the RBF can be suggested for the variable admittance control in human-robot collaboration. In general, comparing different types of the NNs to the same dataset is important for searching for the best desired performance.

General Comments: Coronavirus disease (COVID-19) is an infectious disease caused by a newly discovered coronavirus. The severity of COVID-19 symptoms can range from very mild to severe. Using artificial NNs is very important and necessary for making predictions or analysis for this virus. This will help for detecting the genetic material of the coronavirus. In addition, in general, we can say that the NNs can be used for fighting against this new virus.

In final, we must admit that there are still some limitations of neural network models, as follows:

1) NN models are black boxes, for example, we cannot know how much each input variable is influencing/affecting the output variables.

2) There are no fix rules to determine the structure as well as the parameters of NN models. It all depends on the researchers' experience.

Table 4: The Main Advantages and Disadvantages of the RBF

Type of NN	Advantages	Disadvantages
RBF	▪ RBF performs faster [32]. ▪ For function approximation problems, RBF networks are specially recommended for surface with regular peaks and valleys [39]. ▪ When dealing with noised input data set, RBF networks perform more robustly and tolerantly than MFFNNs [39].	▪ The complexity of the RBF network is increased with the number of neurons in the hidden layer [67]. ▪ The ordinary RBF has several problems in its structure and training algorithm, so that it is not able to model a strongly nonlinear system [68].

Table 5:   The Comparison between the GRNN, PNN, and CMTNN in Terms of the Advantages

Type of NN	Advantages
GRNN	▪ It can reduce the computation complexity [32, 69] and improve the learning process [69]. ▪ GRNN is similar to the probabilistic neural network (PNN) because both of them use non-parametric estimators of probability density functions. GRNN is suitable to estimate continuous values [70]. ▪ GRNN can work well in noisy environment if it is trained with enough data [71].
PNN	▪ Probabilistic neural network is a type of radial basis networks [72]. ▪ It can integrate the characteristics of statistical pattern recognition and MFFNN [32]. ▪ PNN has the ability to find boundaries between categories of patterns [32, 70]. ▪ It is applied to several areas including pattern recognition, nonlinear mapping and classification [32]. ▪ PNN is a supervised feedforward neural network. It also consists of three layers with one-pass training algorithm [73]. ▪ There are a number of advantages of using PNN for classification. For example, the computational time of PNN is faster than MFFNN, and it is robust to noise. Furthermore, the training manner of PNN is simple and instantaneous [74].
CMTNN	▪ CMTNN has special features based on MFFNN [32] since it can integrate the truth and false membership values to deal with the uncertainty of classification [32, 75]. The other techniques use only truth membership values.

3) It is computationally very expensive and time consuming for training NN models.

4) NN models require processors with parallel processing power for accelerating the training process.

8. OTHER NNs TYPES

In this section, the properties of other types of NN such as general regression neural network (GRNN), probabilistic neural network (PNN), and complementary neural network (CMTNN) are compared. This comparison is investigated in Table **5**.

9. CONCLUSIONS

In this paper, the artificial neural networks are presented. Their main types and the advantages and disadvantages of each type are investigated. From the discussions presented in this paper, the MLFFNN is a robust NN and can be applied for many applications. It is easy to be implemented and programmed and it can give excellent results.

For function approximation problems, the MLFFNN is preferred as a general model for surfaces without regular peaks and valley. For classification problems, the MLFFNN can get better classification results with much more efficient networks than RBF networks. However, this network type requires a large number of pairs of input and target for the training process.

The RNN is the dynamic neural network. Its size is significantly compact compared with MLFFNNs for the same approximation accuracy of dynamic systems. However, the RNN, in practice, is hard to train properly.

RBF performs faster. For function approximation problems, RBF networks are especially recommended for a surface with regular peaks and valleys. However, it is not able to model a strongly nonlinear system. The properties of other types of NNs such as GRNN, PNN, and CMTNN are also presented.

REFERENCES

[1]   Haykin S. *Neural Networks and Learning Machines*, Third Edit. Pearson, 2009.

[2]   Sharkawy A-N, Aspragathos N. Human-Robot Collision Detection Based on Neural Networks. *Int J Mech Eng Robot Res*, 2018;. 7(2): 150-157.

[3]   Sharkawy A-N, Koustoumpardis PN, Aspragathos N. Manipulator Collision Detection and Collided Link Identification based on Neural Networks. in *Advances in Service and Industrial Robotics RAAD 2018 Mechanisms and Machine Science*, A. Nikos, K. Panagiotis, and M. Vassilis, Eds. Springer, Cham, 2018; pp. 3-12.

[4]   Lu S, Chung JH, Velinsky SA. Human-Robot Collision Detection and Identification Based on Wrist and Base Force / Torque Sensors. in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 2005; no. April, 3796-3801.

[5]   Ito M, Noda K, Hoshino Y, Tani J. Dynamic and interactive generation of object handling behaviors by a small humanoid robot using a dynamic neural network model. Neural Networks 2006; 19(3): 323-337,

[6]   Sharkawy A-N, Koustoumpardis PN, Aspragathos N. Variable Admittance Control for Human - Robot Collaboration based on Online Neural Network Training. in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2018):* 2018.

[7]   Sharkawy A-N, Koustoumpardis PN, Aspragathos N. A Neural Network based Approach for Variable Admittance Control in Human- Robot Cooperation : Online Adjustment of the Virtual Inertia. Intell Serv Robot 1-37: 2020.

[8]   Passricha V, Aggarwal RK. Convolutional Neural Networks for Raw Speech Recognition. in *From Natural to Artificial Intelligence - Algorithms and Applications*, 2018; 21-40.

[9]   Palaz D, Magimai-Doss M, Collobert R. Convolutional Neural Networks-Based Continuous Speech Recognition Using Raw Speech Signal. in *2015 IEEE International Conference on*

Acoustics, Speech and Signal Processing (ICASSP), 2015; 4295-4299.

[10] Fadzil MHA, Bakar HA. Human face recognition using neural networks. in *Proceedings of 1st International Conference on Image Processing*, 1994.

[11] Zhao ZQ, Huang DS, Sun BY. Human face recognition based on multi-features using neural networks committee. Pattern Recognit Lett. 2004; 25(12): 1351-1358,

[12] Fukuoka Y. Artificial Neural Networks in Medical Diagnosis. in *Computational Intelligence Processing in Medical Diagnosis Studies in Fuzziness and Soft Computing*, S. M., T. HN., J. A., J. A., J. S., and J. L.C., Eds. 2002; 197-228.

[13] Moreno-Escobar JA, Gallegos-Funes FJ, Ponomaryov VI. Rank M-type radial basis functions network for medical image processing applications. in *Image Processing: Algorithms and Systems V*, 2007; 6497: 1-12.

[14] Er O, Yumusak N, Temurtas F. Chest diseases diagnosis using artificial neural networks. Expert Syst Appl. 2010; 37(12): 7648-7655.

[15] Sukthomya W, Tannock J. The training of neural networks to model manufacturing processes. J Intell Manuf. 2005; 16(1): 39-51.

[16] Abdelhameed MM, Tolbah FA. Design and implementation of a flexible manufacturing control system using neural network. Int J Flex Manuf Syst. 2002; 14(3): 263-279.

[17] Falat L, Pancikova L. Quantitative Modelling in Economics with Advanced Artificial Neural Networks. Procedia Econ Financ. 2015; 34: 194-201.

[18] Badea LM. (Stroie), Predicting Consumer Behavior with Artificial Neural Networks. Procedia Econ Financ. 2014; 15(14): 238-246.

[19] Du K-L, Swamy MNS. *Neural networks in a softcomputing framework*. London: Springer, 2006.

[20] Most T. Approximation of complex nonlinear functions by means of neural networks. in *2nd Weimar Optimization and Stochastic Days 2005*; 2005.

[21] Nielsen MA. *Neural Networks and Deep Learning*. Determination Press, 2015.

[22] Ferrari S, Stengel RF. Smooth Function Approximation Using Neural Networks. IEEE Trans Neural Networks 2005; 16(1): 24-38.

[23] Hornik K, Stinchcombe M, White H. Universal Approximation of an Unknown Mapping and Its Derivatives Using Multilayer Feedforward Networks. Neurul Networks 1990; 3: 551-560.

[24] Vemuri AT, Polycarpou MM. Neural-Network-Based Robust Fault Diagnosis in Robotic Systems. *I*EEE Trans Neural Networks 1997; 8(6): 1410-1420.

[25] Sharkawy AN, Koustoumpardis PN, Aspragathos N. Neural Network Design for Manipulator Collision Detection Based only on the Joint Position Sensors. Robotica 2019; 1-19.

[26] Chester DL. Why Two Hidden Layers are Better than One. in *International Joint Conference on Neural Networks*, 1990; 265-268.

[27] Thomas AJ, Walters SD, Petridis M, Gheytassi SM, Morgan RE. Accelerated optimal topology search for two-hidden-layer feedforward neural networks. in *Engineering Applications of Neural Networks EANN 2016 Communications in Computer and Information Science*, 2016; 629: J. C. and I. L., Eds. Springer, Cham. 253-266.

[28] Feedforward Neural Networks and Multilayer Perceptrons." [Online]. Available: https://boostedml.com/2020/04/feed-forward-neural-networks-and-multilayer-perceptrons.html.

[29] Fung CC, Iyer V, Brown W, Wong KW. Comparing the Performance of Different Neural Networks Architectures for the Prediction of Mineral Prospectivity. in *2005 International Conference on Machine Learning and Cybernetics*, 2005; 394-398.

[30] Chiang YM, Chang LC, Chang FJ. Comparison of static-feedforward and dynamic-feedback neural networks for rainfall-runoff modeling. J Hydrol. 2004; 290(3-4): 297-311.

[31] Schmidhuber J. Deep learning in neural networks : An overview. Neural Networks 2015; 61: 85-117.

[32] Jeatrakul P, Wong KW. Comparing the performance of different neural networks for binary classification problems. in *2009 8th International Symposium on Natural Language Processing, SNLP '09*, 2009; 111-115.

[33] Chen SC, Lin SW, Tseng TY, Lin HC. Optimization of back-propagation network using simulated annealing approach. in *2006 IEEE International Conference on Systems, Man and Cybernetics*, 2006; 2819-2824.

[34] Sassi MA, Otis MJD, Campeau-Lecours A. Active stability observer using artificial neural network for intuitive physical human-robot interaction. Int J Adv Robot Syst. 2017; 14(4): 1-16.

[35] De Momi E, Kranendonk L, Valenti M, Enayati N, Ferrigno G. A Neural Network-Based Approach for Trajectory Planning in Robot-Human Handover Tasks. Front Robot AI. 2016; 3(June, 1-10.

[36] Rad AB, Bui TW, Li V, Wong YK. A New On-Line PID Tuning Method Using Neural Networks. *IFAC Proc Vol IFAC Work Digit Control Past, Present Futur PID Control*, 2000; 33(4): 443-448.

[37] Elbelady SA, Fawaz HE, Aziz A. MA. Online Self Tuning PID Control Using Neural Network for Tracking Control of a Pneumatic Cylinder Using Pulse Width Modulation Piloted Digital Valves. Int J Mech Mechatronics Eng IJMME-IJENS 2016; 16(3) 123-136.

[38] Hernández-Alvarado R, García-Valdovinos LG, Salgado-Jiménez T, Gómez-Espinosa A, Fonseca-Navarro F. Neural Network-Based Self-Tuning PID Control for Underwater Vehicles. Sensors, 2016; 16(9): 1429, 1-18.

[39] Xie T, Yu H, Wilamowski B. Comparison between traditional neural networks and radial basis function networks. in *Proceedings - ISIE 2011: 2011 IEEE International Symposium on Industrial Electronics*, 2011; 1194-1199.

[40] Anderson D, McNeill G. Artificial neural neworks technology: A DACS state-of-the-art report. Utica, New York, 1992.

[41] Du K, Swamy MNS. *Neural Networks and Statistical Learning*. Springer, 2014.

[42] Marquardt DW. An Algorithm for Least-Squares Estimation of Nonlinear Parameters. J Soc Ind Appl Math. 1963; 11(2): 431-441.

[43] Hagan MT, Menhaj MB. Training Feedforward Networks with the Marquardt Algorithm. IEEE Trans Neural Networks 1994; 5(6): 2-6.

[44] Rumelhart DE, Hinton GE, Williams RJ. Learning internal representations by error propagation. in *Parallel Distributed Processing: Exploration of the Microstructure of Cognition*, D. E. Rumelhart and J. L. McClelland, Eds. Cambridge, MA: MIT Press, 1986; 318-362.

[45] Rojas R. *Neural Networks - A Systematic Introduction*. Berlin: Springer-Verlag, 1996.

[46] Singh A, Yang L, Levine S. GPLAC: Generalizing Vision-Based Robotic Skills Using Weakly Labeled Images. in *Proceedings of the IEEE International Conference on Computer Vision*, 2017; 5852-5861.

[47] Sharkawy AN, Koustoumpardis PN, Aspragathos N. Human-robot collisions detection for safe human-robot interaction using one multi-input-output neural network. *Soft Comput*, 2020; 24(9): 6687-6719.

[48] Jordan MI. Serial Order: A Parallel Distributed Processing Approach. San Diego; La Jolla, CA 92093, 1986.

[49] Elman JL. Finding structure in time. Cogn Sci. 1990; 14(2): 179-211.

[50] Wu W, An SY, Guan P, Huang DS, Sen Zhou B. Time series analysis of human brucellosis in mainland China by using Elman and Jordan recurrent neural networks. BMC Infect Dis. 2019; 19(1) 1-11.

[51] Du K-L, Swamy MNS. *Neural Networks and Statistical Learning.* Springer, 2013.

[52] Maass W, Joshi P, Sontag ED. Computational aspects of feedback in neural circuits. PLoS Comput Biol. 2007; 3(1): 0015-0034.

[53] Siegelmann HT, Sontag ED. Turing computability with neural nets. Appl Math Lett. 1991; 4(6): 77-80.

[54] Zhao X, Chumkamon S, Duan S, Rojas J, Pan J. Collaborative Human-Robot Motion Generation using LSTM-RNN Collaborative Human-Robot Motion Generation using LSTM-RNN. in *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*, 2018.

[55] Torkar C, Yahyanejad S, Pichler H, Hofbaur M, Rinner B. RNN-based Human Pose Prediction for Human-Robot Interaction. in *Proceedings of the ARW & OAGM Workshop 2019*, 2019; 76-80.

[56] Yamada T, Murata S, Arie H, Ogata T. Dynamical Integration of Language and Behavior in a Recurrent Neural Network for Human - Robot. Front Neurorobot. 2016; 10: 1-17.

[57] Schydlo P, Rakovic M, Jamone L, Santos-Victor J. Anticipation in Human-Robot Cooperation: A Recurrent Neural Network Approach for Multiple Action Sequences Prediction. in *IEEE International Conference on Robotics and Automation (ICRA)*, 2018; 5909-5914.

[58] Pascanu R, Mikolov T, Bengio Y. On the difficulty of training recurrent neural networks Razvan. in *Proceedings of the 30th International Conference on Machine Learning*, 2013.

[59] Bengio Y, Simard P, Frasconi P. Learning long-term dependencies with gradient descent is difficult. IEEE Trans Neural Networks 1994; 5(2): 157-166.

[60] Venkatesan P, Anitha S. Application of a radial basis function neural network for diagnosis of diabetes mellitus. Curr Sci. 2006; 91(9): 1195-1199.

[61] Baxt WG. Use of an Artificial Neural Network for the Diagnosis of Myocardial Infarction. Ann Intern Med. 1991; 843-848.

[62] Buchman TG, Kubos KL, Seidler AJ, Siegforth MJ. A comparison of statistical and connectionist models for the prediction of chronicity in a surgical intensive care unit. Crit Care Med. 1994; 22(5): 750-762.

[63] Selker HP, Griffith JL, Patil S, Long WJ, D'Agostino RB. A comparison of performance of mathematical predictive methods for medical diagnosis: identifying acute cardiac ischemia among emergency department patients. J Investig Med. 1995; 43(5):468-476.

[64] Lang E, Pitts L, Damron S, Rutledge R. Outcome after severe head injury: An analysis of prediction based upon comparison of neural network versus logistic regression analysis. Neurol Res. 1997; 19(3): 274-280.

[65] Lapuerta P, Rajan S, Bonacini M. Neural Networks As Predictors of Outcomes in Alcoholic Patients With Severe Liver Disease. Hepatology 1997; 25(2): 302-306.

[66] Lippmann RP, Shahian DM. Coronary artery bypass risk prediction using neural networks. Ann Thorac Surg. 1997; 63(6): 1635-1643.

[67] Kurban T, Beşdok E. A comparison of RBF neural network training algorithms for inertial sensor based terrain classification. Sensors, 2009; 9(8): 6312-6329.

[68] Wang X, Ding Y, Shao H. The improved radial basis function neural network and its application. Artif Life Robot. 1998; 2(1): 8-11.

[69] Song Y, Ren Y. A Predictive Model of Nonlinear System Based on Generalized Regression Neural Network. in *2005 International Conference on Neural Networks and Brain*, 2005; 2009-2012.

[70] Specht DF. A General Regression Neural Network. *IEEE Trans Neural Networks*, 1991; 2(6): 568-576.

[71] Frost F, Karri V. Performance comparison of BP and GRNN models of the neural network paradigm using a practical industrial application. in *ICONIP'99 ANZIIS'99 & ANNES'99 & ACNN'99 6th International Conference on Neural Information Processing Proceedings (Cat No99EX378)*, 1999; 1069-1074.

[72] Ooi SY, Teoh ABJ, Ong TS. Compatibility of biometric Strengthening with probabilistic neural network. in *IEEE-International Symposium on Biometrics and Security Technologies, ISBAST'08*, 2008; 1-6.

[73] Gorunescu F, Gorunescu M, El-darzi E. An Evolutionary Computational Approach to Probabilistic Neural Network with Application to Hepatic Cancer Diagnosis. in *18th IEEE Symposium on Computer-Based Medical Systems*, 2005; 461-466.

[74] Wu SG, Bao FS, Xu EY, Wang Y-X, Chang Y-F, Xiang Q-L. A Leaf Recognition Algorithm for Plant Classification Using Probabilistic Neural Network. in *2007 IEEE International Symposium on Signal Processing and Information Technolog*, 2007; 11-16.

[75] Kraipeerapun P. Neural network classification based on quantification of uncertainty. Murdoch University, 2009.