LoopSOM: A Robust SOM Variant Using Self-Organizing Temporal Feedback Connections

Rafael C. Pinto, Paulo M. Engel

Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
P.O. Box 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

{rcpinto,engel}@inf.ufrgs.br

Abstract. *This paper introduces feedback connections into a previously proposed model of the simple and complex neurons of the neocortex. The original model considers only feedforward connections between a SOM (Self-Organizing Map) and a RSOM (Recurrent SOM). A variant of the SOM-RSOM pair is proposed, called LoopSOM. The RSOM sends predictions to the SOM, providing more robust pattern classification/recognition and solving ambiguities.*

1. Introduction

Previously, [Miller and Lommel 2006] have proposed a model of the simple/complex cortical cells as the basic processing unit of his neocortex model, the HQSOM (Hierarchical Quilted Self-Organizing Map). According to this model, the simple neurons are implemented by SOMs (Self-Organizing Maps, [Kohonen 1998]), providing spatial clustering, classification and recognition, while complex neurons are implemented by RSOMs (Recurrent SOM's, [Koskela et al. 1998]), allowing for the temporal portion.

However, the RSOM in the SOM-RSOM pair can't make predictions for the SOM, as proposed in the Memory-Prediction Framework (MPF) in [Hawkins 2005], which was the base for Miller's work. This work proposes to introduce feedback connections into the SOM-RSOM pair, allowing the RSOM to send predictions to the SOM, providing better pattern classification/recognition and solving ambiguities. Also, with this improvement, it perfectly fits into the MPF. This new algorithm is called LoopSOM.

The rest of this paper is organized as follows. The next section briefly explains the Memory-Prediction Framework. Sections 3 and 4 review the SOM and RSOM algorithms. Section 5 presents the LoopSOM and related works. In section 6, experiments and results comparing the LoopSOM and the SOM-RSOM pair are shown. Section 7 finishes this work with conclusions and future works.

2. The Memory-Prediction Framework

Hawkins proposed a framework for computational modeling of the neocortex, the Memory-Prediction Framework (MPF). Basically it says the basic unit of the model must do spatial and temporal processing, such that the temporal portion sends predictions to the spatial portion. The output of the spatial part serves as input to the temporal part, which recognizes/classifies sequences. The temporal part sends its outputs to the next layer, and also sends predictions back to the spatial part. The cortex is built up from layers with many of these processing units.

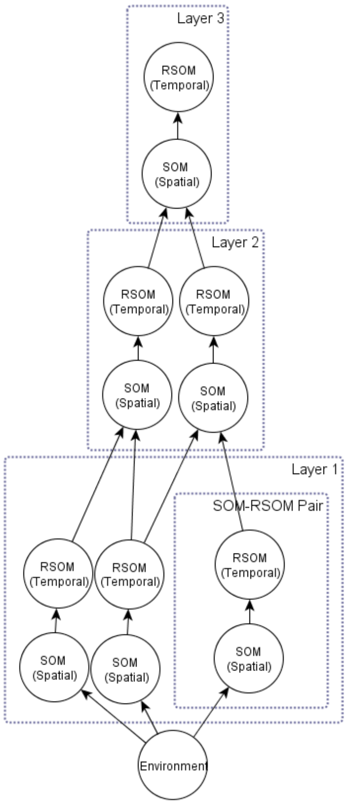

Figure 1. HQSOM diagram and the SOM-RSOM pair. The HQSOM may have any number of SOM-RSOM pairs on each layer and any number of layers.

As higher layers are examined, the concepts become more and more abstract (holding concepts like “person” or “ball”), and more concrete concepts are on the bottom, like pixels of an image. Another important part of the MPF is that every sensory modality must use that same mechanism. The sensory inputs are what makes each modality unique.

There are a few implementations of the MPF by now, like the Hierarchical Temporal Memory (HTM), proposed in [Hawkins and George 2006], and the Hierarchical Quilted Self-Organizing Map (HQSOM) proposed by Miller. The former is a Bayesian network implementation, while the later is a neural network implementation, but lacks prediction connections, as required by the MPF.

The basic processing unit inside the HQSOM is the SOM-RSOM pair (Figure 1), which will be improved by adding the lacking connections, using the RSOM BMU weight vector as feedback (Figure 2).

3. Spatial Clustering

A Self-Organizing Map is an unsupervised neural network intended for spatial clustering, vector quantization, dimensionality reduction and topology preservation. Topology preservation means that patterns close in input space produce patterns close in output (map) space. The neuron map forms a code-book of patterns from input space, composed by synapse weights. The learning takes place in a winner-take-all approach, by selecting

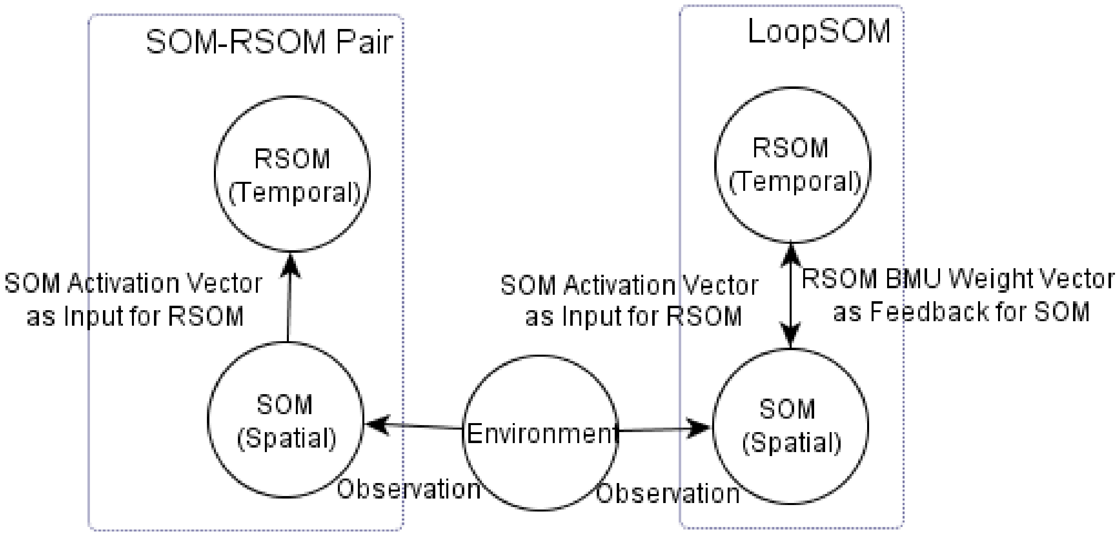

Figure 2. Original SOM-RSOM pair and the proposed LoopSOM

the best matching unit (BMU) with the following equation:

$$\|\mathbf{x}(t) - \mathbf{w}_b\| = \min_{i \in V_O}\{\|\mathbf{x}(t) - \mathbf{w}_i\|\} \tag{1}$$

where V_O is the output space, $\mathbf{x}$ is the input vector and $\mathbf{w}_i$ is the weight vector of the ith unit. After finding the BMU, its weights are updated, as well as the weights of its neighbors, according to the following update rule:

$$\mathbf{w}_i(t+1) = \mathbf{w}_i(t) + \gamma h_{ib}(t)(\mathbf{x}(t) - \mathbf{w}_i(t)) \tag{2}$$

where h_{ib} is a neighborhood function such as:

$$h_{ib}(t) = \exp\left(\frac{-\|\mathbf{I}_i - \mathbf{I}_b\|^2}{2\sigma(t)^2}\right) \tag{3}$$

where $\mathbf{I}_i$ and $\mathbf{I}_b$ are indices of the map units i and b, and $\sigma(t)$ is the gaussian standard deviation. Note that σ is dependent on time and normally is implemented into a cooling schedule. The problem with that approach is that it's not possible to do online learning. To fix it, Miller proposed to change the previous equation by the following one

$$h_{ib}(t) = \exp\left(\frac{-\|\mathbf{I}_i - \mathbf{I}_b\|^2}{\mu_b(t)\sigma^2}\right) \tag{4}$$

where $\mu_b(t)$ is the mean squared error of $\mathbf{w}_b$ compared to the input $\mathbf{x}$(t), and is given as follows:

$$\mu_b(t) = \frac{1}{N}\|\mathbf{x}(t) - \mathbf{w}_b\|^2 \tag{5}$$

where N is the length of the input vector. This enables the SOM for online learning by adjusting dynamically the neighborhood size. Another possible approaches for online learning are the PLSOM (Parameter-less SOM, [Berglund and Sitte 2003]) and its improved version in [Berglund 2009], although Miller's simpler approach will be used for

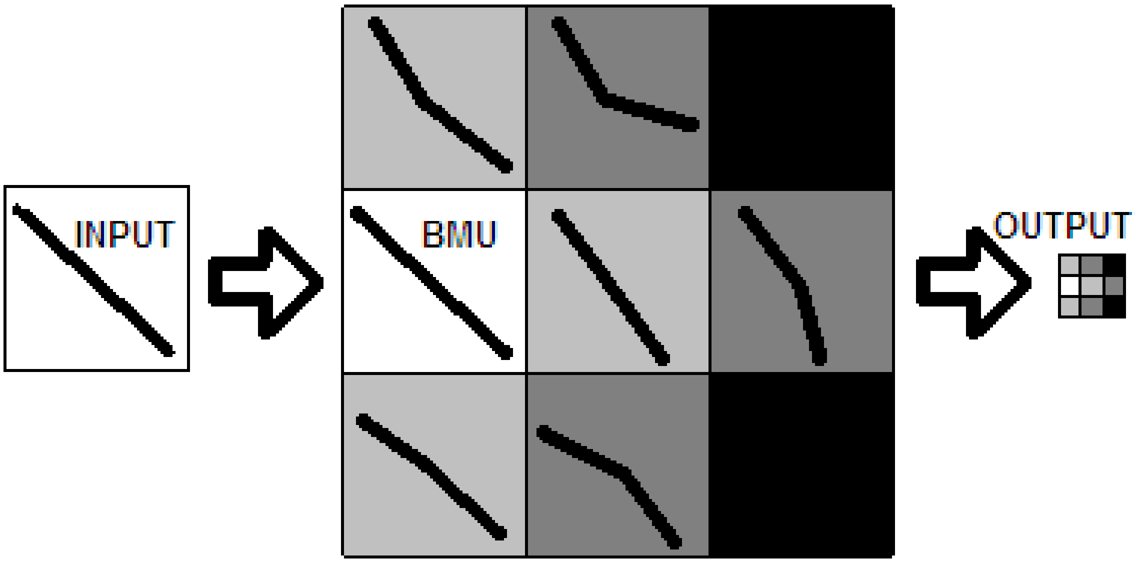

Figure 3. The input pattern is compared to each SOM unit, selecting the BMU. An activation vector may be produced from the distances vector from the input.

now. Additionally, an activation vector may be produced from the distances vector from the input x(t) with some function like a gaussian:

$$y_i(t) = \exp\left(\frac{-\|\mathbf{x}(t) - \mathbf{w}_i\|^2}{2\rho^2}\right) \tag{6}$$

where ρ is the standard deviation of the gaussian. Smaller ρ means more local coding while bigger ρ means denser coding [Foldiak and Young 1995]. This vector will be useful later as input to another map for creating a SOM-RSOM pair. See figure 3 for an example.

4. Temporal Clustering

The Recurrent SOM performs temporal clustering by using decayed traces of previous vector differences. At each time step a recursive difference vector $\mathbf{d}_i(t)$ is calculated as:

$$\mathbf{d}_i(t) = (1-\alpha)\mathbf{d}_i(t-1) + \alpha(\mathbf{x}(t) - \mathbf{w}_i(t)) \tag{7}$$

where $\alpha, 0 \leq \alpha \leq 1$ is the decay factor, **x** is the input vector and $\mathbf{w}_i$ is the weight vector of the ith unit. The memory becomes deeper as α gets closer to 0, being the original SOM a special case of the RSOM where α = 1 (no memory). Now the BMU can be found by using the following equation:

$$\mathbf{d}_b(t) = \min_{i \in V_O}\{\|\mathbf{d}_i(t)\|\} \tag{8}$$

and the new update rule is as follows:

$$\mathbf{w}_i(t+1) = \mathbf{w}_i(t) + \gamma h_{ib}(t)\mathbf{d}_i(t) \tag{9}$$

The result is a set of invariant representations for patterns which are correlated temporally.

A problem with the RSOM is that if the input vectors are not orthogonal, some ambiguity will be created. To overcome this problem, Miller proposed to use the SOM output **y** (equation 6) as the input **x** for the RSOM (equation 7), creating a SOM-RSOM pair. So, the smaller the used ρ, more orthogonal will be the output and the RSOM will work better.

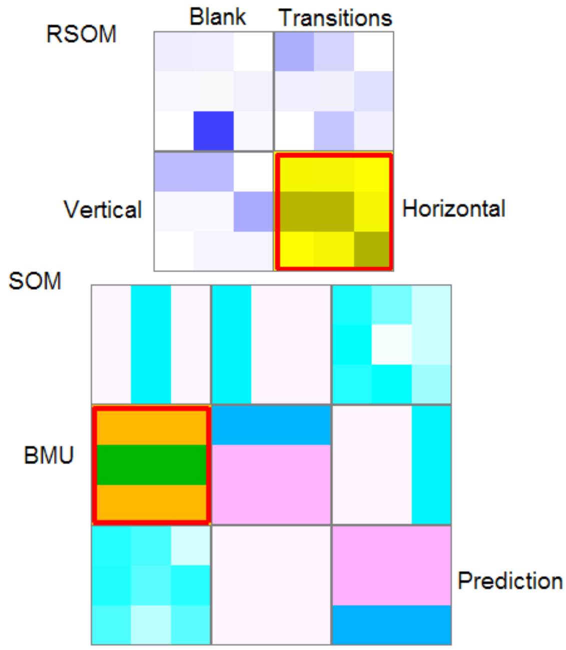

Figure 4. The LoopSOM. Spatial SOM bellow, temporal RSOM above. Activations in yellow, predictions in purple, coincident activations and predictions in red/orange. The BMU is highlighted in red. The used inputs are the ones described in section 6.2.

5. The LoopSOM

The SOM-RSOM pair works very well for spatio-temporal pattern processing, but it doesn't use all of its potential and doesn't conform totally to the MPF, because of the lack of feedback connections carrying predictions from the RSOM to the SOM. Such connections may solve ambiguities on the spatial level and provide higher noise tolerance, resulting in a more robust unit. A new implementation, called LoopSOM, aims to provide a way to implement such connections. In the LoopSOM, the weight vector $\mathbf{w}_b$ of the RSOM BMU is used as an activation prediction $\mathbf{p}(t)$ for the SOM, as shown in figure 4. So, the prediction is as follows:

$$\mathbf{p}(t) = \mathbf{w}_{bt}(t-1) \tag{10}$$

The current SOM activation vector **y**(t) will be combined with the last RSOM prediction **p**(t-1), and this new equation will be used to compute the highest activation and find the BMU:

$$y_b(t) = \max_{i \in V_O} \left(\frac{\xi_s(t) y_i(t) + \xi_t(t-1) p_i(t)}{\xi_s(t) + \xi_t(t-1)} \right) \tag{11}$$

where $\xi_s(t)$ is the spatial SOM output confidence computed as follows:

$$\xi_s(t) = 1 - \frac{1}{2} \left\| \frac{\mathbf{x}_s(t)}{\|\mathbf{x}_s(t)\|} - \frac{\mathbf{w}_{bs}(t)}{\|\mathbf{w}_{bs}(t)\|} \right\| \tag{12}$$

where $\mathbf{x}_s(t)$ is the input vector and $\mathbf{w}_{bs}(t)$ is the weight vector of the BMU. Note that the input must be processed with the original BMU equation (1) in order to get the confidence value, and the SOM learning rate must be kept at 0 while doing it, to avoid interfering with the new BMU calculation. In the same way $\xi_t(t-1)$ can be computed as:

$$\xi_t(t-1) = 1 - \frac{1}{2}\left\| \frac{\mathbf{x}_t(t-1)}{\|\mathbf{x}_t(t-1)\|} - \frac{\mathbf{w}_{bt}(t-1)}{\|\mathbf{w}_{bt}(t-1)\|} \right\| \tag{13}$$

being the variables analogue to the ones on the previous equation. Here the values from the previous pattern presentation must be used, since they are used to predict the current activations. Analyzing these 2 factors, some extreme cases can be shown: when both confidences are similar, current observations and predictions will weight near the same. When $\xi_s > 0$ and ξ_t equals 0, the original SOM-RSOM pair without feedback is obtained. When ξ_s equals 0 and $\xi_t > 0$, pure prediction is obtained and could even generate the most likely sequences decoupled from the environment. This can be seen as a form of fall back behavior, as proposed by [Cohen 1998], so that the classification can fall back to pure spatial classification when temporal classification is sub-optimal.

The two confidence values may be used in other ways too. For instance, the learning rate γ of the RSOM can be a function $\gamma(\xi_s(t))$ (it will learn less if there is error on the spatial SOM).

Such combination with the RSOM prediction allows the SOM to correctly classify ambiguous patterns by providing a way to select between 2 otherwise equally plausible winning nodes.

5.1. Related Works

Another possible replacements for the SOM-RSOM pair are the Anticipatory SOM (AntSOM, [Swarup et al. 2005a]) and the Recurrent AntSOM (RecAntSOM, [Swarup et al. 2005b]). The former performs simple predictions using activation counters and uses only a conventional SOM, while the later makes predictions using Simple Recurrent Networks (SRN, [Elman 1990]). Possible advantages of the LoopSOM over them are the adaptive weighting and the explicit invariant representation, but it's out of the scope of this work to compare such algorithms.

6. Experiments and Results

To compare the SOM-RSOM pair and the LoopSOM performances, two simple experiments were created.

6.1. Four Points in 2D Space

Four points in 2D space were presented to both algorithms: $\{$(0,0),(0,1),(1,0),(1,1)$\}$. They are grouped by their first term: group0 = $\{$(0,0),(0,1)$\}$, group1 = $\{$(1,0),(1,1)$\}$ and are presented in a way that points in the same group are more likely to show in the next step (90% chance), resulting in 50% for each group. After 5000 steps the weights in the SOM and RSOM are frozen and each SOM unit is labeled with 0 or 1 according to the best matches for each of them. Then the same points are presented, but now there's some chance of showing a totally ambiguous point (0.5,0.5). The tests involved chances within the range from 0% to 100%, with 1% increments. The parameters used were:

- SOM: size = 2x2, $\rho = 0.125$, $\sigma = 1$, γ = 0.1 (fixed, not adaptive);

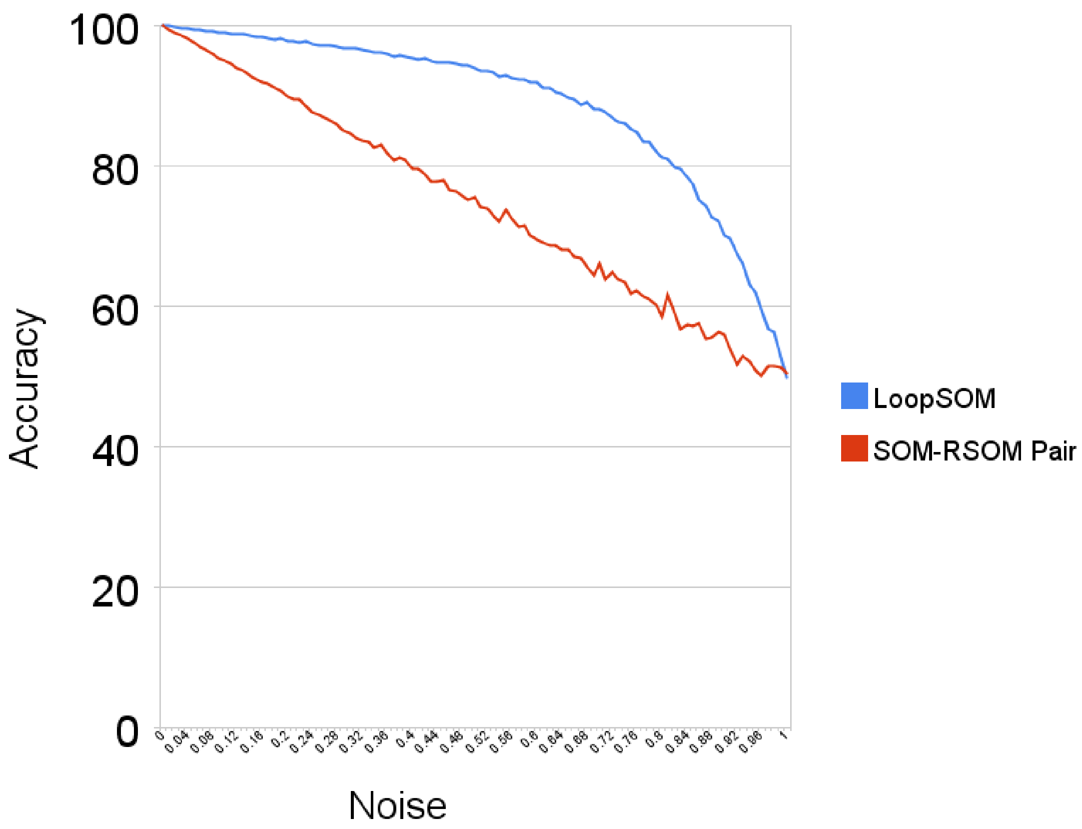

Figure 5. Results comparing the SOM-RSOM pair and the LoopSOM accuracy.

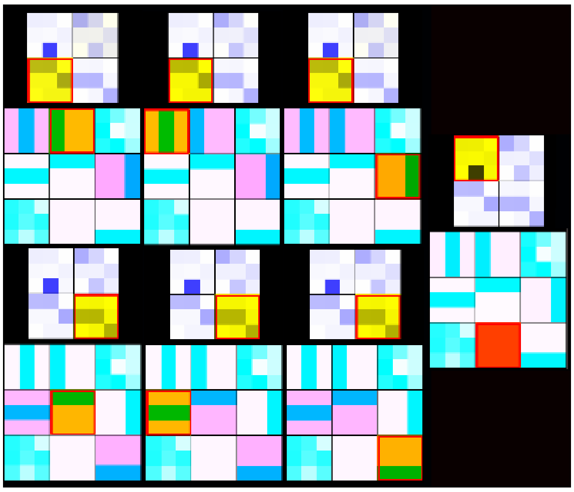

Figure 6. All 7 inputs shown to the LoopSOM and its respective states. Vertical on the top , horizontal on the bottom and blank on the right. Note how each group has only 1 invariant representation on the RSOM.

- RSOM: size = 2x1, $\rho = 0.125$, $\alpha = 0.3$, $\sigma = 0.7$, γ = 0.01 (fixed, not adaptive);
- For the SOM-RSOM pair, the same implementation was used but with fixed $\xi_s = 1$ and $\xi_t = 0$.

Results are shown in Figure 5. The LoopSOM dominates the SOM-RSOM pair, although they are similar on extreme cases (0% and 100%). Note that the errors on the SOM-RSOM pair approximates the a posteriori probabilities of showing an ambiguous pattern with equal priors for each group (50%). The LoopSOM adds information to this prior probabilities, increasing accuracy by moving it from maximum likelihood to maximum a posteriori.

6.2. 2D Lines

In the next experiment, 7 different 2D visual patterns with 3x3 pixels are presented: 3 horizontal lines, 3 vertical lines and 1 empty pattern (Figure 6). They are grouped as horizontal, vertical and blank, and patterns in the same group has 90% chance of showing in next step, resulting in nearly 33% for each group. After 20000 steps the weights in the SOM and RSOM are frozen and each SOM unit is labeled according to the best matches for each group. Then the same patterns are presented, but now there's some chance of showing a totally ambiguous pattern (a cross). Chances within the range from 0% to 100%, with 1% increments were tested. The used parameters were:

- SOM: size = 3x3, $\rho = 0.1875$, $\sigma = 0.3$, γ = 0.1 (fixed, not adaptive);
- RSOM: size = 2x2, $\rho = 0.125$, $\alpha = 0.2$, $\sigma = 0.7$, γ = 0.01 (fixed, not adaptive);
- For the SOM-RSOM pair, the same implementation was used but with fixed $\xi_s = 1$ and $\xi_t = 0$.

Results are shown in Figure 7. They're similar to the previous experiment, except that the prior probabilities are nearly 33% (3 groups) now.

7. Conclusions and Future Works

This paper has shown how to create a more robust variant of the SOM-RSOM pair with little modification. By using the RSOM weight vector of its BMU, a good predictor for

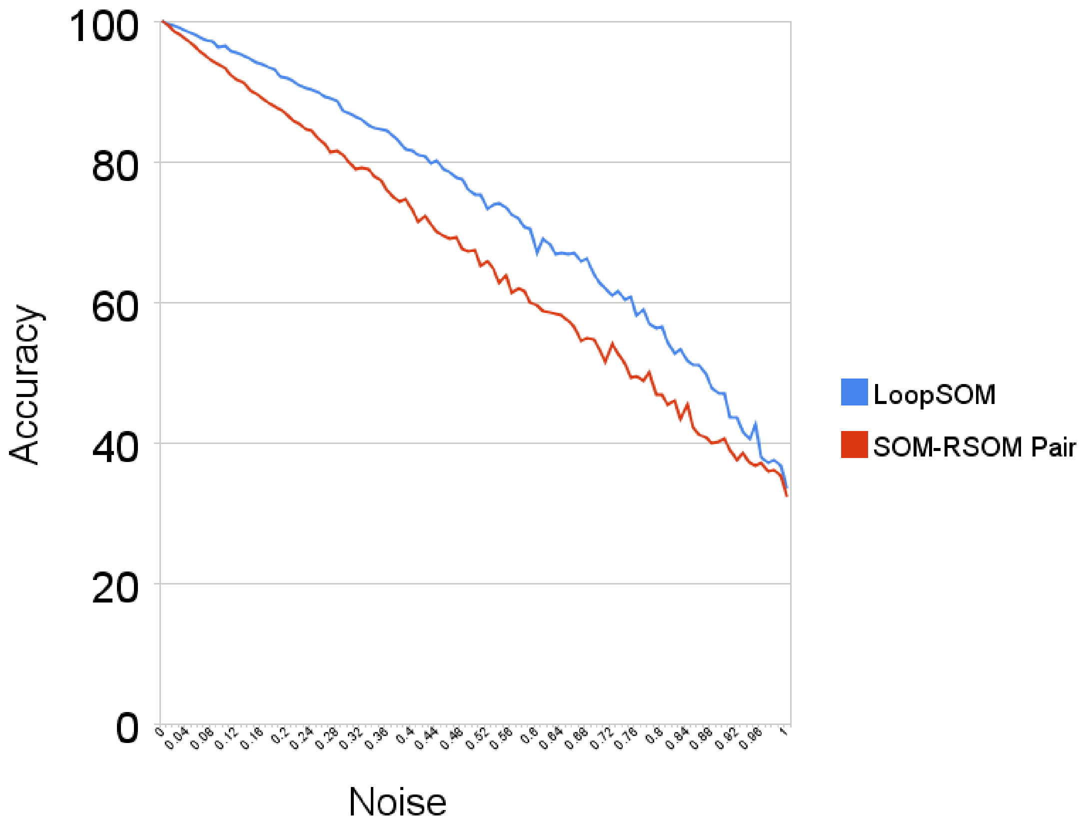

Figure 7. Results comparing the SOM-RSOM pair and the LoopSOM accuracy on a 2D line classification task.

the SOM activations is obtained. Also, it is closer to the MPF than the previous model.

There is still room for many improvements over the LoopSOM, for instance:

- analyze other possible formulas for combining the SOM current activation and the RSOM prediction, preferably with strong probabilistic principles;
- replace the current predictor (RSOM BMU weight vector) by a SRN;
- replace the RSOM or the entire SOM-RSOM pair by a Recursive SOM (RecSOM, [Voegtlin 2002]);
- use the PLSOM adaptive parameters;
- replace the SOM-RSOM pairs by LoopSOM's in a HQSOM and do some comparisons.

All of these items will be explored in future works.

References

Berglund, E. (2009). Improved PLSOM algorithm. *Applied Intelligence*, pages 1–9.

Berglund, E. and Sitte, J. (2003). The parameter-less SOM algorithm. In *Proc. ANZIIS*, pages 159–164.

Cohen, L. (1998). An information-processing approach to infant perception and cognition. *The development of sensory, motor, and cognitive capacities in early infancy*, pages 277–300.

Elman, J. (1990). Finding structure in time. *Cognitive science*, 14(2):179–211.

Foldiak, P. and Young, M. (1995). Sparse coding in the primate cortex. *The handbook of brain theory and neural networks*, pages 895–898.

Hawkins, J. (2005). *On Intelligence*. Owl Books.

Hawkins, J. and George, D. (2006). Hierarchical Temporal Memory–Concepts, Theory, and Terminology. *Whitepaper, Numenta Inc, May*.

Kohonen, T. (1998). The self-organizing map. *Neurocomputing*, 21(1-3):1–6.

Koskela, T., Varsta, M., Heikkonen, J., and Kaski, K. (1998). Temporal sequence processing using recurrent SOM. In *Knowledge-Based Intelligent Electronic Systems, 1998. Proceedings KES'98. 1998 Second International Conference on*, volume 1.

Miller, J. and Lommel, P. (2006). Biomimetic sensory abstraction using hierarchical quilted self-organizing maps. In *Proceedings of SPIE*, volume 6384, page 63840A. SPIE.

Swarup, S., Lakkaraju, K., Klementiev, A., Ray, S., Sassen, E., and Munchen, G. (2005a). An anticipatory self-organized map for robust recognition. In *Proceedings of the IASTED International conference on Artificial Intelligence and Applications (AIA'05), Innsbruck, Austria*.

Swarup, S., Lakkaraju, K., Smith, N., and Ray, S. (2005b). A Recurrent Anticipatory Self-Organized Map for Robust Time Series Prediction. In *The Workshop on Self-Organizing Maps (WSOM), Paris, France, Sep 5-8, 2005*.

Voegtlin, T. (2002). Recursive self-organizing maps. *Neural Networks*, 15(8-9):979–991.