FUZZY NEURAL TREE FOR KNOWLEDGE MODELING: A PROBABILISTIC POSSIBILISTIC FRAMEWORK

ÖZER CIFTCIOGLU*
Department of Architecture, Faculty of Architecture, Maltepe University, Marmara Education Village 34857 Maltepe - Istanbul, Turkey
E-mail: ozerciftcioglu@maltepe.edu.tr

MICHAEL S. BITTERMANN
Department of Architecture, Faculty of Architecture, Maltepe University, Marmara Education Village 34857 Maltepe - Istanbul, Turkey
E-mail: michaelsbittermann@maltepe.edu.tr

A novel fuzzy-neural tree (FNT) is presented, where each tree node uses a Gaussian as a fuzzy membership or possibility distribution in place of sigmoidal function in conventional neural networks. Although neural networks with Gaussian activation functions as well as different types of cooperative neuro-fuzzy systems have been extensively described in the literature, the FNT presented in this paper implies a novel type of cooperation between fuzzy logic and neural structure. The neurons of the neural tree perform fuzzy logic operations, and in contrast to existing neural systems, the parameters of the operations are established not by training from measured data, but by conditioning the neuron with the consistency condition of possibility theory being entirely independent of data. Therefore the FNT uniquely performs transparent knowledge-driven modeling of systems with arbitrary complexity, which is in contrast to data-driven modeling in the existing neural network and neuro-fuzzy systems cases, as well as the transparent knowledge-driven modeling of systems with restricted complexity in the case of the existing fuzzy logic applications. The novel FNT is described in detail pointing out its various potential utilizations, and exemplifying them by means of three computer experiments.

Keywords: Fuzzy logic, genetic algorithms, knowledge modeling, neural tree, Pareto optimality

1. Introduction

Neural networks and neuro-fuzzy systems received much attention in literature for several decades, and their common features are well identified. In this respect some examples are the identification of the fuzzy sets by neural network training, e.g. 1, function approximation by fuzzy logic and neural networks, e.g. 2, equivalence of fuzzy systems to neural networks, e.g. 3-5, fuzzy identification, e.g. 6, and so on. Different network schemes and learning algorithms have been proposed, such as adaptive conjugate learning 7, fuzzy-wavelet radial basis function neural network 8, dynamical recurrent scheme 9, meta-cognitive system 10, self-generating fuzzy neural networks 11, and ensembles of neural networks 12. Different neuro-fuzzy systems may be based on different types of cooperation among fuzzy and neural computation 13. However, in all of the existing works, the role of neural or neuro-fuzzy system is to form a model generalizing the relations among multiple input-output data vector pairs that are available from previous sensor measurement. Therefore their utilization has been mainly in the engineering domain for diverse applications, such as face recognition 14, action recognition 15, non-linear systems control 16,17, traffic incident detection 18, freeway work zone capacity estimation 19, dynamic system identification 20, structural system identification 21, control of building structures 22. Their utilization for soft information processing has relatively remained to be marginal. That is, when measurement data is not available, but when the modeling concerns soft information as contained in the knowledge an expert possesses, the existing neuro-fuzzy systems are not applicable, and fuzzy logic without

* Corresponding Author.
neural network is conventionally employed. Such situations occur for instance in design of buildings, design of products, or design of cities. In such cases a virtual object is sought that should possess certain abstract qualities, while the representation of the qualities is a matter of using existing expert knowledge and not a matter of generalizing from measured data. However, fuzzy logic is restricted to modeling systems with low complexity. As expert knowledge generally contains significant complexity, neither the existing neural and neuro-fuzzy systems nor fuzzy logic apply for the modeling. Emphasizing the dilemma as to knowledge modeling, one notes that fuzzy logic applications are praised as transparent, knowledge-driven solutions, yet they do not handle complexity. On the other hand neural network solutions are praised for their learning capability, although they operate with a black-box, data-driven strategy. This dilemma is uniquely resolved by the novel neuro-fuzzy system presented in this paper, where fuzzy logic and neural networks are made use of with their respective strong points in one application at the same time.

Computationally the novelty of the neural tree presented in this paper is, that the fuzzy membership function is both, considered to be and shown to be a likelihood function. The existing fuzzy neural networks have no interpretation as fuzzy logic system, whereas the present fuzzy neural tree uniquely has this interpretation. Fuzzy neural networks are a matter of convenient utilization of neural network computation, where fuzzy logic takes place as an antecedent in that computation. The consequent part is neural network, which is not subject to fuzzy logic interpretation. The neural system presented in this work has a particular structure known as neural tree. A neural tree is quite similar to a feed-forward neural network in the sense that it has a feed-forward structure with nodes and weights and a single or multiple outputs. However, it is built not layer by layer but node by node so that it has more free dimensions compared to a strictly defined feed forward type neural network. The present work uses neural tree in an entirely novel way. Yet, before providing further information about this novelty, the existing neural trees are introduced.

Neural trees have been applied to problems from diverse areas. These include speech recognition and segmentation 23,24; word and vowel recognition 25,26,27; speaker recognition 28,29; face recognition 30; character recognition 31; image analysis 32; power systems control and power signal pattern classification 33,34,35,36; novelty detection 37; time-series prediction 38,39,40,41,42; intrusion detection 43,44; disease classification 43,45; modulation identification 46; traffic prediction 47; watermarking 48; and protein structure prediction 49.

In the existing neural tree works, tree structure, weights and activation functions at the neurons were selected with the single aim to minimize the model error, without need for interpretability as to the model constituents. That is, the difference between given output values belonging to the input patterns and the values provided by the neural tree outputs, is minimized. However, in such data-driven modeling utilizations black-box is generally sufficient. The structure, weights and functions are optimized with combinations of different methods. In the 1990s splitting and pruning heuristics were predominantly used to optimize the structure, in combination with different learning heuristics for the parameter identification, e.g. see 50,51,33,52,34,48,53,54,55. In the last decade different variants of stochastic search methods became popular for development of the tree structure, and the parameter identification at the same time, e.g. see 53,30,44,47,40,42,45. Several different types of neural trees have been distinguished in the literature based on the kind of optimization strategy employed, such as flexible neural tree 56,57, and balanced neural tree 58. It is emphasized that the existing neural tree works are all black-box type of models. That is, the model constituents do not have an intelligible interpretation, as their role is restricted to being a mathematical object subject to model error minimization.

In contrast to the previous works, this research explores new potentials of neural tree systems for real-life soft computing solutions in various disciplines and multidisciplinary areas, where transparency of a model is demanded. This is the case when the problem domain is complex, so that expert knowledge is soft, and involves multi-faceted linguistic concepts. Examples of such areas are design science disciplines, such as architectural design, industrial design, and urban design. For this exploration, the coordination of the fuzzy logic and neural network concepts in a compact neuro-fuzzy modelling framework is endeavored, introducing some novel peculiarities for solid interesting gains in interdisciplinary implementations. A novel type of neural tree is introduced. It is emphasized that the novel
neural tree has uniquely an interpretation as a fuzzy logic system. Further, next to representing a complex, non-linear relation between input and output vectors, it satisfies the consistency condition of possibility. Due to this additional property, the neural tree emulates a human-like reasoning, and permits the direct integration of existing expert knowledge during the model formation. The new framework is introduced as fuzzy neural tree with Gaussian type fuzzy membership functions being reminiscent of functioning in RBF type networks. The working of fuzzy logic with a neural tree is already an interesting coupling, arousing interesting anticipations in prospective applications.

The organization of the paper is as follows. Section Two describes the fuzzy neural tree concept starting with the neural tree and joining fuzzy logic to it in the same way as in the development of conventional neuro-fuzzy systems. Section Three describes the probabilistic/possibilistic base underlying fuzzy-neural tree, in particular addressing the tree node with logical AND operation. Section Four describes the probabilistic/possibilistic base underlying fuzzy-neural tree, in particular addressing the tree node with logical OR operation. Section Five gives some diverse hypothetical examples about the working of fuzzy-neural tree when it is in actual use. Section Six gives the conclusions.

2. Fuzzy Neural Tree for Knowledge Modeling

Broadly, a neural tree can be considered as a feed-forward neural network organized not layer by layer but node by node. The nonlinear functions at the nodes can be sigmoid as in perceptron networks. In fuzzy neural networks, this nonlinear function is treated as a fuzzy logic element like membership function or possibility distribution. Therefore, fuzzy logic is integrated into a neural tree with the fuzzy information processing executed in the nodes of the tree. A generic description of a neural tree subject to analysis in this research is as follows.

Neural tree networks are in the paradigm of neural networks with marked similarities in their structures. A neural tree is composed of terminal nodes, non-terminal nodes, and weights of connection links between the pairs of nodes. The non-terminal nodes represent neural units and the neuron type is an element introducing a non-linearity simulating a neuronal activity. In the present case, this element is a Gaussian function which has several desirable features for the goals of the present study; namely, it is a radial basis function ensuring a solution and the smoothness. At the same time it plays the role of possibility distribution in the tree structure which is considered to be a fuzzy logic system as its outcome is based on fuzzy logic operations thereby providing associated reasoning. In a conventional neural network structure there is a hierarchical layer structure where each node at the lower level is connected to all nodes of the upper layer nodes. However, this is very restrictive to represent a general system. Therefore, a more relaxed network model is necessary and this is accomplished by a neural-tree the properties of which are as defined above. An instance of a neural tree is shown in figure 1.

![Fig. 1. Structure of a neural tree](image)

Each terminal node, also called leaf, is labelled with an element from the terminal set \( T = \{x_1, x_2, \ldots, x_n\} \) where \( x_i \) is the \( i \)-th component of the external input \( x \) which is a vector. Each link \((i,j)\) represents a directed connection from node \( i \) to node \( j \). A value \( w_{ij} \) is associated with each link. In a neural tree, the root node is an output unit and the terminal nodes are input units. A non-terminal node should have minimally multiple inputs. It may have single or multiple outputs. A node having a single input is a trivial case and is not defined as a node. This is because in this case output of the node practically is approximately equal to the input while it is considered to be exactly equal. The node outputs are computed in the same way as computed in a feed-forward neural network. In this way, neural trees can represent a broad class of feed-forward networks that have irregular connectivity and non-strictly layered structures. In conventional neural tree structures generally connectivity between the branches is avoided, they are used for pattern recognition, progressive decision making, or complex system modeling. In contrast with these works, in the present research a neural tree structure is developed in a fuzzy logic framework for knowledge modelling where fuzzy
probability/possibility as an element of soft computing is central. In this work neural tree functionality is based on likelihood representing fuzzy probability/possibility. This is a significant difference between the existing neural trees in literature and the one in this work. Although in literature a family of likelihood functions is used to define a possibility as the upper envelope of this family \(65, 66\), to the authors’ best knowledge there is no likelihood function approach in the context of neural tree.

In the neural tree considered in this work, the output of \(i\)-th terminal node is denoted \(x_i\) and it is introduced to a non-terminal node. The detailed view of node connection from terminal node \(i\) to internal node \(j\) is shown in figure 2a and from an internal node \(i\) to another internal node \(j\) is shown in figure 2b. The connection weight between the nodes is shown as \(w_{ij}\). In the neural network terminology, a node is a neuron and \(w_{ij}\) is the synaptic strength between the neurons.

![Fig. 2.](image)

**Fig. 2.** The detailed structure of a neural tree with respect to different type of node connections

### 3. Probabilistic-Possibilistic Base Underlying Fuzzy-Neural Tree: Tree Node With Logical AND Operation

The premise of the motivation of this work is to implement soft computing methodology for complex system analysis and design. For this purpose a novel fuzzy neural-tree concept is developed. The fuzzy neural tree is especially for knowledge modelling making use of fuzzy logic for transparency. An input is a measure of degree of fulfillment with respect to the neural tree model. This can be seen by the following similitude between with a multi-input single output (MISO) engineering system and a multi-input neural tree. In MISO, by certain fixed inputs an output is obtained. This is illustrated in figure 3.

![Fig. 3.](image)

**Fig. 3.** A MISO engineering system as a counterpart of a neural tree system in the sense of having multi inputs and a single output

In a neural tree for each terminal input we define a convex fuzzy membership function, whose associated membership function provides a probabilistic/possibilistic value for that input. This is illustrated in figure 4 for triangular fuzzy membership functions.

![Fig. 4.](image)

**Fig. 4.** Inputs to a fuzzy MISO neural tree system as fuzzy sets. Fuzzy membership function provides a membership value which is interpreted as fuzzy probabilistic/possibilistic value.

Referring to figure 2, let us consider two consecutive nodes as shown in figure 5.

![Fig. 5.](image)

**Fig. 5.** Two neurons connected via weight \(w_{ij}\)

In the neural tree, any fuzzy probabilistic/possibilistic input delivers an output at any non-terminal node. We can consider this probabilistic/possibilistic input value as a random variable \(x_i\) which can be modelled as a Gaussian probability density around a mean \(x_{mi}\) that is not known. The probability density is given by

\[
 f_{x_i}(x_i) = \frac{1}{\sqrt{2\pi \sigma}} e^{-\frac{1}{2\sigma^2}(x_i-x_{mi})^2}
\]  

(1)

where \(x_{mi}\) is the mean; \(\sigma\) is the width of the Gaussian.
The likelihood function of the mean value $x_{mi}$ is given by

$$L_i(\theta) = e^{-\frac{1}{2\sigma^2}(x_i-\theta)^2}$$  \hspace{1cm} (2)$$

where $\theta$ is the unknown mean value $x_{mi}$. Likelihood function is considered to be as a fuzzy membership function or fuzzy probability, converting the probabilistic uncertainty to fuzzy logic terms. Thus, $L(\theta)$ plays the role of fuzzy membership function and the node output

$$y_i = L_i(\theta)$$  \hspace{1cm} (3)$$

Referring to figure 5, we consider the input $x_j$ of node $j$ as a random variable given by

$$x_j = y_i w_j$$  \hspace{1cm} (4)$$

where $w_{ij}$ is the synaptic connection weight between the node $i$ and node $j$. In the same way as described above, the pdf of $x_j$ is given by

$$f_{x_j}(x_j) = \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-1}{2\sigma^2}(x_j-x_{mj})^2}$$  \hspace{1cm} (5)$$

and the likelihood function of the mean value $x_{mj}$ with respect to the input $x_j$ is given by

$$L_j(\theta) = e^{-\frac{1}{2\sigma^2}(x_j-\theta_j)^2} = e^{-\frac{1}{2\sigma^2}(w_j y_j-\theta_j)^2}$$  \hspace{1cm} (6)$$

and using (3) in (6), we obtain

$$L_j(\theta) = e^{-\frac{1}{2\sigma^2}(x_j-\theta_j)^2} = e^{-\frac{1}{2\sigma^2}(w_j L_i(\theta)-\theta_j)^2}$$  \hspace{1cm} (7)$$

We consider the neural tree node status where the likelihood is maximum, namely $L(\theta)=1$. In (7) using $L_i(\theta)=1$ we obtain

$$\theta_j = w_j$$  \hspace{1cm} (8)$$

for $L_j(\theta)=1$ where $\theta_j=x_{mj}$ is the mean value of $x_j$. Hence, from (6), we obtain

$$L_j(\theta) = e^{-\frac{1}{2\sigma^2}(x_j-\theta_j)^2}$$  \hspace{1cm} (9)$$

where $w_{ij}$ and $y_j$ are seen in figure 5. Referring to (3), we can write

$$L_j(\theta) = e^{-\frac{1}{2\sigma^2}(x_j-\theta_j)^2}$$  \hspace{1cm} (10)$$

In (10) it is seen that if $L_i(\theta)=1$ then $L_j(\theta)$ is also 1. The explicit input node and inner node connections to the upper nodes are shown in figure 6 where the node outputs are denoted by $O$ as a generic symbol. Referring to (3) and figure 6 the likelihood function in (9) becomes

$$L_j(\theta) = O_j = e^{-\frac{1}{2\sigma^2} w_j (O_j-\theta_j)^2}$$  \hspace{1cm} (11)$$

Fig. 6. Explicit input and inner node connections to the upper nodes

For a leaf node, i.e., an input node to the tree, we define a fuzzy membership function which serves as a fuzzy likelihood function indicating the likelihood of that input relative to its ideal value, which is equal to unity. This input is shown as $x_i$ in figure 6. The important implications of (11) with respect to the axiom of the fuzzy probability/possibility theory are as follows. The likelihood function in its normalized form is a probability which is considered to be as a fuzzy probability, that is, a membership function of a fuzzy set. However the likelihood function can also be considered as a possibility distribution so that the fuzzy membership function represents also a possibility function. These are due to the axioms of probability/possibility measure $\pi(A_i)$ for a fuzzy event $A_i$ given below

$$\forall A_i, i \in I, \pi(\bigcup_{j=1}^{I} A_j) = \max_{i\in I} (\pi(A_j)) \text{ Possibility}$$

$$\forall A_i, i \in I, \pi(\bigcap_{j=1}^{I} A_j) = \min_{i\in I} (\pi(A_j)) \text{ Probability}$$

where $A_i$ is a fuzzy set and $\pi(A_i)$ is the associated probability/possibility distribution.

We can summarize the observations from (11) as follows.
Node outputs always represent a likelihood function which represents a fuzzy probability/possibility function. 

$L_f(\theta)=1$ corresponds to a fuzzy probability/possibility equal to unity and it propagates in the same way so that the following fuzzy likelihood $L_f(\theta)$ component for the corresponding input $O_i$ is also unity, as seen in (11).

In the same way, if all the probabilistic/possibilistic inputs to the neural tree are unity, then all the node outputs of the neural tree nodes are also unity, providing a probabilistic/possibilistic integrity where the maximum likelihood prevails throughout the tree.

Any deviation from the maximum likelihood, that is, deviation from the unity at a leaf node causes associated deviations from the maximum likelihood throughout the tree. Explicitly, any probabilistic/possibilistic deviation from unity at the neural tree input will propagate throughout the tree via the connected node outputs as estimated likelihood representing a probabilistic/possibilistic outcome in the model.

Each inner node in the tree represents a fuzzy probabilistic/possibilistic rule. In a fuzzy modeling the shape and the position of a fuzzy set are essential questions. In the present neural tree approach all the locations are normalized to unity and the shape of the membership function is naturally formed as Gaussian based on the probabilistic considerations.

Each input to a node is assumed to be independent of the others so that the fuzzy memberships of the inputs form a joint multidimensional fuzzy membership. The dependence among the inputs is theoretically possible but actually it is out of concern because each leaf node has its own stimuli and they are not common to the others, in general.

$O_i$ propagates to the following node output $O_j$ in a way determined by the likelihood function. If there is more than one input to a node, assuming that the inputs are independent, the output is given according the relation

$$L(\theta) = L_1(\theta)L_2(\theta)$$

(12)

For a multiple input case of two node inputs, (9) becomes

$$L_j(\theta) = L_1(\theta)L_2(\theta)$$

$$= e^{-\frac{w_1^2}{\sigma^2}(O_1-1)^2}e^{-\frac{w_2^2}{\sigma^2}(O_2-1)^2}$$

(13)

For a case of $n$ multiple input

$$L_j(\theta) = O_j = f(O_j) = \exp[-\frac{1}{2\sigma^2} \sum w_{ji}^2(O_i-1)^2]$$

Logical AND

where $n$ is the number of inputs to the node and $\sigma$ is the common width of the Gaussians. As it is seen from (11), the previous node output $O_i$ plays important role in the following node output and this role is weighted by the connection weight $w_{ji}$. This weight should represent the relation of the node $O_i$ to the node $O_j$. If these nodes are totally related then $w_{ji}$ is unity. Conversely, if there is no relation of the node $O_i$ to the node $O_j$ related then $w_{ji}$ is zero.

In a neural tree node the common width of the Gaussians is designated as $\sigma$ given by $\sigma=\sigma_{w_j}$ as at the $j$-th node. Therefore it includes the knowledge about $w_{ji}$. It is interesting to note that in fuzzy logic terms, the likelihood function (14) can be seen as a two-dimensional fuzzy membership function with respect to the weighted node outputs $x_1$ and $x_2$. In this case the neural tree node output can be seen as a fuzzy rule which can be stated as

$$IF [O_i = X_1 AND O_{i2} = X_2] THEN [O_j \text{ given by (14)}]$$

This is illustrated in figure 7, where the fuzzy set in the consequent space is a singleton. In the antecedent space $O_{i1}$ and $O_{i2}$ are the firing strengths and $f(O_{i1}, O_{i2})$ is the two-dimensional fuzzy set, which is also likelihood.

$$O_1$$

Fig. 7. Fuzzy interpretation of the information processing at the neural-tree node

The implication of this result is used in two ways:

(i) For the nodes of the penultimate level of the tree the likelihoods are considered to be defuzzified outcomes in the consequent space and a multidimensional Pareto optimality among the likelihoods are sought. Such a result is further used
as outcomes of multidimensional fuzzy membership functions and they are subjected to defuzzification for the root node value as the tree-model output.

(ii) For the inner nodes beyond the penultimate layer nodes the likelihood is considered as the outcome of a two-dimensional fuzzy set and a defuzzification is carried out by the connection weights before the information is conveyed to a following node. Explicitly, the multiplication of the output by a weight is defuzzification. Therefore the sum of the input weights associated with a node is unity.

In the neural tree, an input to a node is a probabilistic/possibilistic quantity and (15) applies. Considering the logical AND operation, the node output is determined by (11) as a product operation for the sake of computational accuracy, and the corresponding rule is given by (15). However, in these computations the Gaussian width $\sigma_j$ in (11) is assumed to be known, although it is not determined yet. To determine $\sigma_j$ we impose the fuzzy probability measure for the cases all the inputs to a node are equal as probabilistic/possibilistic condition. In the same way, for the logical OR operation, to determine $\sigma_j$ we impose the fuzzy possibility measure for the cases all the inputs to a node are equal as probabilistic/possibilistic condition. By these impositions there is no sacrifice of accuracy involved. We can determine the Gaussian width $\sigma_j$ by learning the input and output association given in Table I and Table II for 6 inputs, as an example.

<table>
<thead>
<tr>
<th>Table 1</th>
<th>Inputs to a Node (6 Node Inputs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>.1 . . 1 .1 .1 .1 .1 .</td>
<td>.2 .2 .2 .2 .2 .2 .</td>
</tr>
<tr>
<td>.3 .3 .3 .3 .3 .3 .</td>
<td>.4 .4 .4 .4 .4 .4 .</td>
</tr>
<tr>
<td>.5 .5 .5 .5 .5 .5 .</td>
<td>.6 .6 .6 .6 .6 .6 .</td>
</tr>
<tr>
<td>.7 .7 .7 .7 .7 .7 .</td>
<td>.8 .8 .8 .8 .8 .8 .</td>
</tr>
<tr>
<td>.9 .9 .9 .9 .9 .9 .</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table 2</th>
<th>Outputs of a Node</th>
</tr>
</thead>
<tbody>
<tr>
<td>.1 .2 .3 .4 .5 .6 .7 .8 .9</td>
<td>.</td>
</tr>
</tbody>
</table>

If all the inputs are 0.1 then output is 0.1; if all the inputs are 0.2 then output is 0.2; ... and so on. For all the inputs are unity, i.e. $O_i=1$, then output is inherently unity irrespective to the weights of the system which means if the probability/possibility of all events at the input is unity, then probability/possibility of the output should be, and indeed, unity. If at the terminal nodes the inputs are fuzzy probabilities/possibilities, then this result remains the same matching the consistency condition given by

$$P(A) \leq \pi(A) \quad \forall A \subset U$$

where $P(A)$ is the probability measure; $U$ is the universe of discourse; $\pi(A)$ is the possibility measure. Incidentally, if $\pi(A)$ is equal to zero, then $P(A)$ is also zero but the converse may not be true. It is interesting to note that, $\sigma_j$ having been determined using Table I and Table II, (14) can be written in the form

$$L_{\theta}(\theta) = O_j = \exp\left(-\frac{1}{2} \sum_{i} \left(\frac{(O_i-1)}{\sigma_j/w_j}\right)^2\right)$$

which means, for each input there is an associated Gaussian width $\sigma$ determined by the weight $w_j$ given by

$$\sigma_j = \frac{\sigma_j}{w_j}$$

If $w_j$ is zero, the respective $\sigma_j$ is infinite so that the input via the weight $w_j$ has no effect on the output, as the multiplication factor at the logic AND operation becomes unity. Theoretically, if all the inputs are zero, i.e. $O_i=0$, then there is still a finite node output. This is due to the fact that the Gaussian does not vanish at the point where its independent variable vanishes. From the possibilistic viewpoint, this implies that even the event probability or likelihood vanishes, the possibility remains finite. However the preceding node output never totally vanishes as far as $O_i$ is concerned or it does not make sense to consider if terminal node output $x_i$ vanishes. This is because a zero input becomes irrelevant throughout the model. Consistency condition refers to multidimensional triangular fuzzy membership functions, where, in case all independent variables are equal, the multidimensional membership function value is equal to the same number. In particular in the neural tree the membership function value is equal to the fuzzified node output. This is illustrated in figure 8. In the figure the inputs to a node are considered to be independent variables, and their associated membership function value is represented as the node output. We are
using Gaussian multi-dimensional fuzzy membership functions for likelihood computation; however the consistency condition of possibility forces us to remain in the triangular multi-dimensional membership function domain.

![Diagram](image)

**Fig. 8.** Description of the consistency condition for two-dimensional antecedent space (a); one-dimensional consequent space (singleton) (b)

4. Probabilistic-Possibilistic Base Underlying Fuzzy-Neural Tree: Tree Node with Logical OR Operation

A general $n$-weights case, the knowledge model should be devised somewhat differently for logical OR operation, and referring to figure 2, this can be accomplished as follows. The logic OR operation is fulfilled by means of the de Morgan’s law which is given below.

$$O_1 \cup O_2 = O_1 \cap O_2$$

(19)

where the complement of $O$ is given by

$$\tilde{O} = 1 - O$$

(20)

Hence for OR operation corresponding to the AND operation in (17) becomes

$$L_j(\theta) = O_j = f(O_j) = 1 - \exp\left(\frac{-1}{2\sigma^2} \sum_{i} w_{ij}^2 O_i^2\right)$$

Logical OR

To obtain (20) for the node $j$ we take the complement of the incoming node outputs $O_i$ and carry out the logic AND operation which is a multivariable fuzzy probability/possibility distribution; the result gives the complement of the output of the node $O_j$. After this operation complement of this outcome gives the desired final result (21) as $O_j$. In words, first we take the complement of the $O_i$, afterwards we execute multiplication and finally we take the complement of the multiplication. It is important to note that in this computation the Gaussian is likelihood representing a probabilistic/possibilistic entity. In this case the neural tree node output can be seen as a fuzzy rule which can be stated as

$$IF \ [O_i = X_1 \ OR \ O_2 = X_2 ] \ THEN \ [O_j \ given \ by \ (21)]$$

If all the $O_i$ inputs in (21) are zero, the output is also zero. However, if all the inputs are unity, i.e. $O_i=1$, then the node output is apparently not exactly unity because the exponent in (21) given by

$$\exp\left(\frac{-1}{2\sigma^2} \sum_{i} w_{ij}^2 O_i^2\right)$$

remains small but finite. From the probabilistic/possibilistic view point, this implies that when the event fuzzy probability/possibility $O_i$ is 1, the outcome possibility remains less than 1, which apparently does not conform to (16). However, this is circumvented by the consistency provision, namely, if all the inputs to a node are unity, the output of the node is also unity. However the preceding node output is never exactly unity as far as $O_i$ is concerned since such an output becomes irrelevant otherwise throughout the model. On the other hand, as the degree of association $w_{ij}$ goes to zero, the effect of $O_i$ on output $O_j$ in (21) becomes irrelevant, the result being consistent.

5. Computer Experiments

5.1. Experiment Nr. 1

A computer experiment is carried out, where the probabilistic-possibilistic approach is used to quantify the likelihood a certain product suits an intended purpose in ideal manner. The product concerns a laptop computer, where three laptops $A$, $B$ and $C$ are being considered for eventual purchase of one of them. The
neural tree for the problem is shown in figure 9. The input information \(x_1, x_2, \ldots, x_9\) at the terminal nodes \(T_1, T_2, \ldots, T_9\) for the respective laptops is given in table 3.

### Table 3

<table>
<thead>
<tr>
<th>Laptop</th>
<th>(x_1)</th>
<th>(x_2)</th>
<th>(x_3)</th>
<th>(x_4)</th>
<th>(x_5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>2880x1800px</td>
<td>1.0Gb</td>
<td>512Gb SSD</td>
<td>2.7Ghz</td>
<td>16Gb</td>
</tr>
<tr>
<td>B</td>
<td>1900x1080px</td>
<td>2.0Gb</td>
<td>1Tb HDD</td>
<td>2.3Ghz</td>
<td>8Gb</td>
</tr>
<tr>
<td>C</td>
<td>1680x1050px</td>
<td>512Mb</td>
<td>750Mb HDD</td>
<td>2.4Ghz</td>
<td>8Gb</td>
</tr>
</tbody>
</table>

In this experiment, the suitability of the laptops is considered for two different customers, respectively assigning different importance to some node connections. The connection weights for customer nr. 1 are shown in figure 10, and the corresponding outputs of the tree nodes are shown in figure 11. The connection weights for customer nr. 2 are shown in figure 12, and the corresponding outputs of the tree nodes are shown in figure 13. Comparing figures 10 and 12 it is to note for instance that customer nr. 2 attaches more importance to low price and to graphic power compared to customer nr. 1, while the latter attaches some more importance to the overall product quality. In figures 11 and 13 the likelihood values at the node outputs for each of the three purchase decisions are shown. The number in the uppermost line refers to
Ozer Ciftcioğlu and Michael S. Bittermann

Fig. 14. A fuzzy neural tree for design performance assessment; inner nodes performing AND operation are marked by a dot sign; inner nodes performing OR operation are marked by a plus sign.

Fig. 13. Node output values of for the three laptops using the connection weights of customer nr. 2 in figure 12.

laptop A, in the second line laptop B, and in the third line to laptop C. It is noted that the outputs of the terminal nodes in both cases are obtained by means of mapping the input information given in table 1 via respective fuzzy membership functions that are not shown for the sake of brevity of the explanation. Comparing the two figures one notes that for customer nr. 1 laptop A is the best solution, whereas for customer nr. 2 laptop B is the preferred choice. Due to the transparency of the neural tree model, the outputs of the inner nodes are interpretable, so that the reason for the higher performance of one solution over another one can be traced throughout the model.

5.2. Experiment Nr. 2

In a second experiment the fuzzy neural tree is used to compare four different architectural designs as to their respective design performance. Design performance concept refers to a single quantity representing the satisfaction of the design objectives. The fuzzy neural tree used in the experiment is seen in figure 14. From the figure it is noted that the performance concept in the exercise consists of the following major performance aspects: cost performance, utility, and form aspects. The major aspects consist of sub-aspects, for instance cost performance has two components, namely construction and energy cost performance. The construction and energy costs form the inputs to the respective terminal nodes $x_1$ and $x_2$. Both costs are demanded to be low at the same time, so that the associated cost performances are high. This is seen from the shapes of the membership functions at the terminals $T_1$ and $T_2$ that are shown in figure 15a and 15b.

Fig. 15. Membership function at terminal node $T_1$, where $x$ denotes the construction costs (a); at node $T_2$, where $x$ denotes heat energy demand (b)

The functions are right shoulders of Gaussians. For $T_1$ the center of the Gaussian is at $c_1=€280,000$ and the width of the Gaussian is $\sigma_1=€30,000$, as this is seen from
For $T_2$, the center of the Gaussian is at $c_2 = 8\text{kWh/m}^2\text{a}$ and the width of the Gaussian is $\sigma_2 = 8\text{kWh/m}^2\text{a}$, as this is seen from figure 15b. The meaning is that when a design has construction costs that are less or equal to €280,000 the design is considered to possess the ideal construction cost performance of $O_{T_1} = 1$. When a design has a heat energy demand of less or equal than $8\text{kWh/m}^2\text{a}$ the design is considered to possess the ideal energy cost performance of $O_{T_2} = 1$.

In the same way as for the terminal nodes $T_1$ and $T_2$, the outputs of the terminals $T_3 - T_5$ are obtained by means of fuzzification of design features $x_3 - x_5$ via respective fuzzy membership functions seen in figures 16a-c. For the terminal nodes $T_6 - T_{13}$ the output values of the nodes, which range from zero to unity, are obtained directly as statements from the domain expert, i.e. without involvement of membership functions. The fuzzy neural tree contains inner nodes of type AND as well as OR as seen from figure 14, where the AND operation is signified by a dot sign, and OR by a plus sign. For instance, referring to cost performance, the performance is only high under the condition that both construction and energy cost performances are high at the same time, i.e. neither of them should be low. Therefore the logic operation at the inner node $I_1$ is of the type AND operation. An example of OR operation is the contribution of the exterior form of the building to its urban context represented by inner node $I_9$. This contribution is either through the façade materialization or through the shape of the building body. Certainly the presence of both features at the same time would be ideal, however, presence of one of the two features will already yield high output at node $I_9$. The connection weights, encapsulating the designer’s domain knowledge, are shown in figure 14 next to the respective connection. From the figure it is seen for instance that the energy costs are deemed to be more important compared to the construction costs, namely by factor $3/2$. The floor plans of the designs $D1$ and $D2$ are shown in figure 17 and 18 respectively. Design $D3$ is the same as $D1$ concerning building shape and interior layout, while these designs differ with respect to their balcony geometries. Design $D4$ is the same as $D2$ concerning building shape and interior layout, while they differ with respect to balcony, which does not exist in $D4$. Due to the high redundancy of information, plan drawings of $D2$ and $D4$ are omitted.
Exterior renderings of the four designs subject to probabilistic-possibilistic performance analysis are seen in figures 19-22.
The designs’ input values $x_1$-$x_5$ belonging to the terminals $T_1$-$T_5$ are provided below the renderings in each figure, in tabular form, as well as the terminal node outputs $O_{T1}$-$O_{T13}$ in the form of plot.

On order to compare the performance of the four designs, the outputs of the inner nodes $O_{I3}$-$O_{I6}$, $O_{I8}$ and $O_{I9}$ are shown for all designs in the same plot in figure 23; the outputs of the penultimate nodes $O_{I1}$, $O_{I2}$ and $O_{I7}$ in figure 24a; and the root node output $p$ in figure 24b.

From the results it is seen that design $D1$ is slightly superior compared to the other designs. Taking the relative importance among the penultimate performance aspects cost, form and utility as $O_{I1}=O_{I2}=O_{I7}=1/3$, The output for $D1$ at the root node is highest, namely $p=0.73$. This result can be interpreted as the likelihood $D1$ is ideal being 73%. Therefore, among the four alternatives $D1$ is the preferred choice to be built, and a photograph of the constructed building based on $D1$, is shown in figure 25.
5.3. Experiment Nr. 3

Design Problem

A computer experiment is carried out, where the probabilistic-possibilistic modelling is used for automated design. In this case the number of designs subject to neural tree assessment is much greater than the low number in the first two experiments. Therefore, in contrast to the previous experiments, no expert assessment in the form of a fuzzy statement is used in the third experiment, i.e. all terminal node outputs are computed from the node input data using respective fuzzy membership functions.

The task concerns a room with the width $l_1=5.0$ and the length $l_2=9.0$ consisting of four walls, a door, a window, and a chair, as seen in figure 26. The problem is positioning the door, windows and chair, as well as determining the orientation of the chair in such a way that the room has a number of desirable properties. The role of the neural tree is to determine the likelihood that a solution is an ideal one, in order to eventually arrive at a solution with maximum likelihood. The decision variables of the problem are shown in figure 26. Decision variable $d_1$ denotes the distance of the door from the lower-left room corner, the door being located at some place along wall 1, where $w_d/2<d_1<l_1-w_d/2$; $w_d$ denotes the width of the door. Variable $d_2$ denotes the Manhattan distance of the window from the lower right room corner, being located at some place along wall 3 and wall 4, where $w_w/2<d_2<l_1+l_2-w_w/2$; $w_w$ denotes the width of the window. $d_3$ and $d_4$ denote the position of the chair’s centre point relative to the lower-left room corner, respectively in the direction of wall 1 and wall 2; $w_1c/2<d_3<l_2-w_1c/2$ and $w_2c/2<d_4<l_1$; $w_1c$ denotes the width of the chair in the direction of the axis of wall 1; $w_2c$ denotes the width of the chair in the direction of the axis of wall 2. $d_5$ denotes four different orientations of the chair, namely facing wall 1, or wall 2, or wall 3 or wall 4, which are respectively encoded by integer $d_5\in\{0,1,2,3\}$.

Based on the decision variables seven elemental perception and utility related properties of the room are obtained. These are denoted by $x_1, x_2, ..., x_7$, and are converted to fuzzy membership degrees at terminals of a neural tree given by $O_1(x_1), O_2(x_2), ..., O_7(x_7)$. The neural tree will be presented following the description of each membership function used to obtain the terminal outputs. It is to note that terminal node output $O_i(x_i)=T_i$ expresses the fuzzy probability that the solution at hand is matching designer’s demands as to an ideal solution. $x_i$ denotes the degree of perception of the door from the chair, which is computed using a probabilistic perception modelling approach described in another publication. In that approach perception is defined as the probability an observer realizes the presence of an object in his mind. The demand concerning the perception is expressed by the fuzzy membership
function \( O_1(x_1) = O_{T1} \) in figure 27a, which is a Gaussian centered at \( c_1 = 0.025 \) having width \( \sigma_1 = 0.01 \). This implies that the ideal value for the perception is \( x_1 = 0.025 \), and deviation from this value in positive or negative direction diminishes the membership degree \( O_{T1} \), i.e., deviation diminishes the likelihood that the solution being the source of \( x_1 \) is an ideal solution.

\( x_2 \) denotes the degree of perception of the window from the chair, which is computed using the same approach as for \( x_1 \). The demand concerning the perception is expressed by the fuzzy membership function \( O_2(x_2) = O_{T2} \) shown in figure 27b. For \( 0 \leq x_2 \leq 0.25 \) \( O_2 \) is the left shoulder of a Gaussian centered at \( c_2 = 0.25 \) having width \( \sigma_2 = 0.14 \), while for \( x_2 > 0.25 \) \( O_2 = 1.0 \). This implies that ideal values for the perception are \( x_2 \geq 0.25 \), and deviation from the threshold in negative direction diminishes the fuzzy probability \( O_{T2} \).

\( x_3 \) denotes the perceptual density computed at the centre point of the chair, which is a quantity derived from the probabilistic perception modelling approach mentioned before, and it is also described in 73. In that work perceptual density is defined as the differential perception of the whole visible environment at a point \( w \), per unit \( w \), where \( w \) is the location on a line, along which the visual perception is computed. The demand concerning the perceptual density is expressed by the fuzzy membership function \( O_3(x_3) = O_{T3} \) shown in figure 27c. For \( 0 \leq x_3 \leq 0.205 \) \( O_3 \) is the left shoulder of a Gaussian centered at \( c_3 = 0.205 \) having width \( \sigma_3 = 0.002 \), while for \( x_3 > 0.205 \) then \( O_3 = 1.0 \). This implies that ideal values for the perception are \( x_3 \geq 0.205 \), and deviation from the threshold in negative direction diminishes the fuzzy probability \( O_{T3} \).

Fig. 27. fuzzy membership functions at the terminal nodes \( O_{T1}(a); O_{T2}(b); O_{T3}(c); O_{T4}(d); O_{T5}(e); O_{T6}(f); O_{T7}(g) \)
denotes the Euclidean distance between the centre point of the chair and the centre point of the door. The demand concerning the distance is expressed by the fuzzy membership function $O_4(x_4)=O_{T4}$ shown in figure 27d, given by

$$O_{T4}=1-\exp[-0.5(x_4^2/\sigma_4^2)]$$

where $\sigma_4=2.7$. This implies that the chair is demanded to be not near to the door.

$x_5$ denotes the angle between the respective frontal direction vectors of the door and the chair. The demand concerning the angle is that it should not be zero, as this is expressed by the fuzzy membership function $O_5(x_5)=O_{T5}$ in figure 27e, which is given by

$$O_{T5}=1-\exp[-0.5(x_5^2/\sigma_5^2)]$$

where $\sigma_5=\pi/6$.

$x_6$ denotes the perceptual density at the centre point of the door. The demand concerning the density is expressed by the fuzzy membership function $O_6(x_6)=O_{T6}$ in figure 27f. For $0 \leq x_6 \leq 0.20$ $O_{T6}$ is the left shoulder of a Gaussian centred at $c_6=.20$ having width $\sigma_6=.006$, while for $x_6>0.20$ $O_{T6}=1$. This implies ideal values for the perception are $x_6 \geq .20$, and deviation from the threshold in negative direction diminishes the fuzzy probability $O_{T6}$.

$x_7$ denotes the degree of perception of the window from the door, which is computed using the same probabilistic approach mentioned earlier. The demand concerning the perception is expressed by the fuzzy membership function $O_7(x_7)=O_{T7}$ shown in figure 27g. For $0 \leq x_7 \leq 0.17$ $O_{T7}$ is the left shoulder of a Gaussian centred at $c_7=.17$ having width $\sigma_7=.10$, while for $x_7>0.17$ $O_{T7}=1.0$. This implies that ideal values for the perception are $x_7 \geq .17$, and deviation from the threshold in negative direction diminishes the fuzzy probability $O_{T7}$.

Having obtained the fuzzy probabilities $O_1(x_1), O_2(x_2), \ldots, O_7(x_7)$ at the terminal nodes $T_1, T_2, \ldots, T_7$ of a neural tree, the information is subjected to further fuzzy probabilistic processing at the inner nodes of the tree. This is done, so that the likelihood is obtained regarding the event that a solution at hand is ideal. The neural tree of the experiment is shown in figure 28, where the plus sign denotes the logical $\text{OR}$ operation at the respective node, and the dot sign the logical $\text{AND}$ operation. From the tree structure it is to note that $x_1$, $x_2$ and $x_3$ are independent variables determining the performance of the chair denoted by $O_{I4}$, whereas $x_4$, $x_5$, $x_6$, and $x_7$ are independent variables determining the performance of the door, which is denoted by $O_{I5}$ in the figure.

$\text{Fig. 28.}$ Fuzzy neural tree structure of the experiment; inner nodes performing AND operation are marked by a dot sign; inner nodes performing OR operation are marked by a plus sign.

$\text{Fig. 29.}$ The connection weight values.

The values of the connection weights are shown in figure 29.
The problem at hand is maximizing the fuzzy likelihood given by the outputs $O_{14}$ and $O_{15}$. The maximization is carried out by the method of random search, accomplished through the method of genetic algorithm. Genetic algorithm is a stochastic optimization method from the domain of computational intelligence. The algorithm starts from a number of random solutions referred to as members of a population. Each member satisfies the respective objective functions – in the present case given by $O_{14}$ and $O_{15}$ – to some degree, which is termed fitness. In the algorithm population members with a comparatively high fitness will be favored over solutions with low fitness, by giving the former a higher chance to remain in the population and to produce new solutions by combining fit solutions. The combination among solutions is referred to as crossover operation, and it is carried out among pairs of population members referred to as parents. Crossover entails that the parameters constituting a parent are treated as binary strings, and portions of the strings are exchanged among the pair of solutions to create new solutions with features from both parents. This process is repeated for several iterations, and due to the probabilistic favoring of fit solutions, eventually optimal solutions appear in the population. It is emphasized that two objectives are subject to simultaneous satisfaction; namely maximizing $O_{14}$ and $O_{15}$, and these are conflicting. The optimization is accomplished by means of a multi-objective genetic algorithm using Pareto dominance as criterion for solutions’ fitness determination. It is to note that the outputs at the penultimate nodes $I_{14}$ and $I_{15}$ are taken as likelihood in this case.

The population of the first generation of the genetic algorithm is shown in figure 30 in the two-dimensional objective function space formed by the node outputs $O_{14}$ and $O_{15}$.

The Pareto front in objective function space found by the multi-objective genetic algorithm is shown in figure 32, where three representative solutions are highlighted. These solutions are shown in figures 33-35 respectively as 3-D renderings from a top view.

Results

One of these randomized solutions is exemplified in figure 31a by means of a 3-D rendering. From figure 31b it is noted that several demands are poorly met by this design, where the ones that are least met concern perceptual density at the chair, perception of the window from the door, and the chair not to be located in frontal direction of the door.
The transparency of the probabilistic-possibilistic approach presented in this work provides detailed insights into the differences among Pareto optimal solutions. This is in contrast to conventional multi-objective optimization works, where objective functions generally are explicit functions, so that comparison among Pareto solutions is to be done with respect to the objective function values. Due to the involvement of implicit functions at the inner node outputs, and explicit functions at the terminals, the probabilistic-possibilistic approach permits to completely and meaningfully trace the reasons for a certain likelihood output at the root node. For instance, comparing solution $a$ in figure 33 with $b$ in figure 34, one notes that $b$ has a higher chair performance, which is particularly due to higher perception of the window, and better perception of the door from the chair. Conversely solution $a$ is superior to $b$ with respect to performance of the door, which is particularly due to increased perception of the window.
Fig. 35. Pareto solution c shown in figure 32 with $p=0.93$; rendering from top view (a); node outputs (b)

from the door. The same differences are magnified comparing solution a with solution c in figure 35. Namely, the chair performance is almost ideal in solution c, whereas the door performance is significantly lower compared to a, which is particularly due to diminished perception of the window from the door in c. One notes that in all of the Pareto optimal solutions the chair is oriented facing wall 1. One notes that in all of the Pareto optimal solutions the chair is oriented facing wall 1. Another common property among the Pareto solutions is that in all of them the requirement modelled at terminal T5, namely the premise chair not being in front of the door, is poorly satisfied, as seen from figure 35 compared to the initial solution in figure 31. This requirement apparently is in conflict with other ones, so that apparently is necessary to sacrifice satisfaction of this requirement in order to attain a high performance at the penultimate node level.

Another comparison among Pareto solutions can be carried out making use of the $p_{max}$ criterion described in 76, which is given by

$$p_{max} = \frac{Q_1^2 + Q_2^2 + \ldots + Q_n^2}{Q_1 + Q_2 + \ldots + Q_n}, \quad Q_1, Q_2, \ldots, Q_n > 0$$

(22)

where $Q_1, Q_2, \ldots, Q_n$ are n-number of penultimate node outputs that together form the output at the root node of the neural tree. $p_{max}$ quantifies the highest performance a solution can attain, when the weight vector associated with the root node connections – in our case $w_{14p}, w_{15p}$ in figure 29, is selected in a most favorable manner, namely such that the root node output is maximal for the particular solution at hand. The maximal performance for an input pattern is denoted by $p_{max}$ and its computation is described in 76. Considering the three exemplary Pareto solutions with respect to $p_{max}$ one notes that solution b in figure 34 has the highest value among the Pareto solutions, namely $p_{b, max}=0.94$. The weight vector of this solution has the two components $w_{14p}=O_{14}=0.95$ and $w_{15p}=O_{15}=0.92$. Based on this ‘best’ weight vector, the room performance $p$ is computed for solutions a and b, and the results $p_a=O_{14a}O_{14b}=0.87$ and $p_b=O_{15a}O_{15b}=0.93$ are shown in figure 32 next to the respective solution. The ‘best’ weight vector implies that in the absence of any particular a-priori preference regarding importance among door and chair, in this design problem the performance of chair plays a slightly more important role compared to the performance of the door, namely by about 3%. This is relevant information for a decision maker that could be obtained due to application of the probabilistic/possibilistic framework.

6. Conclusions

Knowledge modelling by a novel neural tree structure is presented where fuzzy probability/possibility is central. The novelty is to introduce the probabilistic likelihood concept to neural tree computation, so that the values at the nodal outputs have an interpretation as a fuzzy probability. This probability refers to the event that the stimulus presented at the model input is likelihood with respect to the anticipated ideal input. The neural tree effectively is a fuzzy logic system having a characteristic aspect of transparency which implies the expert knowledge can directly be brought into play in the form of weight information. Another consequence of the transparency is that the reason for a certain output value at the root node can be traced as a root-cause, and conclusions with respect to desirable modifications of the input stimulus are obtained by a human expert. Clearly, it is to note that the fuzzy neural tree, in contrast to existing neural tree works in the literature, is a knowledge driven model. Namely, it is not a conventional black box type of tree model aiming to minimize the model error based on a data set. In the present approach the neural tree serves as an emulator of human-like reasoning, which is far beyond conventional use of neural trees that carry out function approximation without salient interpretability. It is to note that, depending on the usage of the node output, the output has an interpretation as likelihood, or as fuzzy probability, which is a unique property of the fuzzy
neural tree presented in this work. The theoretical base for the consistency condition employed is provided, yielding effective application of possibility theory for knowledge modeling in design, at large. This way the approach combines the favorable features of the two major knowledge modelling approaches, neural computation and fuzzy modelling which respectively are capable of dealing with the complexity of information and providing transparency. Based on the new insights, the analysis of architectural products is executed and their mystic soft attributes are uniquely defined and quantified mathematically. The exemplary applications for diverse purposes in architectural and interior design indicate the generic nature of the framework.

Acknowledgements

This research is carried out under the 2221 program of The Scientific & Technological Research Council of Turkey (TUBITAK). The support is gratefully acknowledged.

References


32. T. Xu, Y. Wang, and Z. Zhang, Pixel-wise skin colour detection based on flexible neural tree, IET Image Processing 7 (2013) 751 - 761.


40. S. Bouaziz, A. Alimi, and A. Abraham, Evolving flexible beta basis function neural tree for nonlinear systems, Neural Networks (IJCNN), (2013) 1-8.


43. Y. Chen, A. Abraham, and B. Yang, Feature selection and classification using flexible neural tree, Neurocomputing - Selected Papers from the 7th Brazilian Symposium on Neural Networks (SBRN '04), 70 (2006) 305-313.


48. A. Rani, B. Raman, and S. Kumar, A robust watermarking scheme exploiting balanced neural tree for rightful


