# Learning-Regulated Context Relevant Topographical Map

Pitoyo Hartono, *Member, IEEE*, Paul Hollensen, and Thomas Trappenberg, *Member, IEEE*

***Abstract*— Kohonen's self-organizing map (SOM) is used to map high-dimensional data into a low-dimensional representation (typically a 2-D or 3-D space) while preserving their topological characteristics. A major reason for its application is to be able to visualize data while preserving their relation in the high-dimensional input data space as much as possible. Here, we are seeking to go further by incorporating semantic meaning in the low-dimensional representation. In a conventional SOM, the semantic context of the data, such as class labels, does not have any influence on the formation of the map. As an abstraction of neural function, the SOM models bottom-up self-organization but not feedback modulation which is also ubiquitous in the brain. In this paper, we demonstrate a hierarchical neural network, which learns a topographical map that also reflects the semantic context of the data. Our method combines unsupervised, bottom-up topographical map formation with top-down supervised learning. We discuss the mathematical properties of the proposed hierarchical neural network and demonstrate its abilities with empirical experiments.**

***Index Terms*— Context relevance, radial basis function (RBF) network, self-organizing map (SOM), supervised learning, topographical maps.**

## I. Introduction

THE expanding number of high-dimensional data sets makes dimensionality reduction techniques increasingly important. One of the most powerful data reduction and visualization methods is Kohonen's self-organizing map (SOM) [1], [2], which is able to map high-dimensional data into a low-dimensional map while preserving some of the data's topological characteristics. In addition to the biological importance [3], [4], the reason for such low-dimensional embedding in technical applications is often to visualize the data for humans by showing their semantic relations. For example, in text mining, we might want to find semantically related articles in a visually driven interface [5]. While closeness—such as measured by Euclidean distance in feature space—has often some striking correlation to semantic relations, this is certainly not always the case. Our long-term vision is to build models that resemble brain organization in which a hierarchy of maps, which are often topographic (at least close to the sensory input), build up semantic representations on different levels of abstraction. In this paper, we discuss a new method that incorporates top-down semantic context into the bottom-up process of embedding high-dimensional data. We name the resulting map the context-relevant SOM (CRSOM). Our map is unique, because it is organized as the internal layer of a hierarchical supervised neural network that we called restricted radial basis function (rRBF) network. The map is organized in the supervised training process of the neural network during which a bottom-up training signal to preserve the topographical characteristics of the data is regulated by a top-down regulatory signal to ensure the relevance of the data's context. Interestingly, our derivation of the method shows that the top-down regulatory signal provides a repelling force to separate similar data with different contexts. This has interesting implications on the map formation process.

The proposed CRSOM can be utilized as an alternative to the conventional SOM. In this paper, we mainly focus on the utilization of the CRSOM in classification problems. The basic method was first proposed in [6] and based on our empirical experiments that indicate strong correlation between the topographical representation and the learning abilities of hierarchical neural networks [3], [7]. Here, we expand on the formalization of the method by specifically discussing the related energy function and by providing a more thorough study of its behavior. We also propose a new measure with which we can quantify the semantic separability of the data. We show that CRSOM has a better representation of class context, where data belonging to different classes are separated with wider margin, while data belonging to the same class are clustered closely to each other. The later characteristic leads to sparser code, which also seems to be fundamental in the brain [8], whereas SOM visualizes the structure of the data alone, CRSOM visualizes the structure of the problem. This kind of visualization may give us information not only on the distribution of the data, but also on the complexity of their classification. Importantly, as rRBF is trained to minimize a well-defined energy function, CRSOM is an optimal internal representation.

## II. Related Topics

Here, some self-organization and dimensionality reduction mechanisms that share some similarities with the proposed rRBF and CRSOM models are reviewed.

Manuscript received November 17, 2013; revised August 6, 2014 and October 6, 2014; accepted November 26, 2014. This work was supported by the Japan Society for the Promotion of Science through the Grants-in-Aid for Scientific Research under Grant 25330419 and Grant 26590182. The work of P. Hollensen and T. Trappenberg was supported by the Natural Sciences and Engineering Research Council of Canada.

P. Hartono is with the Department of Electrical and Electronics Engineering, Chukyo University, Nagoya 466-8666, Japan (e-mail: hartono@ieee.org).

P. Hollensen and T. Trappenberg are with the Faculty of Computer Science, Dalhousie University, Halifax, NS B3H 4R2, Canada (e-mail: hollensen@cs.dal.ca; tt@cs.dal.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNNLS.2014.2379275

The CRSOM shares a similar objective with the self-organizing semantic maps [9] that is to extend the principle of the conventional SOM to include semantical expression in visualizing the underlying properties of high-dimensional data. The objective is important for higher level information processing where the relationship between data points is less obvious from their intrinsic features only. The primary novelty of the proposed rRBF is that in contrast to the self-organizing semantic map that treats the semantical expression as an extension for the data's intrinsic features; here, the semantical expression is the output of a higher layer of a hierarchical network that utilizes the map as its internal representation. In the rRBF, a semantical expression is more correctly treated as the abstraction of a data point that should not be mixed with the intrinsic features. Consequently, the map formation process in rRBF is also significantly different from that of SOM.

The rRBF network's structure is similar to the counterpropagation network (CPN) [10], [11]. However, the CPN executes a two-phase training procedure: 1) the hidden layer is self-organized using the input data independent of their contexts and 2) the self-organizing process of the same layer using the teacher signals. While the idea of CPN for connecting input and output based on their strongest correlation is intuitive, there is no explicit energy function to optimize. The proposed rRBF is different, in that the map is generated as an implication of a learning process that minimizes the context error in the output layer; hence, the map is an optimal representation of the learned context.

Similar to the proposed CRSOM, in the past, various methods have been proposed to map high-dimensional data into a low-dimensional space by optimizing a well-defined energy function while preserving the data's local characteristics. For example, the linear locally embedding dimensionality reduction method [12] preserves the linear relationship of neighboring data points. Stochastic neighbor embedding (SNE) [13] and its variant t-SNE [14] are elegant dimensionality reduction algorithms that map high-dimensional data into low-dimensional maps while preserving the conditional probability of their neighbor relation. These algorithms are to some extent inspired by multidimensional scaling [15] and the Sammon algorithm [16], which are dimensionality reduction mechanisms that reflect the distance metric of the given data into a low-dimensional map. While most of the methods utilize Euclidean distance as a similarity measure, an interesting map in [17] utilized geodesic or manifold distance. When these methods are used to find low-dimensional visualizations of semantic relations, it is implicitly assumed that semantically related entities are close in the high-dimensional feature space. However, this assumption often does not hold, and the proposed model is hence more general than the topology-preserving algorithms. In addition, in contrast to the proposed method, it is not possible in most of the above methods to map a new input, which is unobserved in the learning process, into the generated maps.

Conventionally, SOM is a topology preserving mechanism that is trained in an unsupervised manner. Several proposals have been made to add supervised training procedures to the standard SOM. The hypermap [18] was one of the earliest attempts to combine SOM with a supervised training mechanism, with the main objective to utilize SOM to visualize high-dimensional data and at the same time to classify unlabeled data. The hypermap takes labeled training data and executes a two-phase learning procedure, where in the first phase, the data are self-organized in an unsupervised manner as in the conventional SOM, while in the second-phase, the map is reorganized in a supervised manner. The supervised phase is executed according to the training mechanism of learning vector quantization [2], where a reference vector sharing the same label as the input is modified toward the input, while a reference vector having different labels is repelled from the input. While the two-phase training mechanism of the hypermap is intuitive, the generated map is not optimal because there is no explicit energy function that is optimized with this heuristic. The proposed rRBF also exerts a repelling force during its training process, but it is based on a well-defined energy function, hence the generated map is optimal. Furthermore, rRBF does not require separate training phases, as the self-organizing process is regulated by the top-down supervised learning.

Models of SOM that concatenate labels into input vectors were also proposed in [19]–[23] as classifiers. In the learning process of these models, the best matching unit (BMU) is decided based on the extended input, while in the classification phase, given an unlabeled input, BMU is decided based only on the input's features while the extended components of the reference vector associated with the BMU are used to classify the input. While these SOMs incorporated data labels in their training process, they fundamentally inherited SOMs characteristic of pulling together similar inputs but not explicitly repelling away similar inputs when they have different contexts. The proposed rRBF offers a more comprehensive relation between the features and the contexts in which the contextual error that occurs in the top layer regulates the self-organization of the input in the bottom layers. The top-down regularization allows the rRBF to repel similar inputs with different contexts or to group together dissimilar inputs sharing the same context. In [24], clustering was executed in the low-dimensional space of SOM and further used for classifying unknown data. The semantic context preservation in rRBF often produces clusters in CRSOM. However, semantic context preservation in rRBF inherently produces context-relevant clusters in its topographical map, and hence does not require an additional clustering mechanism.

Some variants of SOM also incorporate information that are not embedded in the input vectors. For example, SOM for structured data (SOM-SD) [25] was introduced to visualize graph-structured data. GraphSOM [26] is an extension of SOM-SD that further allows the inclusion of the contextual information in a directed graph. It should be noted that context in this paper has a different meaning from that of the previous studies above. While in the previous studies, context refers to the graphical relational structure of the data; in this paper, we consider semantic interpretation of the data as the context, which can be independent of their structure. For example, different labels can be attached to the same data set.

Real-world data often have temporal context, which cannot be directly visualized with the original SOM. A rich collection of SOM models that attempt to visualize temporal context in low-dimensional maps have been proposed. Temporal Kohonen map [27] and its variant recurrent SOM [28] capture temporal information of the input vector by utilizing leaky integrator activation functions. Thus, in contrast to the original SOM, in these models, the activity of a neuron in the map depends not only on the current input but also on previous inputs whose influence decays with time. Self-organizing time map [29]–[31] is a 2-D SOM where a particular row of neurons in the map is a 1-D projection of a multidimensional input at a particular time, and adjacent rows receive inputs from adjacent time steps. Merge SOM [32] merges the weight vectors with a designated context descriptor—the weighted sum of the past BMUs' weight vectors—to characterize a neuron in the map. While in the previous study, the primary objective is the visualization of high-dimensional data in the context of their dynamics, the proposed rRBF visualizes the data in the perspective of semantical context that can be considered as the higher level abstraction to describe the data.

There were also variants of SOM, which build topographical maps based not solely on the similarities of the data. For example, Visualization-Induced Self-Organizing Map [33] combines topographical preservation with local distance preservation of the data, while the SOM model in [34] exerts a repelling force so that the distances between the data in their original high-dimensional space are more accurately preserved. While the rRBF also generates a nonconventional SOM, it is unique in that the self-organization process is influenced by the semantic context of the data and optimizes a well-defined energy function.

The rRBF is also unique in that during its regulated bottom-up self-organizing process, the repelling force acts as an inhibitory signal for some hidden neurons. This is because the activations for those neurons decrease when they are repelled. This inhibition results in sparser hidden representations compared with the conventional SOM. Although the inhibition in rRBF is fundamentally different from that of Willshaw–von der Malsburg SOM model [35], it is interesting that rRBF, through repelling force, inhibits neurons to produce sparser maps.

The multilayered perceptron [36] can also be used to map high-dimensional data into a lower dimension in its hidden layer. However, it does not generate a topographical structure of the problem. For example, there is no mechanism to prevent two similar inputs belonging to the same class from being mapped into two distant points, so long as the weighted value into the output neurons are identical. The topological restriction in the activation in the hidden neurons in the rRBF generates visualizable structures.

## III. Restricted Radial Basis Function Network

### A. Learning-Regulated Topographical Map

The proposed rRBF network shown in Fig. 1 is based on the conventional RBF network [37], [38]. It is a three-layered network where the neurons in the hidden layers are aligned in a 2-D grid like SOM. Receiving input vector $X(t)$ at time $t$, rRBF selects a winning neuron to win among all the hidden neurons according to

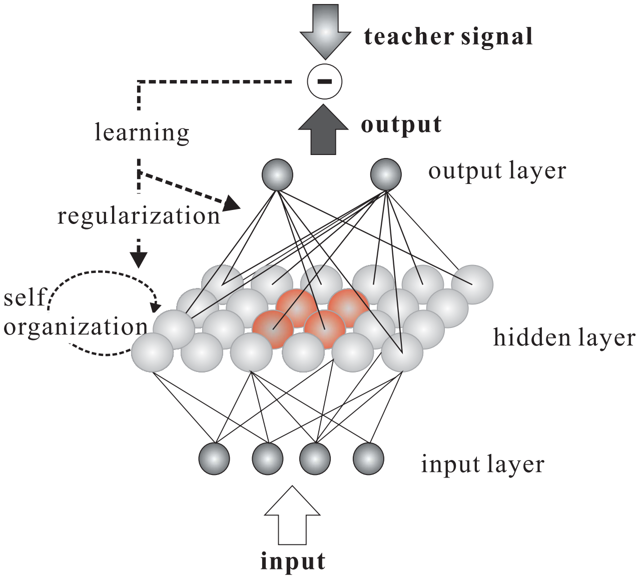


Fig. 1. rRBF network that includes a self-organizing representational layer and an output layer that performs classification. Self-organization, which is commonly considered a purely bottom-up process, is regulated by a top-down teacher signal.

$$\text{win} = \arg\min_j \|X(t) - W_j(t)\|^2 \tag{1}$$

in which $W_j$ is a prototype vector associated with the $j$th hidden neuron. Once the winner is decided, the output of the $i$th hidden neuron can be calculated as

$$\begin{aligned} O_i^h(t) &= e^{-I_i^h(t)}\sigma(\text{win}, i, t) \\ I_i^h(t) &= \|X(t) - W_i(t)\|^2. \end{aligned} \tag{2}$$

The function $\sigma(\text{win}, i, t)$ is a neighborhood functions defined

$$\begin{aligned} \sigma(\text{win}, i, t) &= e^{-\frac{\text{dist}(\text{win}, i)}{s(t)}} \\ s(t) &= s_0\left(\frac{s_{\text{end}}}{s_0}\right)^{\frac{t}{t_{\text{end}}}}. \end{aligned} \tag{3}$$

In (3), dist(win, $i$) is the distance between the winner neuron and the $i$th neuron in the hidden layer. Parameters $s_0$ and $s_{\text{end}}$ are the neighborhood sizes at the beginning and end of the learning process, respectively, and $t_{\text{end}}$ represents the number of training iterations. They are empirically set to $s_0 = 200$, $s_{\text{end}} = 0.01$, and $t_{\text{end}} = 30\,000$. Thus, the neighborhood is large to begin with such that all nodes are integrated into the map, and then decreases in size to allow the nodes to increase their specificity. For the experiments, we chose a parameter set that produced relatively good MSE for all the problems, though not necessarily optimal for any given problem. Once chosen, the parameters were then fixed for all the experiments.

From (2), the output of the hidden neuron is topologically restricted by the neighborhood function, hence we used the term restricted to name the network. The outputs from the hidden neurons are then propagated to the output layer, so that the output of the $k$th output layer can be calculated as

$$O_k(t) = f(I_k(t)) \tag{4}$$
$$I_k(t) = \sum_i v_{\mathrm{ik}}(t) O_i^h(t) - \theta_k(t)$$
$$f(x) = \frac{1}{1+e^{-x}}. \tag{5}$$

The parameter $v_{\mathrm{ik}}$ is the weight connecting the $i$th hidden neuron and the $k$th output neuron, $\theta_k$ is the bias of the $k$th neuron, while $f$ is a sigmoid function.

Because rRBF is a supervised-training network, once the output is defined, we can then calculate the energy function as

$$E(t) = \frac{1}{2}\sum_k (O_k(t) - T_k(t))^2 \tag{6}$$

where $T_k(t)$ is the $k$th component of the teacher signal that represents the ideal context of the input at time $t$. As in the backpropagation algorithm [36], the learning process modifies the connection weight with a gradient-descent rule

$$v_{\mathrm{ik}}(t+1) = v_{\mathrm{ik}}(t) - \eta_1 \frac{\partial E(t)}{\partial v_{\mathrm{ik}}(t)}. \tag{7}$$

Here, $\eta_1$ is the learning rate for the connection weights leading to the output layer, and adopting the delta rule [36] yields the following:

$$v_{\mathrm{ik}}(t+1) = v_{\mathrm{ik}}(t) - \eta_1 \delta_k(t) O_i^h(t)$$
$$\delta_k(t) = (O_k(t) - T_k(t)) O_k(t)(1 - O_k(t)). \tag{8}$$

The modification of the bias is as follows:

$$\begin{aligned}\theta_k(t+1) &= \theta_k(t) - \eta_1 \frac{\partial E(t)}{\partial \theta_k(t)} \\ &= \theta_k(t) + \eta_1 \delta_k(t).\end{aligned} \tag{9}$$

Similarly, the prototype vector of the $i$th hidden neuron is modified as

$$W_i(t+1) = W_i(t) - \eta_2 \frac{\partial E(t)}{\partial W_i(t)} \tag{10}$$

in which $\eta_2$ is the learning rate for the prototype vector modification. The modification of connection weight from the $j$th input neuron to the $i$th hidden neuron, $w_{ij}$, can be calculated

$$\begin{aligned}\frac{\partial E}{\partial w_{ij}} &= \sum_k \frac{\partial E}{\partial O_k}\frac{\partial O_k}{\partial I_k}\frac{\partial I_k}{\partial w_{ij}} \\ &= \sum_k \frac{\partial E}{\partial O_k}\frac{\partial O_k}{\partial I_k}\frac{\partial I_k}{\partial O_i^h}\frac{\partial O_i^h}{\partial w_{ij}} \\ &= e^{-I_i^h}\left(\sum_k \delta_k v_{\mathrm{ik}}\right)\sigma(\mathrm{win}, i, t)(x_j - w_{ij}).\end{aligned} \tag{11}$$

If we define $\delta_i^h(t)$ as

$$\delta_i^h(t) = -e^{-I_i^h(t)}\left(\sum_k \delta_k(t) v_{\mathrm{ik}}(t)\right) \tag{12}$$

the modification of the code vector associated with the $i$th hidden neuron, $W_i$, becomes

$$W_i(t+1) = W_i(t) + \eta_2 \delta_i^h(t)\sigma(\mathrm{win}, i, t)(X(t) - W_i(t)). \tag{13}$$

The prototype vector modification in (13) is similar to the weight modification in a standard SOM algorithm except that in SOM, it is always corrected in the direction of input vector $X$. The modification in regular SOM can be achieved with $\delta_i^h(t) = 1$ in (13). In this paper, the sign of $\delta_i^h(t)$ is decided by the value of $\sum_k \delta_k v_{\mathrm{ik}}$. It is obvious that the formation process of CRSOM differs from the conventional SOM, in that when $\delta_i^h(t) > 0$, the prototype vectors are modified toward the input vector, while when $\delta_i^h(t) < 0$ they are repelled from the input. Thus, $\delta_i^h(t)$ acts as a regulatory signal to the modification of the prototype vectors. Because the value of $\delta_i^h(t)$ is defined by the error from the supervised layer that tries to learn the context of the data in the output layer, this signal regulates the map formation process to reflect the context of the data. Thus, only prototype vectors that contribute to the decrease of the energy function are reinforced by being moved toward the input, while prototype vectors that are not contributing to the minimization of the energy function will be repelled from the input. The repelling is a kind of penalty, especially to the winning neurons that are not contributing to the learning process, because by being repelled their competitiveness of being chosen as the winning neurons decreases. Gradually, there will be fewer neurons that are chosen as winners, resulting in more efficient usage of hidden neurons. The conventional SOM implicitly repels an input vector from the neurons associated with dissimilar references vectors by selecting a winner with more similar vector. The repelling mechanism is more explicit in rRBF, in that an input vector can be repelled from the winner according to the feedback from the output layer.

### B. Semantic Relevance Index

Here, semantic relevance index (SRI) to quantify the ability of CRSOM to preserve the semantic relevance of the data in a low-dimensional space is introduced. In this paper, the focus is on classification problems, thus the semantic relevance of a data point is its class label. The SRI is the ratio between the interclass index that measures how well the CRSOM separates data belonging to different classes and the intraclass index that measures how well the CRSOM binds data belonging to the same class label together. The interclass index, $I_{\mathrm{class}}$, is calculated as follows:

$$I_{\mathrm{class}} = \frac{1}{N(M-1)}\sum_{i=1}^{N}\sum_{j\neq C(X_i)} \|H(X_i) - H(\mathrm{min}_{\mathrm{out}}(X_i, j))\|. \tag{14}$$

Here, $H(X_i)$ is the coordinate of the winning neuron in the map and $\mathrm{min}_{\mathrm{out}}(X_i, j)$ is the nearest data point to $X_i$ in their original high-dimensional space, belonging to class $j$. $C(X_i)$ is the class label of data point $X_i$. $N$ and $M$ are the number of the data points and the number of classes, respectively. The intraclass index, $O_{\mathrm{class}}$, is calculated as

$$O_{\mathrm{class}} = \frac{1}{N}\sum_{i=1}^{N} \|H(X_i) - H(\mathrm{max}_{\mathrm{in}}(X_i))\|. \tag{15}$$

Here, $\max_{\text{in}}(X_i)$ denotes the data point belonging to $C(X_i)$ having the largest distance from $X_i$ in their original high-dimensional space. The SRI is then defined as

$$\text{SRI} = \frac{O_{\text{class}}}{I_{\text{class}}}. \tag{16}$$

A semantic preserving map generates a large value of $O_{\text{class}}$, because data points belonging to different classes are repelled from each other, while at the same time generates a small value of $I_{\text{class}}$ because data belonging to the same class are pulled toward each other. SRI gives a quantitative measure of the topographical map's semantic sharpness independent from the size of the map.

### *C. Generalized SOM*

The unsupervised training process of a regular SOM is based on an heuristic prototype vector modification rule that minimizes the difference between the input vector and the winning neuron and its neighbors, and there is no known global energy function for this learning process. Unlike the conventional SOM, CRSOM is organized based on an explicit energy function, as defined in (6). If the proposed rRBF can be trained to generate conventional SOM, we can argue that CRSOM is a generalized SOM. To generate SOM with rRBF, the teacher signal for the energy function in (6) needs to be defined. For deciding the teacher signal that will produce SOM with rRBF, we can assume the activation of the hidden layer will be sparse. This is because the activation function of the hidden neurons, defined in (2), ensures that only neurons in the vicinity of the winner are relevant to the learning process. We can then uniquely generate a teacher signal that will produce an SOM if we assume that the number of the hidden neurons that significantly contribute to the learning process is $K$, the same number as the output neurons. While this may be a strong assumption in the beginning of the learning process, it is mostly true after some learning iterations. It is sufficient that we design a teacher signal so that $\delta^h$ in (13) is 1. Hence

$$\begin{aligned} \delta_i^h &= -O_i^h \sum_k v_{\text{ik}} \delta_k = 1 \\ i &\in N_K. \end{aligned} \tag{17}$$

In the above equation, $N_K$ is the set of the winner neuron and its nearest $K-1$ neighbors, while $\delta_k$ is the error signal from the $k$th output neuron, as defined in (8). Defining

$$D = \begin{pmatrix} O_1(1-O_1) & 0 & \dots & 0 \\ 0 & O_2(1-O_2) & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & O_K(1-O_K) \end{pmatrix}$$

$$V = \begin{pmatrix} v_{11}^n & v_{12}^n & \dots & v_{1K}^n \\ v_{21}^n & v_{22}^n & \dots & v_{2K}^n \\ \vdots & \vdots & \ddots & \vdots \\ v_{K1}^n & v_{K2}^n & \dots & v_{KK}^n \end{pmatrix}$$

and

$$W^N = VD \tag{18}$$

where $v_{ij}^n$ is the connection weight leading from the $i$th neuron in the neighborhood set, $N_k$, into the $j$th output neuron, and considering only neurons in $N_K$, (17) can be written as follows:

$$W^N \begin{pmatrix} T_1 \\ T_2 \\ \vdots \\ T_K \end{pmatrix} = \begin{pmatrix} \frac{1}{O_{n_1}^h} \\ \frac{1}{O_{n_2}^h} \\ \vdots \\ \frac{1}{O_{n_k}^h} \end{pmatrix} + W^N \begin{pmatrix} O_1 \\ O_2 \\ \vdots \\ O_k \end{pmatrix}. \tag{19}$$

Here, $n_i \in N_k$. Hence

$$\begin{pmatrix} T_1 \\ T_2 \\ \vdots \\ T_{N_o} \end{pmatrix} = W_n^{-1} \begin{pmatrix} \frac{1}{O_{N_1}^h} \\ \frac{1}{O_{N_2}^h} \\ \vdots \\ \frac{1}{O_{N_k}^h} \end{pmatrix} + \begin{pmatrix} O_1 \\ O_2 \\ \vdots \\ O_k \end{pmatrix}. \tag{20}$$

Equation (20) shows the teacher signal that guarantee $\delta_i^h = 1, i \in N_k$, and thus generating conventional SOM in the hidden layer of rRBF. Unlike the conventional SOM, the rRBF-generated SOM has an explicit energy function, at least for the neurons neighboring the winner. Equation (19) shows that the generated teacher signal causes the same modification as SOM for the prototype vector associated with the neighborhood neurons. However, it is easy to increase the number of output neurons so that it equals the number of hidden neurons. Hence, SOM is a special case of CRSOM, with floating teacher signals that change according to the output and weights of the hidden layer.

### *D. Computational Scalability*

One of the reasons for SOMs popularity is its computational scalability with regard to the increase in data and map sizes. Here, it is shown that rRBF inherits SOMs computational scalability. Let $L_{\text{SOM}}$ be the calculation time needed to update all the reference vectors in an SOM, $d_1$ be the dimension of the input vector, and $M$ be the number of neurons in that SOM. Naturally

$$L_{\text{SOM}} \propto M d_1. \tag{21}$$

For rRBF, let $L_{\text{rRBF}}$ be the calculating time needed to update all the reference vectors, the connection weights between the hidden and the output layers and the biases of the output neurons, and $d_2$ be the number of the output neurons, hence

$$L_{r\text{RBF}} \propto M d_1 + M d_2 + d_2. \tag{22}$$

Hence

$$\frac{L_{r\text{RBF}}}{L_{\text{SOM}}} = 1 + \frac{d_2}{d_1} + \frac{d_2}{M d_1}. \tag{23}$$

Because it is natural that $M d_1 \gg d_2$ and $d_2 \leq d_1$

$$\frac{L_{r\text{RBF}}}{L_{\text{SOM}}} \approx 1 + \frac{d_2}{d_1} \leq 2. \tag{24}$$

Equation (24) shows that the computational time for training rRBF is at most twice that of SOM, hence rRBF inherits the computational scalability of SOM.

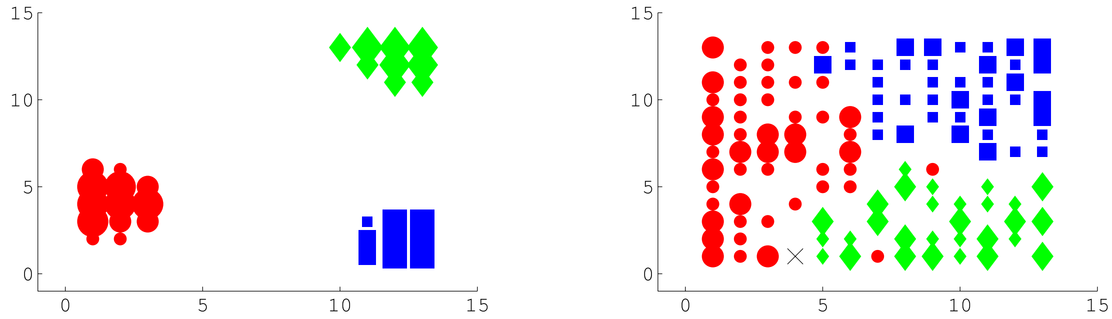


Fig. 2. Wine classification. Left: CRSOM(SRI = 32.76). Right: SOM(SRI = 1.23).

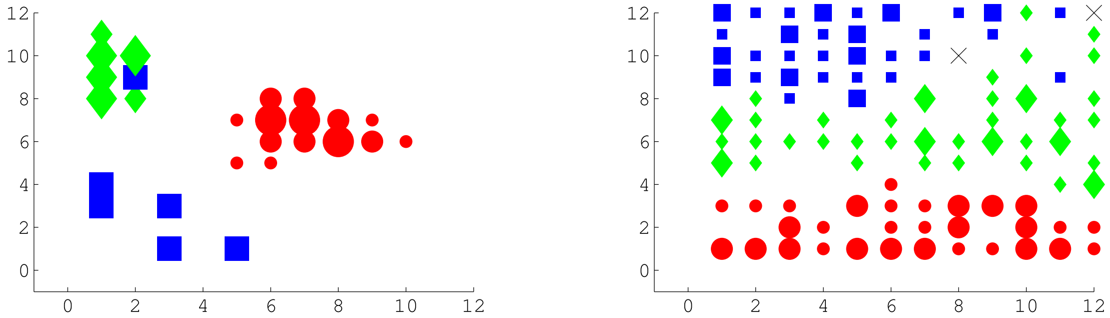


Fig. 3. Iris classification. Left: CRSOM(SRI = 5.47). Right: SOM(SRI = 0.57).

## IV. Experiments

In the experiments, we tested the proposed rRBF to create CRSOM for a number of benchmark problems taken from UCI machine learning repository [40]. For result comparisons, in each experiment, the size of the map is set to $\lfloor\sqrt{N}\rfloor \times \lfloor\sqrt{N}\rfloor$, where $N$ is the size of the data.

### A. Labeled Data

Fig. 2 shows the topographical maps for Wine classification, which is a 13-D three-class classification problem. In these maps, neurons that were selected as the winners for the inputs belonging to the respective class are illustrated with three different shapes (□, ○, and ◇). The size of these shapes corresponds to their winning frequencies. Fig. 2 (left) shows the CRSOM for this problem, while Fig. 2 (right) shows the conventional SOM for this problem, where × is a neuron that is chosen as a winner by multiple inputs from different classes. A high SRI value for CRSOM clearly indicates that there are distinct clusters of classes. Here, it is obvious that in their current semantic context (three class labels), this classification problem can be nicely separated into three clusters. While from the conventional SOM shown in Fig. 2 (right), the topographical information of the data can be extracted, CRSOM in Fig. 2 (left) offers information about the separability of the problem. The large empty space in CRSOM is the implication of the repelling force during the learning process. A winner that is repelled from a particular input vector will become less sensitive to the same input and eventually will lose to other neuron, leaving it empty, thus creating large margins between the clusters.

Fig. 3 shows two topographical maps for Iris classification, a 4-D three-class problem. It is well known that in this problem, one of the classes is linearly separable from the other two, while those two are not. Given the natural semantic of this problem, CRSOM in Fig. 3 (left) nicely captures the well-known characteristics of this problem, in which the class represented by ○ clearly forms a cluster that is separable from the other two, while the separability of those two are less obvious. The SOM in Fig. 3 (right) also visualizes the topological characteristics of the data, where each class forms a tight cluster, but fails to visualize the semantic relation. The difference in semantic context preserving abilities between CRSOM and SOM is also quantitatively represented by the difference in their SRI values.

Fig. 4 (left) shows the CRSOM for thyroid classification problem, two of the classes (□ and ○) are, respectively, represented by more than one clusters. This figure actually captures the balancing mechanism between keeping the original topological characteristics of the data and reflecting their semantic contexts in the low-dimensional map. In this case, CRSOM did not exert enough force to bring together the three groups of □s to create one large cluster because of the repelling forces from the nearby neurons. The conventional SOM for this problem is shown in Fig. 4 (right).

The first three examples have shown the visualization performance of CRSOM against relatively easy classification problems. The next examples show the internal visualization of more challenging problems.

Fig. 5 (left) shows that the CRSOM for Pima classification problem did not form solid clusters for each of the two classes, which is also quantified by its low SRI value.

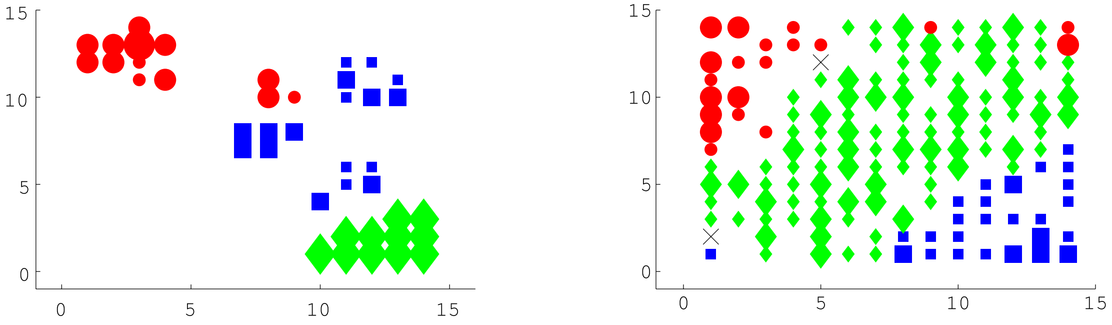


Fig. 4. Thyroid classification. Left: CRSOM(SRI = 11.08). Right: SOM(SRI = 0.44).

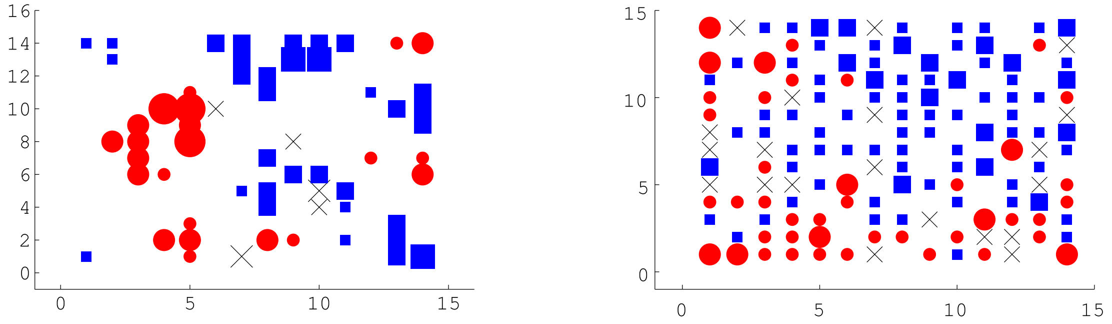


Fig. 5. Pima classification. Left: CRSOM(SRI = 0.85). Right: SOM(SRI = 0.055).

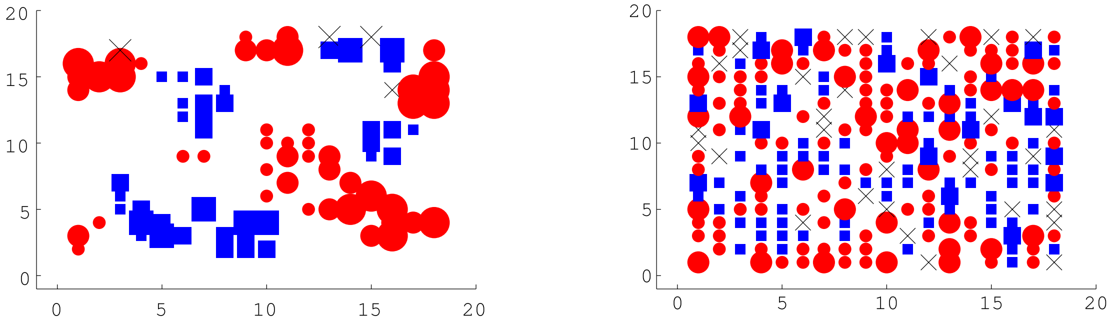


Fig. 6. Bupa classification. Left: CRSOM(SRI = 0.81). Right: SOM(SRI = 0.070).

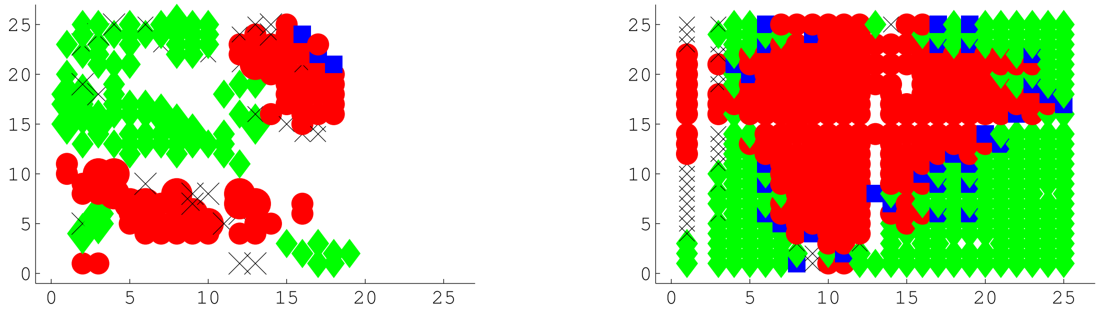


Fig. 7. Balance classification. Left: CRSOM(SRI = 0.71). Right: SOM(SRI = 0.21).

Hence, there are multiple borders between the two clusters, which strongly indicate that this is a nonlinear classification problem. The conventional SOM for the same problem is shown in Fig. 5 (right) where the borders between the two classes are less obvious.

The next problem, Bupa classification, for which the CRSOM is shown in Fig. 6 (left), is similar to the previous one in that the two classes did not form two solid clusters but many smaller clusters bordering each other. There are also more × neurons, which indicates some misclassified data. The conventional SOM for the same problem is shown in Fig. 6 (right) where no distinctive cluster can be observed.

The final example is the balance classification problem, a three-class problem, with a very few examples for one of the classes. In CRSOM in Fig. 7 (left), the

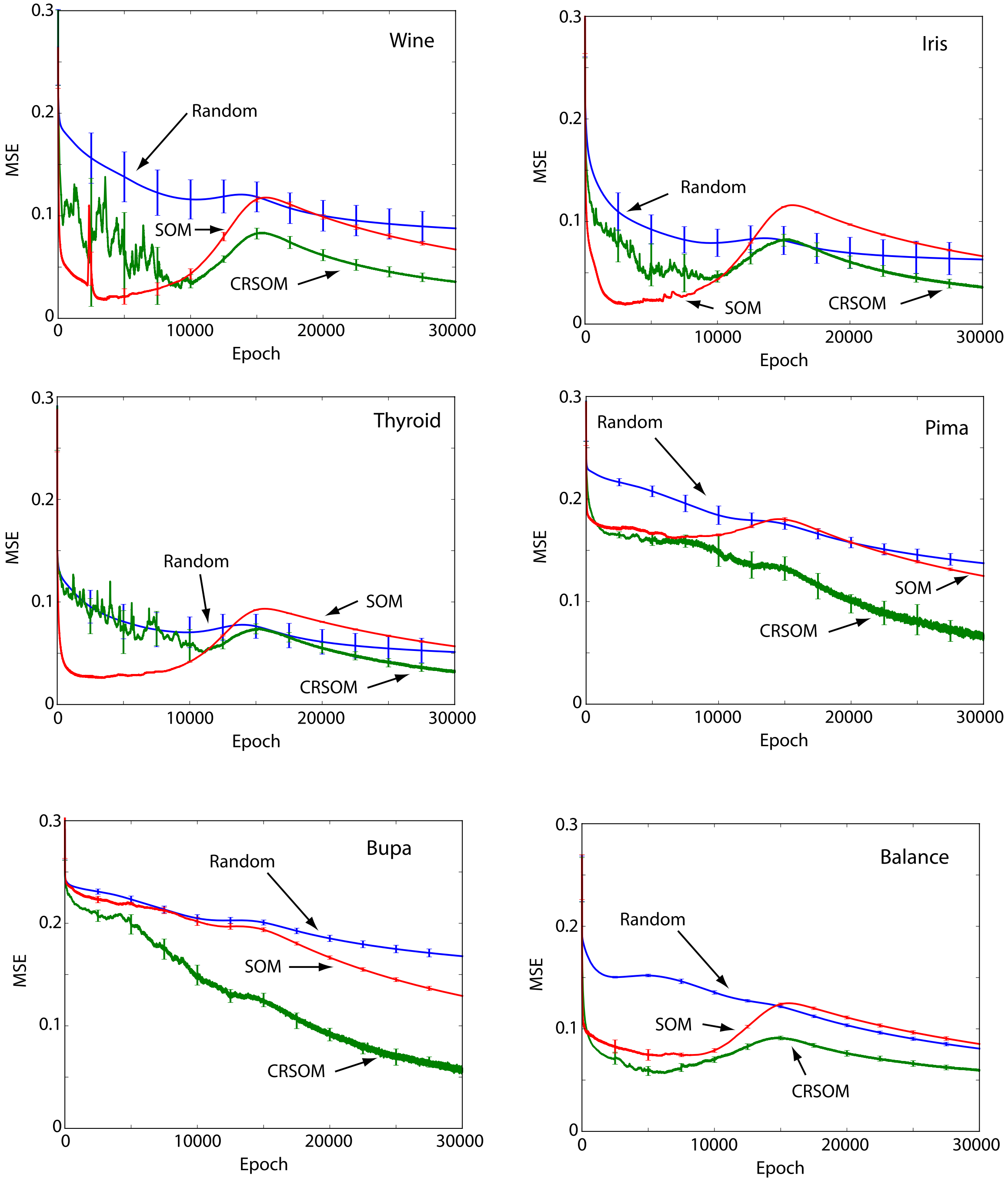


Fig. 8. Learning curves with random representations, with SOM and CRSOM representations. The curves include example error bars depicting standard deviation.

under-represented class is shown by □, and the two classes formed a number of relatively large clusters. Many × indicate that this is a relatively difficult classification problem.

Figs. 2–7 are shown to illustrate the difference in visualizing the structure of the respective problem. Hence, training data from each problem were used to build those maps.

Fig. 8 shows the learning curves of rRBF for the six classification problems. The learning process of rRBF, depicted as CRSOM, is compared with two learning algorithms, one with $\delta_i^h(t)$ in (13) fixed to 1 thus generating the conventional SOM in its hidden layer, shown as SOM, and the other with $\eta_2 = 0$ thus having a random hidden representation, shown as random, where the maps are shown in Fig. 9. Fig. 8 shows that rRBF, with CRSOM as its internal representation, learns more effectively than the other two networks, especially in the last three problems where its internal representation appears more organized than SOMs. The error bars in Fig. 10 shows the mean of the classification over 10 random initializations of the networks' connection weights, where it can be observed that CRSOM outperforms the other two networks in its classification ability.

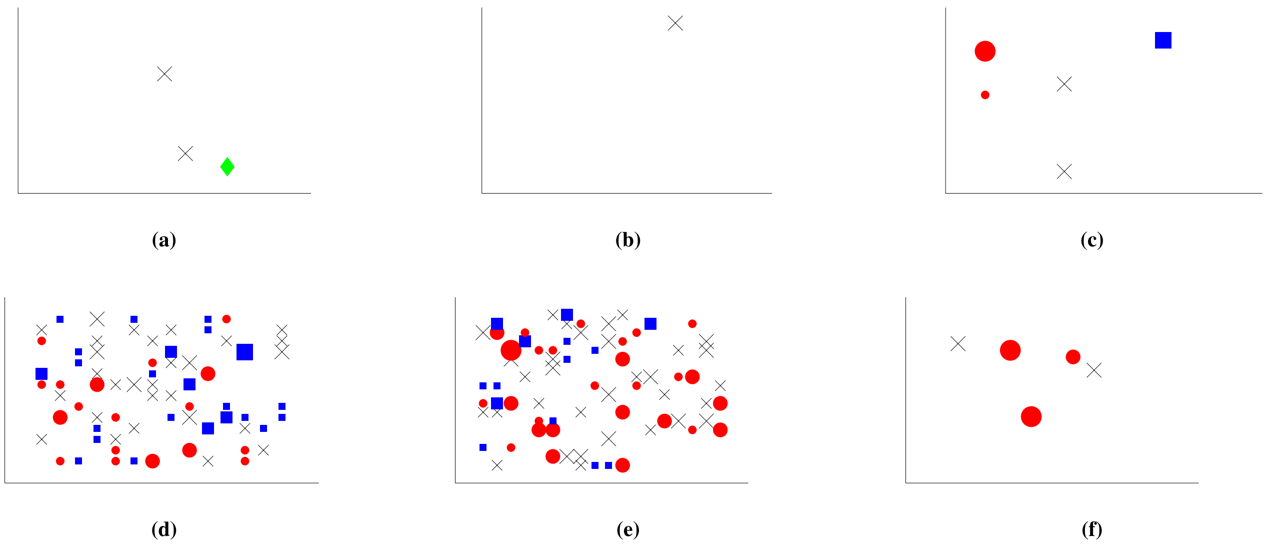


Fig. 9. Maps for the different problems with random initial condition when the learning rate of the self-organizing layer is set to zero. The results are therefore random maps from which the perceptron has to work. (a) Wine. (b) Iris. (c) Thyroid. (d) Pima. (e) Bupa. (f) Balance.

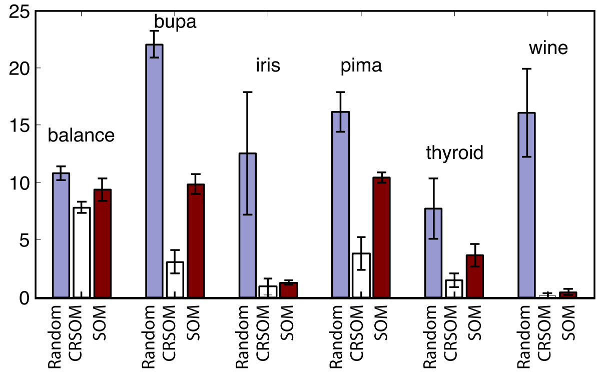


Fig. 10. Mean classification error and standard deviation over 10 runs with random initializations.

TABLE I
MEAN SRI

| Dataset | Random | CRSOM | SOM |
|---|---|---|---|
| Balance | $0.0558 \pm 0.0021$ | $\mathbf{0.969} \pm 0.105$ | $0.1830 \pm 0.0182$ |
| Bupa | $0.0252 \pm 0.0025$ | $\mathbf{0.8392} \pm 0.18714$ | $0.0698 \pm 0.0016$ |
| Iris | $0.2940 \pm 0.0780$ | $\mathbf{7.490} \pm 1.161$ | $0.4737 \pm 0.0068$ |
| Pima | $0.0439 \pm 0.0043$ | $\mathbf{1.0294} \pm 0.2165$ | $0.0566 \pm 0.0037$ |
| Thyroid | $0.1414 \pm 0.0544$ | $\mathbf{15.792} \pm 2.2119$ | $0.4569 \pm 0.0175$ |
| Wine | $0.1251 \pm 0.0345$ | $\mathbf{22.88} \pm 9.156$ | $1.140 \pm 0.175$ |

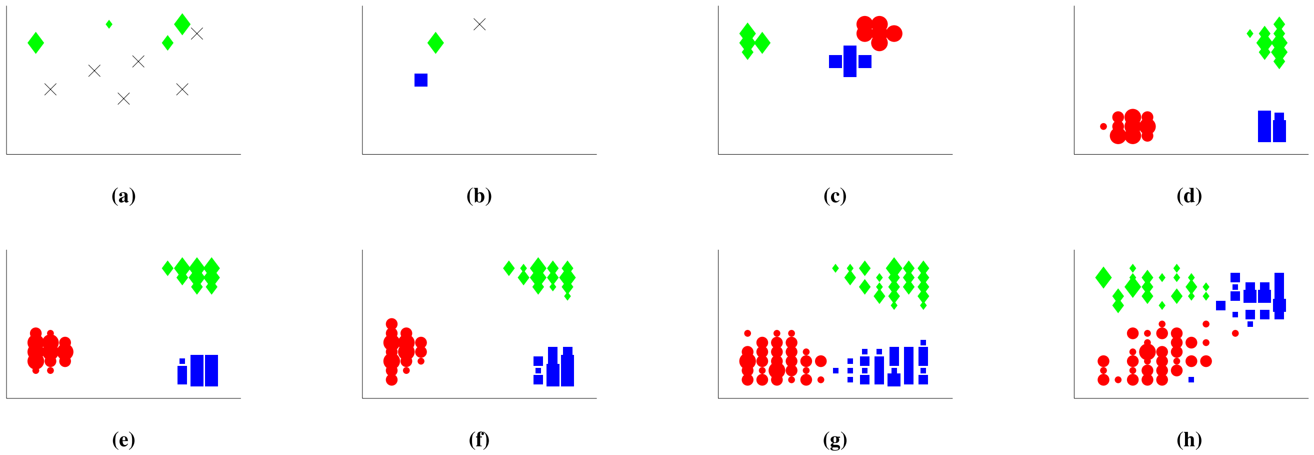
In this paper, the main objective is to show the superiority of CRSOM in preserving the semantic context of the data, which we qualitatively evaluate using SRI defined in (16). Table I shows the mean SRIs of random map, CRSOM, and SOM. For all the classification problems, CRSOM consistently outperforms SOM in SRI, which indicates that the learning process of rRBF generates a low-dimensional map that balances the topological preservation of the data with the consideration of their semantic context. The context-preserving characteristic of CRSOM gives us an alternative to the conventional SOM, in that now we are able to visualize semantic context-relevant high-dimensional data.

We also empirically analyze the sensitivity of the learning process with regards to the change of the learning hyperparameters. Here, because our focus is on the visualization characteristics of CRSOM, we analyze the effect of the bottom-up learning rate $\eta_2$. Fig. 11 and Fig. 12 shows the effect on the formation of CRSOM of varying $\eta_2$ between 0 and 0.1 while $\eta_1$ is fixed. Fig. 11(a) shows the CRSOM in the case of $\eta_2 = 0$, where no bottom-up learning occurs. Along with the change in visual appearance of the CRSOM, it is also interesting to observe the change in SRI value. It can be observed that there is no significant change in the appearance of the CRSOM and SRI value, when the value of $\eta_2$ is between 0.01 and 0.02. From this experiment, we learned that the formation of CRSOM is to some extent sensitive to the learning rate, $\eta_2$, such that there is a critical value, which has to be empirically set to generate CRSOM with a large SRI value. However, our experiments with various problems, and thus various map sizes, indicates that once the appropriate $\eta_2$ is found, it can be used in a wide range of problems.

### *B. Visualizing Robot's Internal Model*

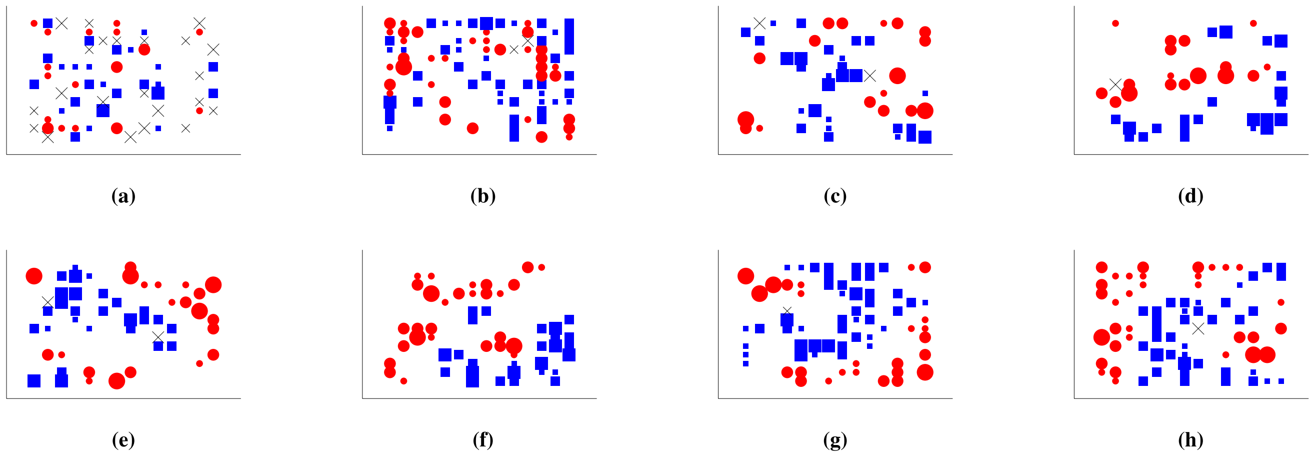
In the next experiment, the rRBF was tested against real-world data acquired from a robotic experiment shown in Fig. 13. In this experiment, a small robot, e-puck [39], with eight proximity sensors ran in an environment with several obstacles. The robot was trained to classify its current situation as safe when it is located free from obstacles, or dangerous when it was close to some obstacles. In this case, the inputs to the rRBF were the sensor values and the outputs were their context of safe or dangerous. Thus, the generated CRSOM corresponds to visualization of the internal concept of safe and dangerous for this robot.

The maps formed by the robot during interacting in this environment are shown on the right side of Fig. 13, the CRSOM results are on the top, while the SOM results are


(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 11. Bottom-up learning sensitivity for Wine data. The maps with an increasing bottom-up learning rate are shown. Notice that there is an optimal learning rate with respect to the SRI. (a) $\eta_2 = 0$, SRI $= 0.8$. (b) $\eta_2 = 0.001$, SRI $= 5.04$. (c) $\eta_2 = 0.005$, SRI $= 30.8$. (d) $\eta_2 = 0.01$, SRI $= 42.2$. (e) $\eta_2 = 0.015$, SRI $= 32.8$. (f) $\eta_2 = 0.02$, SRI $= 25.7$. (g) $\eta_2 = 0.05$, SRI $= 7.96$. (h) $\eta_2 = 0.1$, SRI $= 5.13$.


(a) (b) (c) (d)

(e) (f) (g) (h)

Fig. 12. Bottom-up learning sensitivity for Pima data. The maps with an increasing bottom-up learning rate are shown. There is an optimal intermediate learning rate with respect to the SRI. (a) $\eta_2 = 0$, SRI $= 0.63$. (b) $\eta_2 = 0.001$, SRI $= 0.78$. (c) $\eta_2 = 0.05$, SRI $= 0.95$. (d) $\eta_2 = 0.01$, SRI $= 1.2$. (e) $\eta_2 = 0.015$, SRI $= 1.2$. (f) $\eta_2 = 0.02$, SRI $= 1.1$. (g) $\eta_2 = 0.05$, SRI $= 0.83$. (h) $\eta_2 = 0.1$, SRI $= 0.82$.

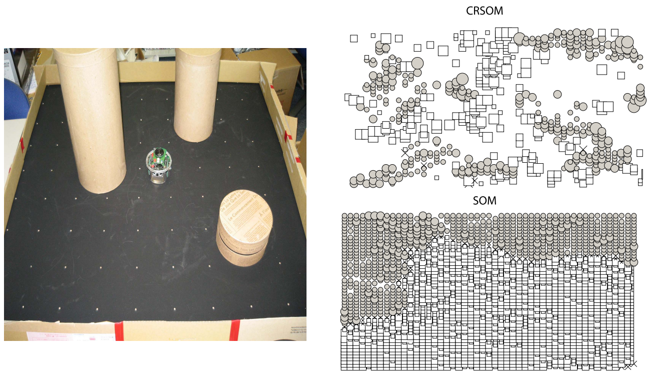


Fig. 13. Robot experiment. Left: setup of the robot learning example with an e-puck in an enclosed area that includes three obstacles. Right: maps formed from experience with CRSOM (top) and SOM (bottom).

on the bottom. The neurons that react for the safe and dangerous situations have thereby different symbols and shadings. It is obvious that CRSOM shows a more complex structure than SOM, in which the concepts of safety and danger are not bound as single cluster but distributed as pocket of clusters bordering each other. This structure is intuitive, because dangerous situations in the environment can usually be turned into safe ones with a number of maneuvers of the robot.

### C. Autoencoder

We also tested the proposed rRBF against problems with different contexts. In the next experiments, rRBF was trained using the same six problems not to classify the input but to generate an autoencoder [41], in that the rRBF has to reconstruct a given input in its output layer. Hence, the internal layer is a kind of abstraction of the high-dimensional input space. The CRSOM for the six problems are shown in Fig. 14. Clearly, the CRSOMs with autoencoder context differ from the respective CRSOM with label context on all the problems. For these six CRSOMs, although the classes of the input did not have any influence on the formation of the map, for the purpose of clarity, each neuron is illustrated

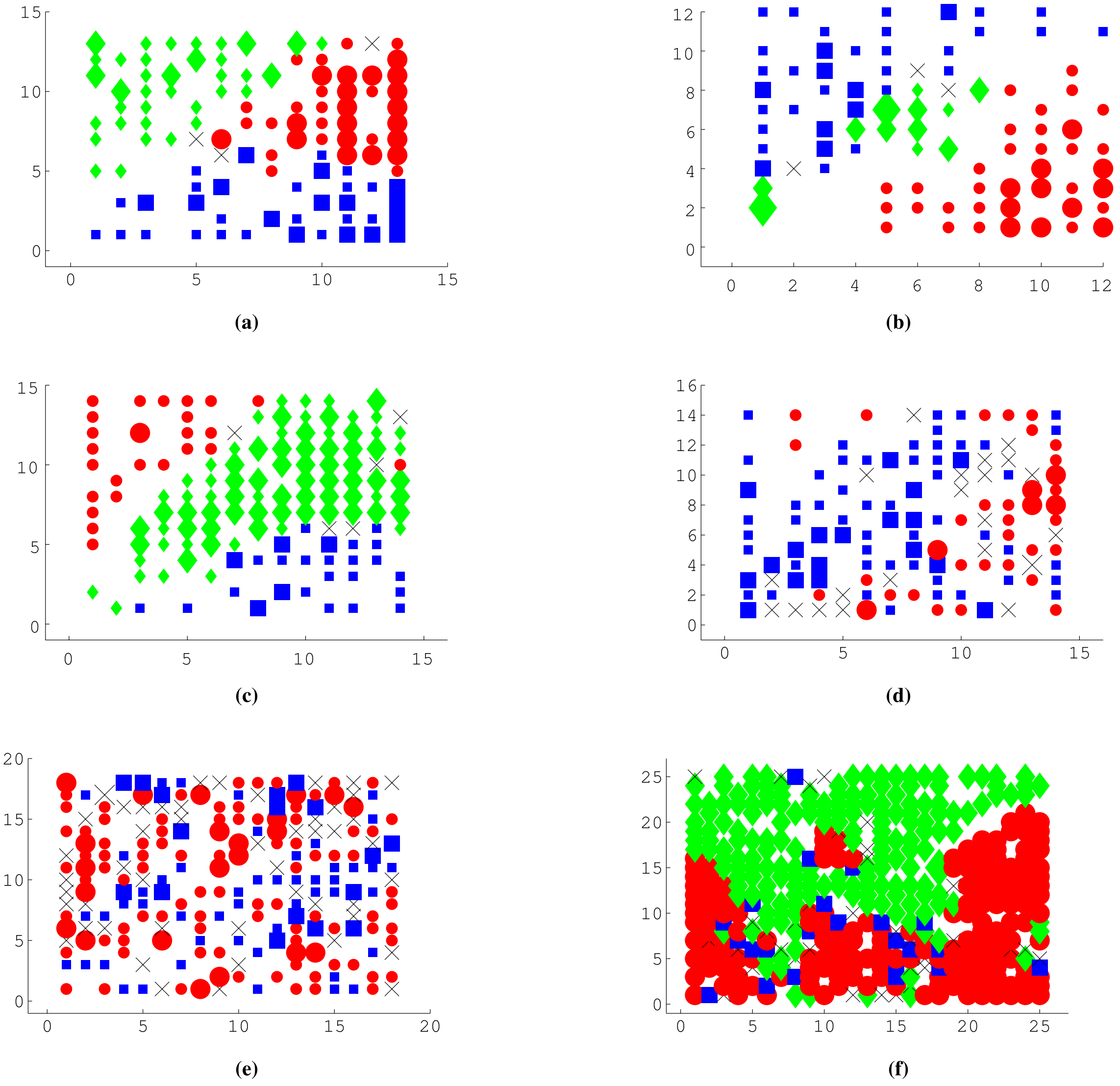


Fig. 14. CRSOM for autoencoder. (a) Wine. (b) Iris. (c) Thyroid. (d) Pima. (e) Bupa. (f) Balance.

with a different shape as before. The difference between CRSOM with the same input but different contexts confirms that rRBF is able to build context relevant topographic maps.

## V. Conclusion

In this paper, we first explained the learning properties of rRBF where the mathematical derivation shows that rRBF organized high-dimensional data differently from the conventional SOM. The empirical experiments support our arguments that, unlike SOM which self-organized high-dimensional data based on their topological properties regardless of their semantic context, CRSOM incorporates the semantic context in its top-down regulated self-organization process. As different semantic contexts can be attached to the same data based on free interpretation, CRSOM offers visualization not of the data but of the problem, which is fundamentally different from SOM and other similarity-based dimension reduction techniques.

Our long-term objective is to develop a network model that resembles brain organization and is based on layered maps that represent hierarchical concepts. Hence, the immediate future topic is to expand the rRBF into a deep structured network [42], [43], and investigate how a multilayered rRBF forms a set of CRSOMs. It is also of interest to observe the evolution of the map over the learning process, as this can give insights into the formation of internal concepts inside a classifier. It is also interesting to observe the change of internal representation of a classifier during concept drifting in nonstationary problems [44]. In this paper, we only implemented rRBF with one hidden layer; however, it is easy to extend the number of hidden layers. Although in this paper, the main focus was the context-preserving visualization of high-dimensional data, in the next study, the generalization performance of the rRBF, as a classifier, will be investigated. The sparsity in the internal representation of rRBF may cause overfitting, hence modifications to the learning rule, including mapping and learning parameters, need to be considered.

The proposed rRBF can also potentially be applied for semisupervised training [45], since once rRBF is partially trained, it is easy to map unlabeled data into its internal layer, and utilize CRSOM to generate the labels for the data before executing the supervised learning.

## Acknowledgment

The authors would like to thank the reviewers for their valuable input.

## References

[1] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biol. Cybern.*, vol. 43, no. 1, pp. 59–69, 1982.

[2] T. Kohonen, "Essentials of the self-organizing map," *Neural Netw.*, vol. 37, pp. 52–65, Jan. 2013.

[3] T. Trappenberg, P. Hartono, and D. Rasmusson, "Top-down control of learning in biological self-organizing maps," in *Proc. 7th Int. Workshop Self-Org. Maps (WSOM)*, vol. 5629. 2009, pp. 316–324.

[4] T. Trappenberg, *Fundamentals of Computational Neuroscience*, 2nd ed. London, U.K.: Oxford Univ. Press, 2010.

[5] D. Merkl, "Text classification with self-organizing maps: Some lessons learned," *Neurocomputing*, vol. 21, nos. 1–3, pp. 61–77, 1998.

[6] P. Hartono and T. Trappenberg, "Classificability-regulated self-organizing map using restricted RBF," in *Proc. IEEE IJCNN*, Aug. 2013, pp. 160–164.

[7] P. Hartono and T. Trappenberg, "Learning initialized by topologically correct map," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, 2009, pp. 2802–2806.

[8] P. Földiák, *Sparse Coding in the Primate Cortex*, M. A. Arbib, Ed., 2nd ed. Cambridge, MA, USA: MIT Press, 2002.

[9] H. Ritter and T. Kohonen, "Self-organizing semantic maps," *Biol. Cybern.*, vol. 61, no. 4, pp. 241–254, 1989.

[10] R. Hecht-Nielsen, "Counterpropagation networks," *Appl. Opt.*, vol. 26, no. 23, pp. 4979–4984, 1987.

[11] J. Göppert and W. Rosenstiel, "Self-organizing maps vs. backpropagation: An experimental study," in *Proc. Workshop Design Methodol. Microelectron. Signal Process.*, 1993, pp. 153–162.

[12] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[13] G. E. Hinton and S. T. Roweis, "Stochastic neighbor embedding, in *Advances in Neural Information Processing Systems*, vol. 15. Cambridge, MA, USA: MIT Press, 2002, pp. 833–840.

[14] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE, *J. Mach. Learn. Res.*, vol. 9, no. 85, pp. 2579–2605, 2008.

[15] J. B. Kruskal, "Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis," *Psycometrika*, vol. 29, no. 1, pp. 1–27, 1964.

[16] J. W. Sammon, "A nonlinear mapping for data structure analysis," *IEEE Trans. Comput.*, vol. C-18, no. 5, pp. 401–409, May 1969.

[17] J. B. Tenenbaum, V. de Selva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[18] T. Kohonen, "The hypermap architecture," in *Artificial Neural Networks*, vol. 2, T. Kohonen, K. Mäkisara, O. Simula, and J. Kangas, Eds. Amsterdam, The Netherlands: Elsevier, 1991, pp. 1357–1360.

[19] A. Ultsch, G. Guimaraes, and W. Schmidt, "Classification and prediction of hail using self-organizing neural networks," in *Proc. IEEE Int. Conf. Neural Netw. (ICNN)*, Jun. 1996, pp. 1622–1627.

[20] T. Koga, K. Horio, and T. Yamakawa, "The self-organizing relationship (SOR) network employing fuzzy inference based heuristic evaluation," *Neural Netw.*, vol. 19, no. 6, pp. 799–811, 2006.

[21] T. Yamakawa and T. Horio, "Self-organizing relationship (SOR) network," *IEICE Trans. Fundam. Electron., Commun. Comput. Sci.*, vol. E82-A, no. 8, pp. 1674–1677, 1999.

[22] G. A. Barreto and A. F. R. Araujo, "Identification and control of dynamical systems using the self-organizing map," *IEEE Trans. Neural Netw.*, vol. 15, no. 5, pp. 1244–1259, Sep. 2004.

[23] H. Ichiki, M. Hagiwara, and M. Nakagawa, "Kohonen feature maps as a supervised learning machine," in *Proc. IEEE Int. Conf. Neural Netw.*, Mar. 1993, pp. 1944–1948.

[24] P. Sarlin and D. Marghescu, "Visual predictions of currency crises using self-organizing maps," *Intell. Syst. Accounting, Finance Manage.*, vol. 18, no. 1, pp. 15–38, 2011.

[25] M. Hagenbuchner, A. C. Tsoi, and A. Sperduti, "A supervised self-organizing map for structured data," in *Advances in Self-Organizing Maps*, N. Allinson, H. Yin, L. Allinson, and J. Slack, Eds. Berlin, Germany: Springer-Verlag, 2001, pp. 21–28.

[26] S. Zhang, M. Hagenbuchner, A. C. Tsoi, and A. Sperduti, "Self organizing maps for the clustering of large sets of labeled graphs," in *Advances in Focused Retrieval* (Lecture Notes in Computer Science), vol. 5631, S. Geva, J. Kamps, and A. Trotman, Eds. Berlin, Germany: Springer-Verlag, 2009, pp. 469–481.

[27] G. J. Chappell and J. G. Taylor, "The temporal Kohonen map," *Neural Netw.*, vol. 6, no. 3, pp. 441–445, 1993.

[28] T. Koskela, M. Varsta, J. Heikkonen, and K. Kaski, "Time series prediction using RSOM with local linear models," *Int. J. Knowl.-Based Intell. Eng. Syst.*, vol. 2, no. 1, pp. 60–68, 1998.

[29] P. Sarlin, "Self-organizing time map: An abstraction of temporal multivariate patterns," *Neurocomputing*, vol. 99, no. 1, pp. 496–508, 2012.

[30] P. Sarlin, "A self-organizing time map for time-to-event data," in *Proc. IEEE Symp. Comput. Intell. Data Mining*, Apr. 2013, pp. 230–237.

[31] P. Sarlin, "Replacing the time dimension: A self-organizing map over any variable," in *Proc. Workshop New Challenges Neural Comput.*, B. Hammer, T. Martinetz, and T. Villmann, Eds., 2013.

[32] M. Strickert and B. Hammer, "Merge SOM for temporal data," *Neurocomputing*, vol. 64, pp. 39–72, Mar. 2005.

[33] H. Yin, "ViSOM—A novel method for multivariate data projection and structure visualization," *IEEE Trans. Neural Netw.*, vol. 13, no. 1, pp. 237–243, Jan. 2002.

[34] Y. M. Cheung and L. T. Law, "Rival-model penalized self-organizing map," *IEEE Trans. Neural Netw.*, vol. 28, no. 1, pp. 289–295, Jan. 2007.

[35] D. J. Willshaw and C. von der Malsburg, "How patterned neural connections can be set up by self-organization," *Proc. Roy. Soc. London B, Biol. Sci.*, vol. 194, no. 1117, pp. 431–445, 1976.

[36] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," in *Parallel Distributed Processing*, vol. 1. Cambridge, MA, USA: MIT Press, 1984, pp. 318–362.

[37] J. Moody and C. J. Darken, "Fast learning in networks of locally-tuned processing units," *Neural Comput.*, vol. 1, no. 2, pp. 281–294, 1989.

[38] T. Poggio and F. Girosi, "Networks for approximation and learning," *Proc. IEEE*, vol. 78, no. 9, pp. 1481–1497, Sep. 1990.

[39] F. Mondada *et al.*, "The e-puck, a robot designed for education in engineering," in *Proc. 9th Conf. Auto. Robot Syst. Competitions*, 2009, vol. 1, no. 1, pp. 59–65.

[40] *UCI Repository*. [Online]. Available: http://www.ics.uci.edu/~mlearn/MLRepository.html, accessed Aug. 1, 2013.

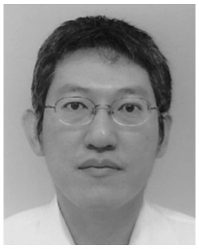
[41] H. Bourlard and Y. Kamp, "Auto-association by multilayer perceptrons and singular value decomposition," *Biol. Cybern.*, vol. 59, nos. 4–5, pp. 291–294, 1988.

[42] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2279–2324, Nov. 1998.

[43] Y. Bengio, "Learning deep architectures for AI," *Found. Trends Mach. Learn.*, vol. 2, no. 1, pp. 1–127, 2009.

[44] R. Elwell and R. Polikar, "Incremental learning of concept drift in nonstationary environments," *IEEE Trans. Neural Netw.*, vol. 22, no. 10, pp. 1517–1531, Oct. 2011.

[45] R. G. F. Soares, H. Chen, and X. Yao, "Semisupervised classification with cluster regularization," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 11, pp. 1779–1792, Nov. 2012.



**Pitoyo Hartono** (M'96) received the B.Eng., M.Eng., and Dr.Ing. degrees in applied physics from Waseda University, Tokyo, Japan, in 1993, 1995, and 2002, respectively.

He was with Hitachi, Ltd., Tokyo, from 1995 to 1998. From 2001 to 2005, he was a Research Associate and Visiting Lecturer with Waseda University, and an Associate Professor with Future University Hakodate, Hakodate, Japan, from 2005 to 2010. He is currently a Professor with the School of Engineering, Chukyo University, Nagoya, Japan. His current research interests include computational intelligence and robotics.

**Paul Hollensen** studied cognitive science with the University of Toronto, Toronto, ON, Canada, followed by computer science with Dalhousie University, Halifax, NS, Canada, where he is currently pursuing the Ph.D. degree.

His current research interests include computational neuroscience and hierarchical machine learning methods.

**Thomas Trappenberg** (M'14) received the Ph.D. degree in physics from RWTH Aachen University, Aachen, Germany.

He held research positions at Dalhousie University, Halifax, NS, Canada, the RIKEN Brain Science Institute, Wako, Japan, and Oxford University, Oxford, U.K. He is currently a Full Professor of Computer Science and has authored a book entitled *Fundamentals of Computational Neuroscience—Second Edition*. His current research interests include computational neuroscience, machine learning, and neurocognitive robotics.