An Enhanced Fuzzy Min–Max Neural Network for Pattern Classification

Mohammed Falah Mohammed and Chee Peng Lim

***Abstract*— An enhanced fuzzy min–max (EFMM) network is proposed for pattern classification in this paper. The aim is to overcome a number of limitations of the original fuzzy min–max (FMM) network and improve its classification performance. The key contributions are three heuristic rules to enhance the learning algorithm of FMM. First, a new hyperbox expansion rule to eliminate the overlapping problem during the hyperbox expansion process is suggested. Second, the existing hyperbox overlap test rule is extended to discover other possible overlapping cases. Third, a new hyperbox contraction rule to resolve possible overlapping cases is provided. Efficacy of EFMM is evaluated using benchmark data sets and a real medical diagnosis task. The results are better than those from various FMM-based models, support vector machine-based, Bayesian-based, decision tree-based, fuzzy-based, and neural-based classifiers. The empirical findings show that the newly introduced rules are able to realize EFMM as a useful model for undertaking pattern classification problems.**

***Index Terms*— Fuzzy min–max (FMM) model, hyperbox structure, neural network learning, pattern classification.**

I. Introduction

An artificial neural network (ANN) is a computational model that consists of an interconnected group of artificial neurons that simulates the biological neural system in our brain [1]. Nowadays, ANNs are used in many fields, e.g., healthcare [2], power systems [3], and fault detection [4]. Pattern classification is one of the active ANN application domains [5]. As an example, ANN models have been successfully applied to classification tasks in business and science [5] and industrial fault detection and diagnosis [6].

In terms of ANN training, one of the main problems related to batch learning, such as in standard Multi-Layer Perceptron (MLP) and Radial Basis Function (RBF), is catastrophic forgetting [7], [8]. Catastrophic forgetting is concerned with the inability of a learning system to remember what it has previously learned when new information is absorbed [9]. The backpropagation ANN was found to create a new solution based on the most recent information only, when it was given two or more pieces of information to learn [7]. This is obviously different from the functionality of our brain.

Manuscript received February 20, 2013; revised January 3, 2014; accepted March 28, 2014. Date of publication April 30, 2014; date of current version February 16, 2015. This work was supported by the Fundamental Research under Grant 6711195 and Grant 6711229.

M. F. Mohammed is with the School of Electrical and Electronic Engineering, University Science Malaysia, Nibong Tebal 14300, Malaysia (e-mail: mofmah2009@gmail.com).

C. P. Lim is with the Centre for Intelligent Systems Research, Deakin University, Geelong, VIC 3220, Australia (e-mail: chee.lim@deakin.edu.au).

Digital Object Identifier 10.1109/TNNLS.2014.2315214

The catastrophic forgetting problem is also addressed as the stability plasticity dilemma [10]–[13]. The dilemma addresses several issues in learning systems, e.g., how a learning system can be plastic enough to learn new knowledge and, at the same time, be stable enough to retain previously learned knowledge from corruption [14]. Solving the stability plasticity dilemma is crucial especially when an ANN has to learn from data samples in one-pass using an online learning strategy.

To overcome the stability-plasticity dilemma, a number of ANN models have been proposed, which include the adaptive resonance theory (ART) networks [10], [11] and fuzzy min–max (FMM) networks [12], [15]. Specifically, there are two FMM networks, i.e., a supervised classification model [12] and an unsupervised clustering model [15]. The supervised classification FMM model [12] is used throughout this paper.

The supervised FMM network [12] (hereafter addressed as FMM) entails a dynamic network structure with an online learning capability. Its number of prototypes (encoded as hyperboxes) can be increased when necessary. This avoids the problem of retraining as faced by ANNs with offline or batch learning methods. There are several salient learning properties associated with FMM as follows [12].

1) *Online learning:* the ability to learn new information stability without losing old information. This property is important to solve the stability plasticity dilemma.
2) *Nonlinear separability:* the ability to build a nonlinear decision boundary to separate different classes.
3) *Overlapping classes:* the ability of the nonlinear decision boundary to minimize misclassification by eliminating the overlapping regions of different classes.
4) *Training time:* the ability to learn and revise the nonlinear decision boundary with one-pass learning through the training data within a short training time.

Despite the salient properties of FMM, there are limitations pertaining to the FMM learning dynamics. As such, the main motivation of this research is to improve the FMM learning algorithm and enhance its classification ability. Specifically, we discover that the current hyperbox expansion process can affect FMM negatively by increasing the overlapping regions between different classes. Besides that, the existing hyperbox overlap test rule is unable to detect all overlapping regions, and this affects the subsequent hyperbox contraction process.

In this paper, an enhanced FMM (EFMM) network is proposed. The main contributions of this research comprise three heuristic rules that overcome the current limitations of the FMM learning algorithm, as follows:

1) a new hyperbox expansion rule to minimize the overlapping regions of hyperboxes from different classes;
2) an extended hyperbox overlap test rule to identify all overlapping regions of hyperboxes from different classes;
3) a new hyperbox contraction rule to solve overlapping cases that are not covered by the existing contraction process.

This paper is organized as follows. Hybrid fuzzy-neural models and FMM-based models are reviewed in Section II. In Section III, the FMM learning algorithm is analyzed. The EFMM model and its evaluation are presented in Sections IV and V, respectively. Conclusion is given in Section VI.

II. Related Work

Pattern recognition and classification have been intensively researched with fuzzy sets after Zadeh introduced the fuzzy set theory [16] in 1965. Since that time, researchers have paid great attention to develop intelligent systems by combining the inference properties of fuzzy systems (in handling uncertain information) and ANNs learning ability [17]. A variety of hybrid models based on the combination of fuzzy sets and ANNs have been developed. In this section, some examples of hybrid fuzzy-neural models are explained. A review of FMM-related models is then presented.

A. Hybrid Fuzzy-Neural Models

Three clustering techniques, i.e., k-means, fuzzy k-means, and the subtractive models, were used to obtain rule antecedent and consequent in Mamdani-type and Takagi-Sugeno-Kang-type of fuzzy inference systems (FISs) [18]. They are able to process a large volume of raw data automatically, identify the most significant patterns, and extract useful if-then rules. The results from the Iris and Mackey Glass problems show that the subtractive model outperforms other clustering methods [18].

An investigation hybrid type-1 and -2 FISs with the modular neural networks (MNNs) for multimodal biometry tasks was presented [19]. Simple genetic algorithms (GAs) are used to optimize the FIS models and allow an accurate comparison of type-1 and -2 fuzzy logic in the proposed model. The results indicate that type-2 FIS performs better than type-1 FIS in tackling biometry face, fingerprint, and voice recognition problems [19]. Besides that, a multiobjective hierarchical GA based on the micro-GA for optimization of MNNs was proposed [20]. The model has the ability to divide the data samples automatically into several submodules, and select the appropriate data samples for training and testing. The GA is used to determine the number of submodules and the percentage of data for training purposes. An experiment with a set of iris biometric data for human recognition indicates that the proposed method is able to yield good results [20].

A modified Sugeno integral method with an interval type-2 fuzzy logic (IT2FL) model was proposed [21]. The modifications include changing the type-1 fuzzy logic equations of Sugeno measures and Sugeno integral with fuzzy measures and IT2FL. These modifications allow multiple sources of information to be combined with a higher degree of uncertainty. The empirical results show improved recognition rates when benchmark face databases were used [21].

The graphic processing unit (GPU)-fuzzy neural network (FNN) model, i.e., an implementation of FNN using the GPU platform, was introduced [22]. The model aims to reduce the FNN training time by running blocks of threads instead of single-thread CPU processes. The results show a reduction of training time by 30% for problems with high-dimensional attributes [22].

In [23], an improved generalized ART (IGART) model was proposed. A Laplacian-likelihood function is derived to provide a new definition for the vigilance function in ART. In addition, network pruning and rule extraction capabilities are incorporated into IGART. The experimental results demonstrate the effectiveness of IGART in solving real-world problems in power systems, including diagnosis of transformer faults and condition monitoring of a circulating water system in a power generation plant [23].

A hybrid learning algorithm that consists of a clustering model as well as the recursive least square and adaptive backpropagation methods for training of fuzzy wavelet neural networks (FWNNs) was developed [24]. Based on the halving method to find polynomial roots, the algorithm does not include excessive terms in its formulation and, therefore, is less complex [24]. The results in function approximation, nonlinear dynamic plant identification, and chaotic time series prediction problems show that the algorithm is faster and is able to produce better root mean square error rates as compared with those from other FWNN training methods [24].

A generic evolving neuro-FIS (GENEFIS) was proposed in [25]. GENEFIS entails a tradeoff between a high predictive accuracy and a parsimonious rule base. The results from the Mackey–Glass time series data, emission of real car engine data, and valuation of residential premises data depict better predictive accuracy rates with less complex structures as compared with standard evolving neuro-fuzzy systems [25].

A novel recurrent FNN known as the interactively recurrent self-evolving FNN (IRSFNN) for prediction and identification of dynamic systems was proposed [26]. The IRSFNN model incorporates local and global feedback structures. It undertakes time-varying system identification and time series prediction problems with two major characteristics, i.e., online learning and recurrent dynamics. The results with two types of dynamic system problems and three prediction problems show enhanced performances of IRSFNN as compared with those from other recurrent FNN models [26].

B. Review of FMM-Related Models

The FMM structure comprises a number of hyperboxes. Each hyperbox occupies a region of the n-dimensional pattern space. As FMM supports online learning, it is able to create new hyperboxes based on the incoming data samples and to associate them with new classes or refine the existing ones without retraining. Over the years, many FMM variants, which are largely based on the two original models [12], [15] have been proposed. A review of these models is as follows.

An extension of the FMM classification model known as the General Fuzzy Min-Max (GFMM) network was proposed [27]. GFMM is developed based on the expansion and contraction principle. It is able to handle both labeled and unlabeled data. Advantages of GFMM include processing input data with confidence limits, avoiding retraining, and combining both supervised and unsupervised learning strategies within a single algorithm [27]. Later, a general reflex FMM neural network was proposed [28]. GRFMM combines both FMM clustering and classification algorithms as well as the concept of human reflex mechanism together [28]. It is able to identify the underlying data structure as compared with GFMM.

A stochastic FMM network for reinforcement learning was proposed [29], which was an extension of the reinforcement FMM model [30]. The concept of random hyperboxes is introduced. Unlike FMM, the proposed extension [29] uses a stochastic automaton, instead of the action label (or class label), where the probability vector of the stochastic automaton determines the corresponding action through random selection. The location, hyperbox boundaries, and probability vector of each stochastic automaton are adjusted by the stochastic FMM network. An adaptive resolution min–max network was proposed as a dual classifier model [31]. Two algorithms, i.e., the adaptive resolution classifier (ARC) and pruned ARC, are devised. The hyperbox expansion process is not limited by a fixed maximum size, to overcome some of the undesired properties in the original FMM model. The network structure is also less complex as compared with FMM [31].

In [32], an inclusion/exclusion fuzzy hyperbox classifier was proposed. This model creates two types of hyperboxes, i.e., the inclusion and exclusion hyperboxes. The inclusion hyperboxes are used to contain input patterns from the same class. Other overlapped patterns are contained by the exclusion hyperboxes. The use of the exclusion hyperboxes reduces the training process from three steps (expansion, overlap test, and contraction) to two steps (expansion and overlap test). This is achieved through a contentious area of the pattern space to approximate the complex data topology, which helps solve the overlapping problem in FMM [32].

A weighted FMM network (WFMM) was proposed [33]. Both the hyperbox contraction process and overlap test do not restrict hyperbox expansion in WFMM. A new membership function and a learning method is defined for WFMM [33]. On the other hand, a new model called the Fuzzy Min-Max Neural Network Classifier with Compensatory Neurons (FMCN) was proposed [34]. Based on the CN architecture, FMCN is a supervised classification model that supports online learning. The CNs are activated when the test sample falls in the overlapped region between two different classes. The hyperbox contraction process is also eliminated. A data-core-based FMM neural network (DCFMN) model for pattern classification was proposed [35]. The DCFMN model uses new membership functions for two types of neurons, i.e., classifying and overlapping neurons. The membership functions take three factors into consideration, i.e., noise, geometric center of the hyperbox, and data core. The contraction process is removed in DCFMN. When a new pattern falls into the overlapped area of different classes, the membership function of the overlapping neural is used to determine the target class [35].

In [36], a modified FMM (MFMM) network to improve the FMM classification performance was proposed. The aim is to tackle the problem related to the formation of a small number of large hyperboxes in the network. The data set is divided into three subsets: a training set for learning, a prediction set for pruning and rule extraction, and a test set for performance evaluation. Later, MFMM was further enhanced with a GA-based rule extractor to form the FMM-GA model [37].

TABLE I
SUMMARY OF FMM-RELATED NETWORKS

Model	Characteristics
FMM [12]	cater for supervised learning; combine ANN and fuzzy set theory into a common framework for tackling pattern classification problems; support on-line learning and avoid re-training
FMM [15]	cater for unsupervised learning; combine ANN and fuzzy set theory into a common framework for tackling pattern clustering problems; support on-line learning
General Fuzzy Min-Max (GFMM) [27]	handle both labeled data and unlabeled data; cater for supervised and unsupervised learning.
General Reflex Fuzzy Min-Max Neural Network (GRFMN) [28]	combine both FMM clustering and classification algorithms; deploy the concept of human reflex mechanism to solve the problem of class overlaps.
Stochastic FMM [29]	use a stochastic automaton, instead of the action label or class label in original FMM; cater for reinforcement learning.
Adaptive resolution Min-Max model [31]	comprise the Adaptive Resolution Classifier and Pruned Adaptive Resolution Classifier; the hyperbox expansion process is not limited by a fixed maximum threshold as in FMM; possess a less complex network structure than that of FMM.
Inclusion/Exclusion fuzzy hyperbox classifier [32]	use the inclusion hyperboxes to contain input patterns from the same class; use the exclusion hyperboxes to contain overlapped patterns.
Weighted FMM network (WFMM) [33]	does not restrict hyperbox expansion through the hyperbox contraction and overlap test steps; define a new membership function; define a new learning method for hyperbox creation, expansion, contraction, and weight update.
FMM with compensatory neuron (FMCN) [34]	use the compensatory neuron architecture; use new membership functions for two types of neurons, classifying neurons and overlapping neurons.
Data-Core-Based Fuzzy Min–Max Neural Network (DCFMN) [35]	use membership functions for classifying neurons and overlapping neurons; remove the contraction process.
Modified FMM (MFMM) [36]	perform network pruning and rule extraction; use the Euclidean distance and membership function for prediction.
Modified FMM with Genetic Algorithm (MFMM-GA) [37]	perform network pruning and rule extraction; use the Euclidean distance and membership function for prediction; incorporate the Genetic Algorithm for rule extraction
Offline and online FMM-CART models [38, 39]	use FMM for data classification and CART for rule extraction; support offline and online learning properties for fault detection and diagnosis tasks.

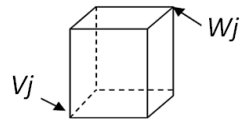

Fig. 1. 3-D hyperbox.

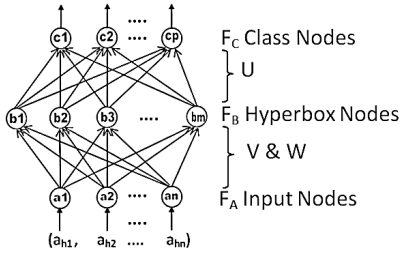

Fig. 2. Three-layer FMM network.

On the other hand, a hybrid model consisting of FMM and the classification and regression tree (CART) was proposed in [38]. FMM-CART uses an offline learning method to learn and classify various motor faults. It is also able to extract useful explanatory rules for justifying its predictions. The experiments with real data from inductor motors demonstrate the effectiveness of FMM-CART for motor fault diagnosis. Later, FMM-CART was further enhanced to perform online learning in [39]. The experimental results with real data show its usefulness for fault detection and diagnosis of inductor motors in an online learning environment. Table I shows a summary of FMM-based networks discussed in the section.

III. Supervised FMM Network

As stated in [40], the concept of FMM is similar to that of fuzzy ART and FALCON-ART. These models use fuzzy hyperboxes as the fundamental computing elements. The detailed FMM learning algorithm is as follows.

A. FMM Learning Algorithm

The FMM learning algorithm comprises a three-step process, viz., hyperbox expansion, hyperbox overlap test, and hyperbox contraction. Learning in FMM starts using a data set consisting of input patterns and target classes, A_h, $h = 1, \ldots, N$, where N is the total number of data samples. Based on the data samples, FMM creates a number of hyperboxes. Each hyperbox is represented by a minimum and maximum point in an n-dimensional space within a unit hypercube (I^n). Fig. 1 shows a 3-D hyperbox with its minimum point (V_j) and maximum point (w_j).

Each hyperbox fuzzy set is defined as

$$B_j = (\{A_h, v_j, w_j, f((A_h, v_j, w_j))\}) \quad \forall A_h \in I^n \tag{1}$$

where B_j is the hyperbox fuzzy set, $A_h = (a_{h1}, a_{h2}, \ldots, a_{hn})$ is the input pattern, and $v_j = (v_{j1}, v_{j2}, \ldots, v_{jn})$ and $w_j = (w_{j1}, w_{j2}, \ldots, w_{jn})$ are the minimum and maximum points of B_j, respectively.

When an input pattern is contained in a hyperbox, it is said that the input pattern has a full class membership of the respective hyperbox. The hyperbox size is controlled by a user-defined parameter called the expansion coefficient (Θ). The FMM structure consists of three layers, as in Fig. 2. F_A is the input layer, which has the number of input nodes equals to the number of input features. F_B is the hyperbox layer. Each F_B node represents a hyperbox fuzzy set created during the learning process. The connections between F_A and F_B nodes are the minimum and maximum points. They are stored in two matrices (V and W), while the membership function is F_B's transfer function [12]. F_C is the output layer. Its number of nodes equals to the number of output classes. Each F_C node represents a class. The output of each F_C node represents the degree to which A_h fits within an output class. The connections between F_B and F_C nodes are binary values, and are stored in matrix U, as defined by

$$u_{jk} = \begin{cases} 1 & \text{if } b_j \text{ is a hyperbox for class } C_k \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

where b_j is the jth F_B node and C_k is the kth F_C node. The outputs can be directly employed when a soft decision is required. For a hard decision, the winner-take-all principle [41] can be used to choose the F_C node with the highest value as the predicted class.

In general, when a new training pattern is provided, FMM uses the membership function, which is the degree-of-fit of a sample with respect to a hyperbox [12], to find the closest hyperbox that matches the sample. The membership function is calculated using (3), as follows:

$$\mathrm{B}_j(\mathrm{A}_h) = \frac{1}{2n}\sum_{i=1}^{n}[\max(0, 1-\max(0, \gamma \min(1, a_{hi} - w_{ji}))) + \max(0, 1 - \max(0, \gamma \min(1, v_{ji} - a_{hi})))] \tag{3}$$

where B_j is the membership function of the jth hyperbox, $A_h = (a_{h1}, a_{h2}, \ldots, a_{hn}) \in I^n$ is the hth input pattern, and γ is a sensitivity parameter that regulates how fast the membership decreases as the distance between A_h and B_j increases.

Note that FMM is a type of hybrid fuzzy-neural model that incorporates fuzzification into its ANN learning algorithm. It retains the basic properties and functions of an ANN, but contains some fuzzified elements. This is realized through the fuzzy hyperbox membership function of (3). By regulating the user-defined sensitivity parameter (γ) in (3), a fuzzy interpretation for an input pattern to attain either a full or reduced membership with respect to each hyperbox can be achieved. Fig. 3 shows the membership function in a 2-D input space. The minimum point is $v_j = (0.4, 0.2)$ and the maximum point is $w_j = (0.6, 0.4)$. The sensitivity parameter (γ) is set to 0.5, 1, 3, and 5 in Fig. 3(a)–(d), respectively. The sensitivity parameter regulates the degree of changes from a full membership (within the hyperbox) to a reduced membership (outside the hyperbox).

B. Hyperbox Expansion, Overlap Test, and Contraction

Hyperbox expansion is performed to include the input pattern in the respective hyperbox class, provided that the

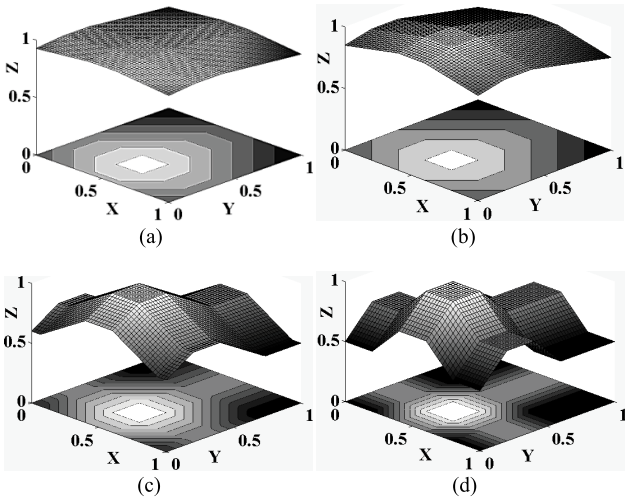

Fig. 3. Membership for different γ settings. (a) $\gamma = 0.5$. (b) $\gamma = 1$. (c) $\gamma = 3$(d). (d) $\gamma = 5$.

hyperbox size does not exceed the expansion coefficient, Θ, whereby $0 \leq \Theta \leq 1$. As such, when hyperbox B_j is expanded to include a new input pattern A_h, the following constraint must be met [12]:

$$n\Theta \geq \sum_{i=1}^{n} (\max(w_{ji}, a_{hi}) - \min(v_{ji}, a_{hi})). \tag{4}$$

If the constraint in (4) is not satisfied, a new hyperbox is created to encode the pattern. This growth process allows new classes to be added without retraining. However, the expansion process can lead to overlaps among hyperboxes. As such, an overlap test is performed to check whether any overlapping regions occur, after the expansion process, between the expanded hyperbox and existing ones that belong to other classes. This overlap test is based on four cases [12].

Case 1:

$$V_{ji} < V_{ki} < W_{ji} < W_{ki}, \quad \delta^{\text{new}} = \min(W_{ji} - V_{ki}, \delta^{\text{old}}). \tag{5}$$

Case 2:

$$V_{ki} < V_{ji} < W_{ki} < W_{ji}, \quad \delta^{\text{new}} = \min(W_{ki} - V_{ji}, \delta^{\text{old}}). \tag{6}$$

Case 3:

$$V_{ji} < V_{ki} < W_{ki} < W_{ji}$$
$$\delta^{\text{new}} = \min(\min(W_{ki} - V_{ji}, W_{ji} - V_{ki}), \delta^{\text{old}}). \tag{7}$$

Case 4:

$$V_{ki} < V_{ji} < W_{ji} < W_{ki}$$
$$\delta^{\text{new}} = \min(\min(W_{ji} - V_{ki}, W_{ki} - V_{ji}), \delta^{\text{old}}). \tag{8}$$

If the hyperboxes from different classes overlap, a hyperbox contraction process is initiated to eliminate the overlapping regions. However, overlapping regions caused by hyperboxes from the same class are allowed, as shown in Fig. 4.

Contraction takes place when an overlapping region for the Δth dimension occurs. This case can be detected by the overlap test, i.e., when $\delta^{\text{old}} - \delta^{\text{new}} > 0$, then $\Delta = i$ and $\delta^{\text{old}} = \delta^{\text{new}}$. The overlap test proceeds to check the next

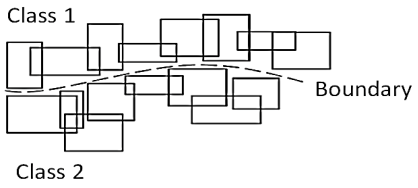

Fig. 4. Example of the FMM hyperbox boundary of a two-class problem.

dimension. If there are no more overlapping regions, the test stops, and the contraction step is indicated as not necessary, i.e., by setting $\Delta = -1$. During the contraction process, the hyperbox size is kept as large as possible by adjusting only one of the n dimensions in each overlapped hyperbox. In other words, the overlapping regions are eliminated by minimally adjusting each hyperbox. The same four cases are examined to determine a proper adjustment, as follows.

Case 1:

$$V_{j\Delta} < V_{k\Delta} < W_{j\Delta} < W_{k\Delta}, \quad W_{j\Delta}^{\text{new}} = V_{k\Delta}^{\text{new}} = \frac{W_{j\Delta}^{\text{old}} + V_{k\Delta}^{\text{old}}}{2}. \tag{9}$$

Case 2:

$$V_{k\Delta} < V_{j\Delta} < W_{k\Delta} < W_{j\Delta}, \quad W_{k\Delta}^{\text{new}} = V_{j\Delta}^{\text{new}} = \frac{W_{k\Delta}^{\text{old}} + V_{j\Delta}^{\text{old}}}{2}. \tag{10}$$

Case 3(a):

$$V_{j\Delta} < V_{k\Delta} < W_{k\Delta} < W_{j\Delta} \text{ and } (W_{k\Delta} - V_{j\Delta}) < (W_{j\Delta} - V_{k\Delta}), \quad V_{j\Delta}^{\text{new}} = W_{k\Delta}^{\text{old}}. \tag{11}$$

Case 3(b):

$$V_{j\Delta} < V_{k\Delta} < W_{k\Delta} < W_{j\Delta} \text{ and } (W_{k\Delta} - V_{j\Delta}) > (W_{j\Delta} - V_{k\Delta}), \quad W_{j\Delta}^{\text{new}} = V_{k\Delta}^{\text{old}}. \tag{12}$$

Case 4(a):

$$V_{k\Delta} < V_{j\Delta} < W_{j\Delta} < W_{k\Delta} \text{ and } (W_{k\Delta} - V_{j\Delta}) < (W_{j\Delta} - V_{k\Delta}), \quad W_{k\Delta}^{\text{new}} = V_{j\Delta}^{\text{old}}. \tag{13}$$

Case 4(b):

$$V_{k\Delta} < V_{j\Delta} < W_{j\Delta} < W_{k\Delta} \text{ and } (W_{k\Delta} - V_{j\Delta}) > (W_{j\Delta} - V_{k\Delta}), \quad V_{k\Delta}^{\text{new}} = W_{j\Delta}^{\text{old}}. \tag{14}$$

The learning process is online, and allows FMM to refine its classes incrementally over time. Online learning refers to the ability of a learning model to create new classes and refine the existing classes without affecting information captured in the model. This enables FMM to add new classes and refine the existing ones without the need for retraining. This is a key property to tackle the stability-plasticity dilemma [14], i.e., how a learning model is able to absorb new information continually (plastic) and, at the same time, prevent previously learned information from being corrupted (stable).

To solve the stability-plasticity dilemma, the ART family of neural networks [10], [11] was proposed. ART stems from

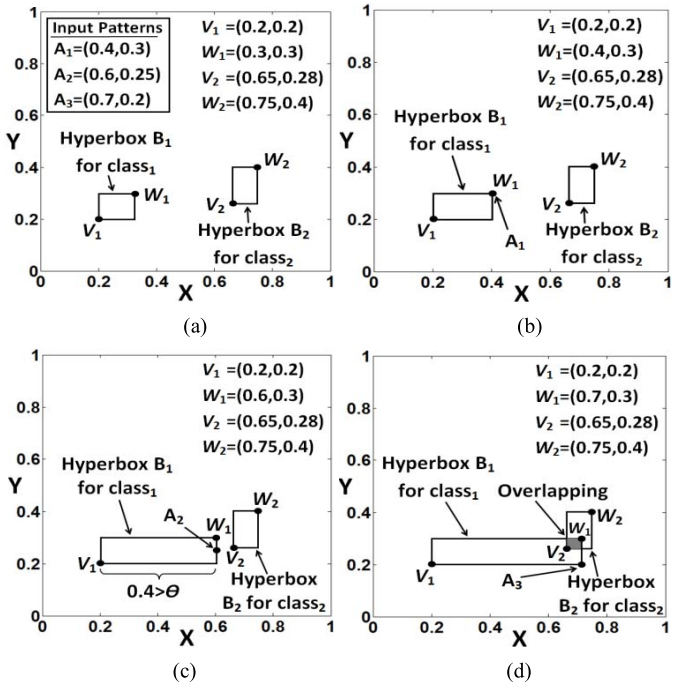

Fig. 5. FMM expansion process.

attempts to solve the instability learning issue in the instar–outstar network [13], [40]. This instability problem can be overcome by freezing the learned prototypes, i.e., reducing the learning rate gradually to zero. However, the network could lose its ability to absorb new information (plasticity). ART circumvents the stability-plasticity dilemma by adapting a learned prototype only when the input is sufficiently similar to the prototype [40]. Similar to ART, FMM share the advantage of resolving the stability-plasticity dilemma. Equations (3) and (4) are used in FMM to realize stable, online learning of data samples [40]. In Section V, how EFMM is able to learn stably through the training data using incremental learning and exhibit the direct access property of the class hyperbox for each trained data sample after one-pass learning is demonstrated.

C. Analysis of the FMM Learning Algorithm

While FMM is a useful network that supports online learning and has the capability of tackling the stability-plasticity dilemma, there are rooms to improve its learning algorithm, in order to avoid certain limitations in its learning process. We identify three key shortcomings that can compromise the FMM performance, as analyzed in the following sections.

1) Hyperbox Expansion: FMM applies the expansion process to include a new input pattern into one of the hyperboxes that belongs to the same class, provided that the constraint in (4) is not violated. Here, we show that the existing process can affect the performance of FMM negatively by increasing the overlapping region between different classes. FMM computes the sum of difference between the maximum and minimum points of all dimensions, and compares the resulting score with $(n\Theta)$. This can lead to incorrect predictions when some dimensions exceed the expansion coefficient while the overall score (i.e., sum of all dimensions) still resides within the limit of $(n\Theta)$.

The expansion problem can be explained using the example in Fig. 5. Consider two hyperboxes from two classes in a 2-D space, i.e., hyperbox $B_1 \in C_1$ (Class 1) with $V_1 = (0.2, 0.2)$ and $W_1 = (0.3, 0.3)$, and hyperbox $B_2 \in C_2$ (Class 2) with $V_2 = (0.65, 0.28)$ and $W_2 = (0.4, 0.75)$. Three input data of C_1, i.e., A_1, A_2, and $A_3 \in C_1$, are shown, as in Fig. 5(a). Parameter Θ is set to 0.3. When $A_1 = (0.4, 0.3)$, is applied, B_1 is selected as the winner according to (3). Because the constraint in (4) is satisfied

$$\sum_{i=1}^{n}(\text{Max}(W_{ji}, X_{hi}) - \text{Min}(V_{ji}, X_{hi})) = 0.3 \leq n\Theta$$

B_1 is expanded to include A_1, as in Fig. 5(b). As such, B_1 now has the same minimum point, $V_1 = (0.2, 0.2)$, but its maximum point is changed to $W_1 = (0.4, 0.3)$.

When $A_2 = (0.6, 0.25)$, is applied, B_1 is selected as the winner again, and is expanded to include A_2. In this case, while the sum of difference between the minimum and maximum points of all dimensions is smaller than $(n\Theta)$, (i.e., $0.5 \leq 0.6$), the difference in dimension X is actually larger than Θ. Since the constraint in (4) is still satisfied, B_1 is expanded, resulting in $V_1 = (0.2, 0.2)$ and $W_1 = (0.6, 0.3)$, as in Fig. 5(c). This creates a potential problem, whereby a hyperbox is allowed to expand even though at least one of the differences between the minimum and maximum points exceeds Θ. When A_3 is applied, B_1 is expanded to include A_3. The hyperbox structure of B_1 is expanded again, i.e., $V_1 = (0.2, 0.2)$ and $W_1 = (0.7, 0.3)$, as in Fig. 5(d). This results in an overlapping region between B_1 and B_2, as shown in Fig. 5(d). This problem can affect the FMM performance, as shown in Section V.

2) Hyperbox Overlap Test: In FMM, four cases are examined during the hyperbox overlap test to check whether there is an overlapping region between two hyperboxes from different classes. However, some overlapping regions cannot be detected by the existing four test cases. When this happens, FMM assumes an overlapped region as a nonoverlapped one, and stops the overlap test. Fig. 6 shows some overlapping cases that cannot be detected by FMM.

As in Fig. 6(a) and (b), when hyperboxes from different classes overlap with equal minimum or maximum points, FMM considers them as nonoverlapped hyperboxes based on inequalities (16)–(22), as explained in Section IV-B. Fig. 6(c) shows another undetected overlapping region, whereby a hyperbox with equal minimum and maximum points overlaps with another hyperbox that belongs to a different class. Fig. 6(d) shows a totally overlapped case, whereby one or more dimensions of the hyperboxes from different classes totally overlap each other (as in case study III, Section V).

3) Hyperbox Contraction: A hyperbox contraction process is deployed to eliminate the overlapping regions between hyperboxes from different classes in FMM. However, contraction is based on the four cases in the hyperbox overlap test, which can result in some undetected overlapping regions, as already shown in Fig. 6.

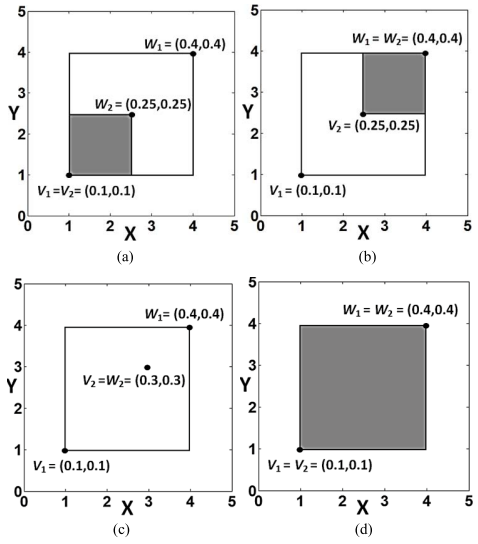

Fig. 6. Unrecognized overlapping cases.

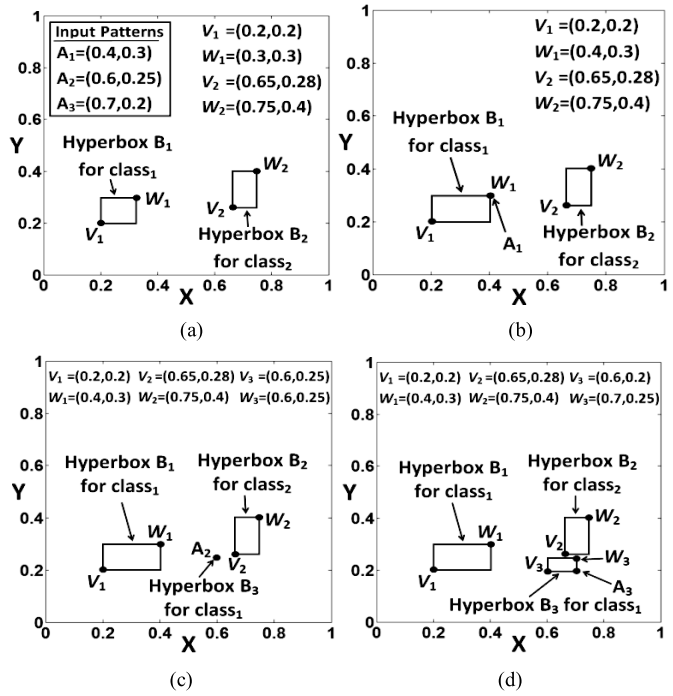

Fig. 7. Modified hyperbox expansion rule.

Based on the above analysis, we propose useful modifications to overcome the problems in the hyperbox expansion, overlap test, and contraction processes. These modifications lead to EFMM with the ability to improve the classification performance, as detailed in the following section.

IV. Enhanced FMM Network

In EFMM, three heuristic rules for hyperbox expansion, overlap test, and contraction are proposed, as follows.

A. Hyperbox Expansion Rule

As explained in Section III-A, the existing FMM expansion process can cause possible overlapping regions of hyperboxes from different classes in subsequent operations. To solve this problem, a new constraint is formulated, as follows:

$$\text{Max}_n(W_{ji}, a_{hi}) - \text{Min}_n(V_{ji}, a_{hi}) \leq \Theta. \quad (15)$$

Based on (15), each dimension of the jth hyperbox is checked separately to determine whether it exceeds the expansion coefficient (Θ). The expansion process is applied *if and only if* all hyperbox dimensions do not exceed Θ.

During the expansion process, FMM computes the sum of all dimensions [as in (4)] and compares the resulting score with ($n\Theta$). This can potentially lead to some overlapping regions between hyperboxes from different classes, as in Fig. 5. However, EFMM considers each dimension individually and checks the difference between the maximum and minimum points of each dimension against Θ separately. This overcomes the expansion problem explained in Section III-A.

The proposed rule can be further clarified by revisiting the example in Section III-A and Fig. 5. Fig. 7 shows the new scenario. When $A_1 = (0.4, 0.3)$ is applied, B_1 is selected as the winner. As the constraint in (15) for all dimensions is met, B_1 is expanded to include A_1, as depicted in Fig. 7(b). Now, B_1 now the same minimum point $V_1 = (0.2, 0.2)$, but its maximum point is changed to $W_1 = (0.4, 0.3)$. When A_2 is applied, B_1 is prohibited from expansion, because dimension X violates the constraint in (15). In this case, a new hyperbox is created, $B_3 \in C_1$, where $V_3 = W_3 = (0.6, 0.25)$, as in Fig. 7(c). When A_3 is applied, B_3 is expanded to include A_3, whereby $V_3 = (0.6, 0.2)$ and $W_3 = (0.7, 0.25)$, as in Fig. 7(d). It is clear that the proposed hyperbox expansion rule increases the number of hyperboxes (i.e., by creating B_3 in the example). However, adding B_3 is able to avoid the overlapping region, as shown in Fig. 7(d). The resulting network can also improve the classification performance, as shown in Section V.

B. Hyperbox Overlap Test Rule

As explained in Section III-B, using the current four cases (5)–(8) during the hyperbox overlap test is insufficient to identify all overlapping cases, as shown in Fig. 6. To tackle this problem, additional cases need to be included, to detect other possible overlapping regions. An overlapping region exists when one of the following nine cases is met. Note that (5) and (6) are two existing cases in FMM, while (16)–(22) are newly introduced cases.

Case 1 :

$$V_{ji} < V_{ki} < W_{ji} < W_{ki}, \quad \delta^{\text{new}} = \min(W_{ji} - V_{ki}, \delta^{\text{old}}). \quad (5)$$

Case 2:

$$V_{ki} < V_{ji} < W_{ki} < W_{ji}, \quad \delta^{\text{new}} = \min(W_{ki} - V_{ji}, \delta^{\text{old}}). \quad (6)$$

Case 3:

$$V_{ji} = V_{ki} < W_{ji} < W_{ki}$$
$$\delta^{\text{new}} = \min(\min(W_{ji} - V_{ki}, W_{ki} - V_{ji}), \delta^{\text{old}}). \quad (16)$$

Case 4:

$$V_{ji} < V_{ki} < W_{ji} = W_{ki}$$
$$\delta^{\text{new}} = \min(\min(W_{ji} - V_{ki}, W_{ki} - V_{ji}), \delta^{\text{old}}). \quad (17)$$

Case 5:

$$V_{ki} = V_{ji} < W_{ki} < W_{ji}$$
$$\delta^{\text{new}} = \min(\min(W_{ji} - V_{ki}, W_{ki} - V_{ji}), \delta^{\text{old}}). \quad (18)$$

Case 6:

$$V_{ki} < V_{ji} < W_{ki} = W_{ji}$$
$$\delta^{\text{new}} = \min(\min(W_{ji} - V_{ki}, W_{ki} - V_{ji}), \delta^{\text{old}}). \quad (19)$$

Case 7:

$$V_{ji} < V_{ki} \leq W_{ki} < W_{ji}$$
$$\delta^{\text{new}} = \min(\min(W_{ji} - V_{ki}, W_{ki} - V_{ji}), \delta^{\text{old}}). \quad (20)$$

Case 8:

$$V_{ki} < V_{ji} \leq W_{ji} < W_{ki}$$
$$\delta^{\text{new}} = \min(\min(W_{ji} - V_{ki}, W_{ki} - V_{ji}), \delta^{\text{old}}). \quad (21)$$

Case 9:

$$V_{ki} = V_{ji} < W_{ki} = W_{ji}, \quad \delta^{\text{new}} = \min(W_{ki} - V_{ji}, \delta^{\text{old}}). \quad (22)$$

Assuming that $\delta^{\text{old}} = 1$ initially, by conducting a dimension-by-dimension inspection, an overlapping region is detected when $\delta^{\text{old}} - \delta^{\text{new}} < 1$. Then, by setting $\Delta = i$ and $\delta^{\text{old}} = \delta^{\text{new}}$, the overlap test proceeds to check the next dimension. The test stops when no more overlapping regions are detected. In this case, $\delta^{\text{old}} - \delta^{\text{new}} = 1$.

C. Hyperbox Contraction Rule

The contraction rule is developed based on the nine cases [i.e., (5), (6), and (16)–(22)] of the hyperbox overlap test. Here, all cases are examined to determine a proper adjustment. Note that (9) and (10) are the two existing cases in FMM, while (23)–(32) are newly proposed cases.

Case 1:

$$V_{j\Delta} < V_{k\Delta} < W_{j\Delta} < W_{k\Delta}, \quad W_{j\Delta}^{\text{new}} = V_{k\Delta}^{\text{new}} = \frac{W_{j\Delta}^{\text{old}} + V_{k\Delta}^{\text{old}}}{2}. \quad (9)$$

Case 2:

$$V_{k\Delta} < V_{j\Delta} < W_{k\Delta} < W_{j\Delta}, \quad W_{k\Delta}^{\text{new}} = V_{j\Delta}^{\text{new}} = \frac{W_{k\Delta}^{\text{old}} + V_{j\Delta}^{\text{old}}}{2}. \quad (10)$$

Case 3:

$$V_{j\Delta} = V_{k\Delta} < W_{j\Delta} < W_{k\Delta}, \quad V_{k\Delta}^{\text{new}} = W_{j\Delta}^{\text{old}}. \quad (23)$$

Case 4:

$$V_{j\Delta} < V_{k\Delta} < W_{j\Delta} = W_{k\Delta}, \quad W_{j\Delta}^{\text{new}} = V_{k\Delta}^{\text{old}}. \quad (24)$$

Case 5:

$$V_{k\Delta} = V_{j\Delta} < W_{k\Delta} < W_{j\Delta}, \quad V_{j\Delta}^{\text{new}} = W_{k\Delta}^{\text{old}}. \quad (25)$$

Case 6:

$$V_{k\Delta} < V_{j\Delta} < W_{k\Delta} = W_{j\Delta}, \quad W_{k\Delta}^{\text{new}} = V_{j\Delta}^{\text{old}}. \quad (26)$$

TABLE II
DATA SETS OBTAINED FROM THE UCI MACHINE LEARNING REPOSITORY [42]

Data sets	Samples	Attributes	Classes
Sonar	208	60	2
Glass	214	9	6
Wine	178	13	3
Ionosphere	351	34	2
Iris	150	4	3
WBC (Origin)	699	9	2
Heart (Statlog)	270	13	2
Soybean (small)	47	35	4

Case 7(a):

$$V_{j\Delta} < V_{k\Delta} \leq W_{k\Delta} < W_{j\Delta} \text{ and } (W_{k\Delta} - V_{j\Delta}) < (W_{j\Delta} - V_{k\Delta}), \quad V_{j\Delta}^{\text{new}} = W_{k\Delta}^{\text{old}}. \quad (27)$$

Case 7(b):

$$V_{j\Delta} < V_{k\Delta} \leq W_{k\Delta} < W_{j\Delta} \text{ and } (W_{k\Delta} - V_{j\Delta}) > (W_{j\Delta} - V_{k\Delta}), \quad W_{j\Delta}^{\text{new}} = V_{k\Delta}^{\text{old}}. \quad (28)$$

Case 8(a):

$$V_{k\Delta} < V_{j\Delta} \leq W_{j\Delta} < W_{k\Delta} \text{ and } (W_{k\Delta} - V_{j\Delta}) < (W_{j\Delta} - V_{k\Delta}), \quad W_{k\Delta}^{\text{new}} = V_{j\Delta}^{\text{old}}. \quad (29)$$

Case 8(b):

$$V_{k\Delta} < V_{j\Delta} \leq W_{j\Delta} < W_{k\Delta} \text{ and } (W_{k\Delta} - V_{j\Delta}) > (W_{j\Delta} - V_{k\Delta}), \quad V_{k\Delta}^{\text{new}} = W_{j\Delta}^{\text{old}}. \quad (30)$$

Case 9(a):

$$V_{j\Delta} = V_{k\Delta} < W_{j\Delta} = W_{k\Delta}, \quad W_{j\Delta}^{\text{new}} = V_{k\Delta}^{\text{new}} = \frac{W_{j\Delta}^{\text{old}} + V_{k\Delta}^{\text{old}}}{2}. \quad (31)$$

Case 9(b):

$$V_{k\Delta} = V_{j\Delta} < W_{k\Delta} = W_{j\Delta}, \quad W_{k\Delta}^{\text{new}} = V_{j\Delta}^{\text{new}} = \frac{W_{k\Delta}^{\text{old}} + V_{j\Delta}^{\text{old}}}{2}. \quad (32)$$

When the maximum point (W_j) of one or more dimension that belongs to a hyperbox (i.e., H_j) is expanded and becomes totally overlapped with another hyperbox, (i.e., H_k), as shown in Fig. 6(d), EFMM uses case 9(a) to perform contraction. Likewise, case 9(b) is applied when the minimum point (V_j) of one or more dimension that belongs to H_j is expanded and becomes totally overlapped with H_k. Note that cases 9(a) and (b) are derived from cases (1) and (2) of original FMM.

V. PERFORMANCE EVALUATION

Four case studies were used to evaluate the effectiveness of EFMM using benchmark and real-world problems. In case study I, EFMM was compared with different variants of FMM using the Iris data set, with the results compared with those reported in the literature. The learning stability of EFMM was analyzed too. In case study II, FMM and EFMM performances were compared using four benchmark data sets. A real medical

diagnosis problem was used to further analyze the learning algorithms of EFMM and FMM in the case study III. Finally, EFMM was compared with different support vector machine (SVM) models using three benchmark data sets. In addition, EFMM was compared with Bayesian-based, decision tree-based, fuzzy-based, and neural-based classifiers using four benchmark data sets. Table II shows the statistics of all benchmark data sets used, which can be obtained from the UCI machine learning repository [42].

The bootstrap method [43] was used to quantify the EFMM results and indicate their stability. Bootstrap is a statistical method that does not depend on the assumption where the data samples must be drawn from a normal distribution. Using the principle of resampling, it is useful for quantifying performance indicators when a small data set is available. In this paper, 3000 resamplings were used to estimate the performance indicates, e.g., average accuracy rates and 95% confidence intervals, as well as standard deviations.

A. Case Study I

In this case study, the aim was to compare EFMM with other popular variants of FMM. In accordance with [27], [34], and [35], an evaluation with varying training set sizes using the Iris data set was conducted. By following the procedure in [35], the size of the training set was varied from 30% to 70%, while the total data samples were used for testing. Similarly, the expansion coefficient, Θ, was varied from 0.01 to 0.1, with a step of 0.01, and 10 runs were conducted for each Θ setting.

As in Table III, EFMM shows smaller misclassification rates when the training set sizes were between 40% and 70%, as compared with other FMM variants. For 30% training size, EFMM performed slightly worse than DCFMN. The lowest error rate of EFMM was recorded for 70% training size. By reducing the number of training samples, the misclassification rates increased for all FMM networks. Since the number of hyperboxes represent knowledge in FMM-based networks, reducing the training size results in minimizing the learned knowledge; therefore, increasing the classification errors.

To ascertain the EFMM learning stability, another study with the Iris data set was conducted. Learning stability is assessed with direct access of the respective class hyperboxes that encode the input patterns. As EFMM (and also FMM) creates hyperboxes incrementally, the order of input pattern presentation affects the hyperbox formation process. While data presentation order is not critical in offline learning models (since all data samples are learned iteratively), it is important in online learning models (e.g., FMM) as learning is conducted once in one-pass through all data samples.

In the experiment, we analyze the learning stability of EFMM by evaluating whether previously trained patterns are able to directly access their associated class hyperboxes accurately when they are presented again. If learning in EFMM is stable, it is expected to achieve 100% accuracy when the same order of training patterns is presented again.

Similar to the previous experiment, the training set size of the Iris data was varied from 30% to 70%. Training was conducted with one-pass learning. Then, the same training patterns were presented again for evaluation. The expansion coefficient, Θ, was set to 0.05 to create small hyperbox sizes and γ was set to 1. A total of 10 runs were conducted for each training set size. The average results of EFMM are shown in Table IV. EFMM exhibits perfect accuracy rates for all runs; therefore, indicating its learning stability and direct access capability to the target hyperboxes for the training patterns after one-pass learning.

To further evaluate the capability of the trained EFMM network, the remaining data samples (those not used for training) were employed for testing. The test errors are shown in Table IV. The trained EFMM network exhibits very high test accuracy rates (in averages), i.e., between 98% and 100%. In short, the results in Table IV show that EFMM is able to learn stably with the direct access property in one-pass learning and, at the same time, exhibit high generalization capability in classifying new test patterns.

TABLE III
COMPARISON BETWEEN EFMM AND OTHER FMM-RELATED MODELS USING THE IRIS DATA SET. THE RESULTS (PERCENTAGES OF MISCLASSIFICATION RATES) OF DCFMN, FMCN, FMM, AND GFMN ARE EXTRACTED FROM [35]

%	EFMM		
	Min	Max	Avg.
30	1.34	3.67	2.1
40	0.67	3.34	1.43
50	0	2	1
60	0	1.34	0.5
70	0	1.34	0.4
%	DCFMN		
	Min	Max	Avg.
30	0.67	4	1.99
40	0	2.67	1.49
50	0	2.67	1.09
60	0	2	0.8
70	0	2	0.51
%	FMCN		
	Min	Max	Avg.
30	0.67	4.67	2.43
40	0.67	4	1.87
50	0	2.67	1.39
60	0	2.67	1.01
70	0	2	0.73
%	FMM		
	Min	Max	Avg.
30	0.67	4.67	2.45
40	0.67	4	1.87
50	0	3.33	1.47
60	0	2.67	1.11
70	0	2	0.8
%	GFMN		
	Min	Max	Avg.
30	0.67	6	2.62
40	0	4	2.06
50	0	3.33	1.46
60	0	2.67	1.15
70	0	2.67	0.84

TABLE IV
EFMM LEARNING STABILITY AND DIRECT ACCESS EVALUATION

Training samples%	EFMM error samples for one epoch	
	Training error	Testing error
30	0	2
40	0	1
50	0	1
60	0	1
70	0	0

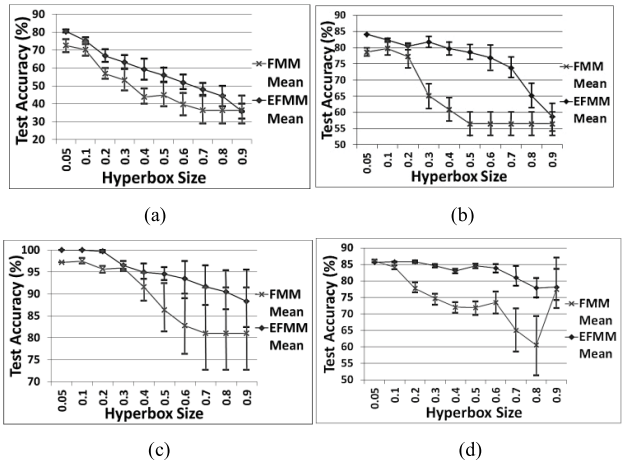

Fig. 8. Average (bootstrap) test accuracy rates of EFMM and FMM for different data sets. The error bars indicate the 95% confidence intervals. (a) Glass data set. (b) Sonar data set. (c) Wine data set. (d) Ionosphere data set.

TABLE V
COMPARISON BETWEEN FMM AND EFMM IN TERMS OF THE AVERAGE NUMBER OF HYPERBOXES

Models / Data Sets	FMM		EFMM	
	Θ =0.05	Θ =0.9	Θ =0.05	Θ =0.9
Sonar	104	2	126	11
Glass	73	6	110	10
Wine	106	3	107	6
Ionosphere	172	22	200	45

B. Case Study II

In this case study, the aim was to assess the effects of the expansion coefficient, Θ (hyperbox size), on the performances of FMM and EFMM using different data sets. Four data sets, i.e., sonar, glass, wine, and ionosphere, were used [42]. For all data sets, 60% of the data samples (randomly selected) were used for learning and 40% for testing. A series of systematic evaluations was conducted by increasing Θ from 0.05 to 0.9, and each test was repeated 10 times (i.e., 10 times for each Θ setting). Fig. 8(a)–(d) shows the average test accuracy rates computed using the bootstrap method [43]. The error bars indicate the associated 95% confidence intervals computed with the significance level (α) set at 0.05.

As can be observed in Fig. 8, by incorporating the proposed modifications, EFMM is able to improve the performance of FMM. It can also be observed in Fig. 8 that the performances of FMM and EFMM are affected by Θ. In terms of network complexity (i.e., the number of hyperboxes created), FMM is less complex than EFMM, as shown in Table V. As explained previously, FMM tends to create a fewer hyperboxes at the expense of keeping some overlapping regions that belong to different classes in its network structure (more details are presented in case study III). However, the increase in EFMM network complexity leads to improved results.

In both the networks, increasing Θ reduced the number of hyperboxes. At higher settings of Θ, the test accuracy rates of both FMM and EFMM decreased, since the number of hyperboxes decreased, i.e., knowledge coded in the network became sparse. However, one discrepancy is shown in Fig. 8(d) with the ionosphere data set. A closer examination revealed that when $\Theta = 0.9$, 21 out of 22 FMM hyperboxes belonged to the good class, with only one hyperbox belonged to the bad class. This helped FMM to improve its accuracy rate by biasing the predictions to the good class, which contained most of the data samples, i.e., 225 samples (or 64%) of the data samples belonged to the good class.

In general, it is obvious that EFMM is able to produce better performances than FMM. In other words, the proposed hyperbox expansion, overlap test, and contraction rules in EFMM are effective in reducing the misclassification rates.

TABLE VI
OVERALL RESULTS OF FMM AND EFMM FOR THE ACS DATA SET

Hyperbox Size	EFMM			
	Min	Mean	Max	Avg. Hyperbox No.
0.4	74.16	75.84	77.50	94
0.5	80.00	82.60	85.83	38
0.6	82.50	85.52	87.91	23
Hyperbox Size	FMM			
	Min	Mean	Max	Avg. Hyperbox No.
0.4	57.08	62.91	69.58	68
0.5	57.50	63.83	70.00	51
0.6	72.08	78.55	87.08	15

C. Case Study III

In this case study, EFMM was used to tackle a real medical diagnosis problem. This problem revealed the limitations of FMM in processing ternary-valued data. A data set with real acute coronary syndrome (ACS) patient records from a hospital was used. The data set contained 118 samples (patient records). After consultation with medical experts, a total of 16 features comprising physical symptoms (e.g., sweating and chest pain), background information (hypertension and smoking), and EEG signal representations, from each patient record were extracted. All features, except one, were ternary valued, where 1 and 0.5 represented presence and absence of each symptom while 0 represented missing value. The only continuous feature (i.e., duration of pain) was normalized between 0 and 1. Given the patient information, the task was to provide a prediction of the diagnosis, i.e., whether the patient was suffering from ACS or otherwise. To evaluate the performance of EFMM and FMM, 80% of data samples were used for training with the remaining (20%) used for testing. Since the ACS data samples contained only three types of values, i.e., 0, 0.5, and 1, three levels of $\Theta (0.0 < \Theta < 0.5, \Theta = 0.5, 0.5 < \Theta < 1.0)$ were of interest. As such, a series of experimental runs with Θ set to 0.4, 0.5, and 0.6 was conducted. Table VI shows the overall results.

Both EFMM and FMM results were affected by two factors: Θ and the nature of ACS data. When $\Theta = 0.4$, the EFMM test accuracy was about 76% with 94 hyperboxes. The number of hyperboxes reduced to 38 when $\Theta = 0.5$, while its test accuracy rate increased to about 83%. When $\Theta = 0.6$, EFMM generated 23 hyperboxes with about 86% accuracy. The result implied that the data set was noisy, as reducing Θ resulted in increasing accuracy (i.e., with less noisy hyperboxes).

The performances of FMM were inferior to those of EFMM. However, the FMM hyperbox numbers were smaller too.

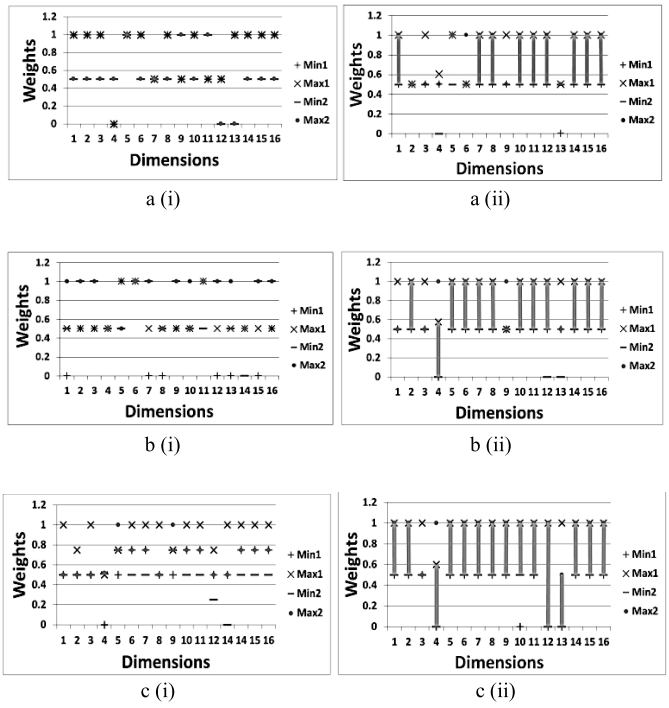

Fig. 9. Comparison between FMM and EFMM hyperbox structures for (a) $\Theta = 0.4$, (b) $\Theta = 0.5$, and (c) $\Theta = 0.6$. Black and red symbols represent hyperboxes of classes 1 and 2, respectively). Overlapping regions (gray bars) on individual dimensions occurs in FMM hyperboxes, but not those in EFMM. (a.i) EFMM hyperboxes. (a.ii) FMM hyperboxes. (b.i) EFMM hyperboxes. (b.ii) FMM hyperboxes. (c.i) EFMM hyperboxes. (c.ii) FMM hyperboxes.

However, a close look at the FMM hyperbox structures revealed that overlapping regions occurred frequently in individual dimensions of the hyperboxes from different classes. This scenario was avoided in the EFMM hyperbox structures. Fig. 9 shows a comparison between two hyperboxes, one from FMM and another from EFMM, on a dimension-by-dimension basis. From Fig. 9(a.ii), (b.ii), and (c.ii), it is clear that overlapping regions (indicated by bars) on individual dimensions occurred frequently in the hyperbox structures of FMM. As an example, when $\Theta = 0.4$, a total of 9 (out of 16) dimensions exhibited overlapping regions (i.e., dimensions 1, 7, 8, 10–12, and 14–16), owing to the expansion process of FMM. When the hyperbox size increased, the number of overlapped dimensions increased too. As shown in Fig. 9(b.ii) and (c.ii), the numbers of overlapped dimensions were 12 (2, 4 to 8, 10–12, and 14–16) when $\Theta = 0.5$, and 15 (all except 3) when $\Theta = 0.6$, respectively. All these overlapping dimensions did not appear at all in the EFMM hyperbox structures, as shown in Fig. 9(a.i), (b.i), and (c.i).

In summary, this ACS problem highlights the limitations of FMM in processing ternary-valued data samples.

1) The expansion process could lead to overlapping dimensions of hyperboxes from different classes.
2) The number of overlapping dimensions increases when the hyperbox size increases.

Therefore, the FMM network complexity (i.e., the number of hyperboxes) reduces corresponding to increasing Θ, but with increasing number of overlapping dimensions in the hyperbox structures. These problems can be overcome with the proposed heuristic rules in EFMM.

TABLE VII
COMPARISON BETWEEN EFMM AND SVM CLASSIFIERS IN [41]

Methods	SVM classifiers	WBC (Origin)	Heart (Statlog)	Soybean (small)
Gaussian RBF kernel	MAdaBoostSVM	95.54	78.30	36.67
	SingleSVM	95.26	79	36.67
	BaggingSVM	96.57	83.48	38.89
	Arc-x4SVM	95.35	78.56	36.67
	AdaBoostSVM	95.21	78.41	37.11
Linear kernel	MAdaBoostSVM	96.72	83.19	91.78
	SingleSVM	96.69	83.37	99.56
	BaggingSVM	96.78	83.19	99.56
	Arc-x4SVM	95.99	79.22	99.56
	AdaBoostSVM	96.65	82.11	99.56
Polynomial kernel	MAdaBoostSVM	95.31	77.70	93.11
	SingleSVM	94.12	75	100
	BaggingSVM	94.83	78.48	100
	Arc-x4SVM	94	76.26	100
	AdaBoostSVM	94.37	76.89	100
Average	*Gaussian RBF kernel*	*95.58*	*79.55*	*37.20*
	Linear kernel	*96.56*	*82.21*	*98*
	Polynomial kernel	*94.52*	*76.86*	*98.62*
Bootstrap (mean value)	EFMM	96.61	81.1	100

D. Case Study IV

In the previous three case studies, the performances of EFMM and FMM-related models were compared. To further evaluate the effectiveness of EFMM as compared with other classifiers, two experiments were conducted, with the results compared with those reported in the literature. In the first experiment, the SVM classifier (one of the most popular classifiers) with three different kernel functions (Gaussian RBF, polynomial, and linear) was employed [44]. Three UCI benchmark data sets were used, i.e., Wisconsin breast cancer (WBC), Statlog (heart), and Soybean (small). The experimental procedure followed that in [44], to have a fair comparison with the results reported therein. As such, the tenfold cross-validation method was used for the WBC and heart problems, where each test was repeated 100 times (10 times for each fold), while for the Soybean problem, the fivefold cross-validation method was used (each repeated 50 times). Note that for the WBC data set, 16 data samples that contained missing values were omitted [44]. The average results of EFMM were obtained using the bootstrap method. Table VII shows the overall results.

As shown in Table VII, the performances of SVM classifiers were affected by their kernel functions (Gaussian, polynomial, and linear). For the WBC and heart problems, BaggingSVM with linear and Gaussian kernels produced the best results, respectively. For the Soybean problem, four SVM classifiers (SingleSVM, BaggingSVM, Arc-x4SVM, and AdaBoostSVM) with polynomial kernel produced perfect classification accuracy. EFMM also achieved the same perfect result. Note that gray cells in Table VII indicate results that are better than or equal to those of EFMM. It can be seen that SVM with linear kernel (except Arc-x4SVM) outperformed EFMM for the WBC and heart problems. However, this was not the case

TABLE VIII
COMPARISON BETWEEN EFMM AND VARIOUS CLASSIFIERS IN [45]

Methods \ Datasets	Iris	WBC	Wine	Glass
Naive Bayes	96.00±0.30	95.90±0.20	96.75±2.32	42.90±1.70
C4.5	95.13±0.20	94.71±0.09	91.14±5.12	67.90±0.50
SMO	96.69±2.58	97.51±0.97	97.87±2.11	58.85±6.58
Fuzzy gain measure	96.88±2.40	98.14±0.90	98.36±1.26	69.14±4.69
HHONC	97.46±2.31	97.17±1.17	97.88±2.29	56.50±7.58
EFMM	99.14±1.70	98.27±0.01	99.94±0.04	77.53±4.46

for the Soybean problem. On the other hand, EFMM outperformed SVM with Gaussian RBF and polynomial kernels for the WBC and heart problems, except for BaggingSVM with Gaussian kernel in the heart problem. For the Soybean problem, EFMM outperformed all types of SVM classifiers with Gaussian and linear kernels, and achieved the same performance as compared with SVM classifiers with polynomial kernel (except MAdaBoostSVM).

We also computed the average results by combining all SVM classifiers for each kernel function, as shown Table VII (row average). The outcome revealed that all EFMM results were better than those from SVM with different kernel functions, except one, i.e., slightly more than 1% lower than the average result of SVM classifiers with linear kernel for the heart problem. In short, EFMM was able to demonstrate good and stable results that were comparable with various types of SVM classifiers in this experiment.

In the second experiment, four benchmark problems (i.e., iris, WBC, wine, and glass) were used. The experimental procedure in [45] was followed, to have a fair comparison between EFMM and other classifiers in [45]. As such, the train-test method was adopted, where 25% of data samples were used for testing with the remaining for training. The results (mean accuracy and standard deviation) were computed from 200 runs, as shown in Table VIII.

As shown in Table VIII, the EFMM was able to yield better test accuracy rates as compared with those from Naive Bayes, C4.5, sequential minimal optimization, fuzzy gain measure, and hybrid higher order neural classifier. This outcome indicated that EFMM was able to outperform Bayesian-based, decision tree-based, fuzzy-based, and neural-based classifiers as reported in [45].

VI. CONCLUSION

In this paper, an EFMM network that is able to yield a high classification performance while maintaining the salient features of FMM is proposed. FMM is a useful online learning model that is able to overcome the stability-plasticity dilemma. Nevertheless, we identify a number of existing limitations pertaining to the current FMM learning dynamics. First, it is possible for the current FMM hyperbox expansion process to increase the overlapping regions between different classes. Second, the existing hyperbox overlap test rule is insufficient to detect all overlapping regions of hyperboxes. Third, the hyperbox contraction process is affected since it is possible for undetected overlapping regions to exist after the hyperbox overlap test is performed. These shortcomings form the main motivations of this paper. We focus on analyzing the efficacy of FMM in dealing with overlapping regions of hyperboxes that belong to different classes. Therefore, the EFMM network is proposed with the aim to resolve the limitations of FMM and improve its classification ability.

The main contributions of this paper are the three heuristic rules introduced in EFMM to enhance its learning algorithm. First, a new hyperbox expansion rule to reduce the FMM classification errors by minimizing the overlapping regions of hyperboxes that belong to different classes during the expansion process is proposed. Second, the existing hyperbox overlap test is extended so that all overlapping regions from hyperboxes that belong to different classes can be identified. Third, a new hyperbox contraction rule to solve different overlapping cases that are not covered by the existing hyperbox contraction process is derived. Based on the three heuristic rules, we have analyzed the learning dynamics of EFMM comprehensively. Specifically, the efficacy of EFMM has been evaluated with eight benchmark data sets (as in Table II) with different settings as well as a medical diagnosis task with real patient records. The EFMM results are compared with those from FMM variants, SVMs, Bayesian-based, decision tree-based, fuzzy-based, and neural-based classifiers. The outcomes positively ascertain the usefulness of the proposed heuristic rules in enhancing the classification performance of EFMM, and enable EFMM to tackle pattern classification tasks competently in comparison with other classifiers in the literature.

It should be noted that the network complexity of EFMM is higher than that of FMM, as a side effect of the heuristic rules. For further work, it is important to reduce the EFMM network complexity and, at the same time, maintain its performance. One way is to prune EFMM hyperboxes that have low confidence factors. Besides that, it is useful to merge EFMM hyperboxes based on some cluster validity measure. On the other hand, applicability of EFMM to other classification tasks can be examined. Different decision combination methods can also be used to form an ensemble of EFMM networks to further improve its robustness in tackling complex real-world pattern classification problems.

REFERENCES

[1] D. Graupe, *Principles of Artificial Neural Networks*. Singapore: World Scientific, 1997.
[2] P. J. G. Lisboa, "A review of evidence of health benefit from artificial neural networks in medical intervention," *Neural Netw.*, vol. 15, no. 1, pp. 11–39, 2002.
[3] M. Wu and P. Rastgoufard, "Optimum decision by artificial neural networks for reactive power control equipment to enhance power system stability and security performance," in *Proc. IEEE Power Eng. Soc. General Meeting*, vol. 2. Jun. 2004, pp. 2120–2125.
[4] M.-Y. Chow and S. O. Yee, "Methodology for on-line incipient fault detection in single-phase squirrel-cage induction motors using artificial neural networks," *IEEE Trans. Energy Convers.*, vol. 6, no. 3, pp. 536–545, Sep. 1991.
[5] G. P. Zhang, "Neural networks for classification: A survey," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 30, no. 4, pp. 451–462, Nov. 2000.
[6] A. Quteishat, C. P. Lim, J. Tweedale, and L. C. Jain, "A neural network-based multi-agent classifier system," *Neurocomputing*, vol. 72, no. 7, pp. 1639–1647, Mar. 2009.

[7] M. McCloskey and N. J. Cohen, "Catastrophic interference in connectionist networks: The sequential learning problem," in *The Psychology of Learning and Motivation*, G. H. Bower, Ed. New York, NY, USA: Academic, 1989, pp. 109–165.
[8] R. Ratcliff, "Connectionist models of recognition memory: Constraints imposed by learning and forgetting functions," *Psychol. Rev.*, vol. 97, no. 2, pp. 285–308, 1990.
[9] R. Polikar, L. Upda, S. S. Upda, and V. Honavar, "Learn++: An incremental learning algorithm for supervised neural networks," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 31, no. 4, pp. 497–508, Nov. 2001.
[10] S. Grossberg, "Adaptive pattern classification and universal recording: I. Parallel development and coding of neural feature detectors," *Biol. Cybern.*, vol. 23, no. 3, pp. 121–134, Jul. 1976.
[11] S. Grossberg, "Adaptive pattern classification and universal recording: II. Feedback, expectation, olfaction, illusions," *Biol. Cybern.*, vol. 23, no. 4, pp. 187–202, Dec. 1976.
[12] P. K. Simpson, "Fuzzy min-max neural networks. I. Classification," *IEEE Trans. Neural Netw.*, vol. 3, no. 5, pp. 776–786, Sep. 1992.
[13] G. A. Carpenter and S. Grossberg, "A massively parallel architecture for a self-organizing neural pattern recognition machine," *Comput. Vis., Graph., Image Process.*, vol. 37, no. 1, pp. 54–115, Jan. 1987.
[14] S. Grossberg, "How does a brain build a cognitive code?" *Psychol. Rev.*, vol. 87, no. 1, pp. 1–51, Jan. 1980.
[15] P. K. Simpson, "Fuzzy min-max neural networks—Part 2: Clustering," *IEEE Trans. Fuzzy Syst.*, vol. 1, no. 1, pp. 32–45, Feb. 1993.
[16] L. A. Zadeh, "Fuzzy sets," *Inf. Control*, vol. 8, no. 3, pp. 338–353, Jun. 1965.
[17] S. Yilmaz and Y. Oysal, "Fuzzy wavelet neural network models for prediction and identification of dynamical systems," *IEEE Trans. Neural Netw.*, vol. 21, no. 10, pp. 1599–1609, Oct. 2010.
[18] J. E. Moreno, O. Castillo, J. R. Castro, L. G. Martínez, and P. Melin, "Data mining for extraction of fuzzy IF-THEN rules using Mamdani and Takagi-Sugeno-Kang FIS," *Eng. Lett.*, vol. 15, no. 1, pp. 82–88, 2007.
[19] D. Hidalgo, O. Castillo, and P. Melin, "Type-1 and type-2 fuzzy inference systems as integration methods in modular neural networks for multimodal biometry and its optimization with genetic algorithms," *Inf. Sci.*, vol. 179, no. 13, pp. 2123–2145, Jun. 2009.
[20] D. Sanchez, P. Melin, O. Castillo, and F. Valdez, "Modular granular neural networks optimization with multi-objective hierarchical genetic algorithm for human recognition based on iris biometric," in *Proc. IEEE CEC*, Jun. 2013, pp. 772–778.
[21] P. Melin, O. Mendoza, and O. Castillo, "Face recognition with an improved interval type-2 fuzzy logic Sugeno integral and modular neural networks," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 41, no. 5, pp. 1001–1012, Sep. 2011.
[22] C.-F. Juang, T.-C. Chen, and W.-Y. Cheng, "Speedup of implementing fuzzy neural networks with high-dimensional inputs through parallel processing on graphic processing units," *IEEE Trans. Fuzzy Syst.*, vol. 19, no. 4, pp. 717–728, Aug. 2011.
[23] K. S. Yap, C. P. Lim, and M. T. Au, "Improved GART neural network model for pattern classification and rule extraction with application to power systems," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 2310–2323, Dec. 2011.
[24] M. Davanipoor, M. Zekri, and F. Sheikholeslam, "Fuzzy wavelet neural network with an accelerated hybrid learning algorithm," *IEEE Trans. Fuzzy Syst.*, vol. 20, no. 3, pp. 463–470, Jun. 2012.
[25] M. Pratama, S. G. Anavatti, and E. Lughofer, "GENFIS: Towards an effective localist network," *IEEE Trans. Fuzzy Syst.*, May 2013, doi: 10.1109/TFUZZ.2013.2264938.
[26] Y.-Y. Lin, J.-Y. Chang, and C.-T. Lin, "Identification and prediction of dynamic systems using an interactively recurrent self-evolving fuzzy neural network," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 2, pp. 310–321, Feb. 2013.
[27] B. Gabrys and A. Bargiela, "General fuzzy min-max neural network for clustering and classification," *IEEE Trans. Neural Netw.*, vol. 11, no. 3, pp. 769–783, May 2000.
[28] A. V. Nandedkar and P. K. Biswas, "A general reflex fuzzy min-max neural network," *Eng. Lett.*, vol. 14, no. 1, pp. 195–205, Feb. 2007.
[29] A. Likas, "Reinforcement learning using the stochastic fuzzy min-max neural network," *Neural Process. Lett.*, vol. 13, no. 3, pp. 213–220, 2001.
[30] A. Likas and K. Blekas, "A reinforcement learning approach based on the fuzzy min-max neural network," *Neural Process. Lett.*, vol. 4, no. 3, pp. 167–172, 1996.
[31] A. Rizzi, M. Panella, and F. M. F. Mascioli, "Adaptive resolution min-max classifiers," *IEEE Trans. Neural Netw.*, vol. 13, no. 2, pp. 402–414, Mar. 2002.
[32] A. Bargiela, W. Pedrycz, and M. Tanaka, "An inclusion/exclusion fuzzy hyperbox classifier," *Int. J. Knowl.-Based Intell. Eng. Syst.*, vol. 8, no. 2, pp. 91–98, Aug. 2004.
[33] H. J. Kim and H. S. Yang, "A weighted fuzzy min-max neural network and its application to feature analysis," in *Advances in Natural Computation* (Lecture Notes in Computer Science), vol. 3612, L. Wang, K. Chen, and Y. Ong, Eds. Berlin, Germany: Springer-Verlag, 2005, pp. 1178–1181.
[34] A. V. Nandedkar and P. K. Biswas, "A fuzzy min-max neural network classifier with compensatory neuron architecture," *IEEE Trans. Neural Netw.*, vol. 18, no. 1, pp. 42–54, Jan. 2007.
[35] H. Zhang, J. Liu, D. Ma, and Z. Wang, "Data-core-based fuzzy min-max neural network for pattern classification," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 2339–2352, Dec. 2011.
[36] A. Quteishat and C. P. Lim, "A modified fuzzy min-max neural network with rule extraction and its application to fault detection and classification," *Appl. Soft Comput.*, vol. 8, no. 2, pp. 985–995, Mar. 2008.
[37] A. Quteishat, C. P. Lim, and K. S. Tan, "A modified fuzzy min-max neural network with a genetic-algorithm-based rule extractor for pattern classification," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 40, no. 3, pp. 641–650, May 2010.
[38] M. Seera, C. P. Lim, D. Ishak, and H. Singh, "Fault detection and diagnosis of induction motors using motor current signature analysis and a hybrid FMM-CART model," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 1, pp. 97–108, Jan. 2012.
[39] M. Seera and C. P. Lim, "Online motor fault detection and diagnosis using a hybrid FMM-CART model," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 4, pp. 1–7, Apr. 2014.
[40] C.-T. Lin and C. S. G. Lee, *Neural Fuzzy Systems: A Neuro-Fuzzy Synergism to Intelligent Systems*. New York, NY, USA: Prentice-Hall, 1996.
[41] T. Kohonen, *Self-Organization and Associative Memory*. Berlin, Germany: Springer-Verlag, 1988.
[42] K. Bache and M. Lichman. (2013). *UCI Machine Learning Repository*. School Inf. Comput. Sci., Univ. California, Irvine, CA, USA [Online]. Available: http://archive.ics.uci.edu/ml
[43] B. Efron, "Bootstrap methods: Another look at the jackknife," *Ann. Statist.*, vol. 7, no. 1, pp. 1–26, 1979.
[44] S.-J. Wang, A. Mathew, Y. Chen, L.-F. Xi, L. Ma, and J. Lee, "Empirical analysis of support vector machine ensemble classifiers," *Expert Syst. Appl.*, vol. 36, no. 3, pp. 6466–6476, Apr. 2009.
[45] M. Fallahnezhad, M. H. Moradi, and S. Zaferanlouei, "A hybrid higher order neural classifier for handling classification problems," *Expert Syst. Appl.*, vol. 38, no. 1, pp. 386–393, Jan. 2011.

Mohammed Falah Mohammed received the B.Eng. degree in computer engineering from the Technical College of Mosul, Mosul, Iraq, and the M.Sc. degree in wireless and mobile system and the Ph.D. degree in computational intelligence from the School of Electrical and Electronic Engineering, University of Science Malaysia, NibongTebal, Malaysia, in 2006, 2010, and 2014, respectively.

His current research interests include neural networks, multiagent systems, pattern classification, and computer languages.

Chee Peng Lim received the B.E.E. (Hons.) degree from the University of Technology, Kuala Lumpur, Malaysia, and the M.Sc. (Hons.) degree in control systems engineering and the Ph.D. degree from the University of Sheffield, Sheffield, U.K., in 1992, 1993, and 1997, respectively.

He is currently an Associate Professor with the Centre for Intelligent Systems Research, Deakin University, Geelong, VIC, Australia. His current research interests include computational intelligence, pattern classification, optimization, decision support systems, medical prognosis and diagnosis, and fault detection and diagnosis.