WSOM: Building Adaptive Wavelets with Self-Organizing Maps

Marcos M. Campos and Gail A. Carpenter
Department of Cognitive and Neural Systems
Boston University, 677 Beacon Street, Boston, MA 02215
<mmcampos|gail>@cns.bu.edu

Abstract— **The WSOM (*W*avelet *S*elf-*O*rganizing *M*ap) model, a neural network for the creation of wavelet bases adapted to the distribution of input data, is introduced. The model provides an efficient on-line way to construct high-dimensional wavelet bases. Simulations of a 1D function approximation problem illustrate how WSOM adapts to non-uniformly distributed input data, outperforming the discrete wavelet transform. A speaker-independent vowel recognition benchmark task demonstrates how the model constructs high-dimensional bases using low-dimensional wavelets.**

I. INTRODUCTION

Wavelets offer an economical framework for the representation of signals, images, and functions [1], [2], [3]. Interest in wavelet theory and applications has recently accelerated with the introduction of efficient algorithms for analyzing, approximating, estimating, and compressing functions and signals. The most popular of these algorithms is the discrete wavelet transform (DWT) [4], which uses general-purpose bases that are capable of representing many different types of functions. The bases used in the DWT algorithm are especially suited for uniformly sampled data. However, for an application that seeks to estimate functions with data that might be unevenly sampled, better performance could be obtained if information extracted from the distribution of the data were to help specify the wavelet basis. This paper proposes a new architecture, WSOM (*W*avelet *S*elf-*O*rganizing *M*ap), that uses a self-organizing map to construct wavelet bases that adapt to input data distributions. In addition, the WSOM model can use low-dimensional wavelets to construct bases for high-dimensional input spaces.

In recent years, several other hybrid methods have combined wavelets and neural networks to select bases adapted to particular problems. These systems substitute wavelets for the network activation function. Some use a fixed set of wavelets and adapt the dilation and translation parameters with a gradient descent algorithm such as conjugate gradient [5], [6], [7], [8]. Others incrementally select basis functions from a dictionary of orthogonal wavelets while keeping the wavelet parameters constant [9], [10]. Although these hybrid approaches create adaptive wavelets suited to specific problems, they lack some important features of the discrete wavelet transform, such as orthogonal bases, especially for irregularly sampled input data. Also, these methods do not provide an easy way to construct wavelet bases for problems with three or more input components. Finally, the bases created by these hybrid methods are tuned to a single function. In contrast, WSOM preserves many of the desirable features of traditional wavelet bases.

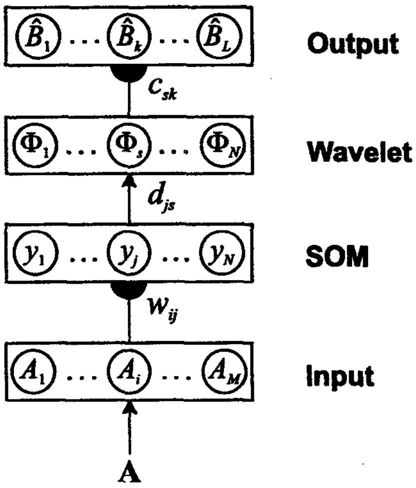
$\hat{B}_1 \cdots \hat{B}_k \cdots \hat{B}_L$
Output
c_{sk}
$\Phi_1 \cdots \Phi_s \cdots \Phi_N$
Wavelet
d_{js}
$y_1 \cdots y_j \cdots y_N$
SOM
w_{ij}
$A_1 \cdots A_i \cdots A_M$
Input
A

Fig. 1. The WSOM network

II. WSOM: WAVELET SELF-ORGANIZING MAP

WSOM is a four-layer feedforward network (Figure 1). The SOM layer quantizes the input space, mapping an input **A** onto an N-node grid via a SOM competitive learning algorithm [11], [12]. The $N \times N$ weight matrix $\mathbf{D} = (d_{js})$ then maps the SOM layer onto

0-7803-4859-1/98 $10.00©1998 IEEE

a wavelet layer. The elements of this matrix are non-adaptive and they encode the discrete wavelet function associated with each node in the wavelet layer. These weights can be easily computed using the inverse discrete wavelet transform. As SOM nodes are successively activated, the activation Φ_s of each wavelet unit plots a piecewise-constant function (a wavelet) of the input space at a given scale. These scales vary from coarse to fine. When the input space is one-dimensional, the number of wavelet units (including one scaling function unit) is the sum $1 + 1 + 2 + 4 + \ldots + 2^{\Gamma-1} = 2^{\Gamma} = N$, where Γ is the number of scales. For a 1D function approximation problem (Section III-A), the SOM layer represents N segments of the input interval $[0, 1]$. Figure 2 illustrates the sum of two scale-1 wavelet units, as A varies from 0 to 1; and the sum of eight scale-3 units.

The adaptive grid (SOM layer) and the wavelet layer together approximate a basis for the space of L^2 functions on the input space. Because of the preprocessing by the SOM layer, the wavelets used in this approximation are adapted to the distribution of the training data. Because the wavelets are defined in the grid coordinates of the SOM layer instead of the coordinates of the input space, the basis functions created with WSOM preserve many properties found in traditional discrete wavelet bases, including orthogonality. The weights c_{sk} encode the *wavelet coefficients.* The approach used in WSOM to compute these coefficients is quite different from most discrete wavelet applications. Traditionally a sampled version of the function to be approximated is stored in memory and the wavelet coefficients are then computed using the discrete wavelet transform. WSOM allows the computation of the wavelet coefficients without the need to store all observations in memory. This is accomplished by presenting each observation (or function sample) one at a time and computing the coefficients using the delta rule.

The number of dimensions of the SOM layer grid is less than or equal to the number of input components M. When the dimension of the grid is less than M, WSOM uses low-dimensional wavelets to construct bases for high-dimensional input spaces. Although SOM and WSOM systems produce the same RMS errors, WSOM has a number of advantages. WSOM requires fewer non-zero weights (c_{sk}) to represent a function. It also provides a multiresolution representation of the function and permits the use of wavelet denoising techniques to recover signal from noise.

The following algorithm implements the WSOM network during training.

Variables:
$\mathbf{A} \equiv (A_1 \ldots A_i \ldots A_M)$ is the input vector
$\mathbf{y} \equiv (y_1 \ldots y_j \ldots y_N)$ is the vector of activities of the SOM layer
$\mathbf{\Phi} \equiv (\Phi_1 \ldots \Phi_s \ldots \Phi_N)$ is the vector of activities of the wavelet layer
$\hat{\mathbf{B}} \equiv (\hat{B}_1 \ldots \hat{B}_k \ldots \hat{B}_L)$ is the vector of network outputs
$\mathbf{w}_j \equiv (w_{1j} \ldots w_{ij} \ldots w_{Mj})$ is the weight vector from the input layer to the j^{th} unit in the SOM layer
$\mathbf{d}_s \equiv (d_{1s} \ldots d_{js} \ldots d_{Ns})$ is the non-adaptive weight vector from the SOM layer to the s^{th} unit in the wavelet layer
$\mathbf{c}_k \equiv (c_{1k} \ldots c_{sk} \ldots c_{Nk})$ is the weight vector from the wavelet layer to the k^{th} unit in the output layer
$\mathbf{X}_j$ is the position, on an integer-valued grid, of the j^{th} unit in the SOM layer

Parameters:
α is the learning rate for the weights c_{sk}
β is the learning rate for the weights w_{ij}
σ is the neighborhood size used in the SOM algorithm
J is the index of the winning unit at the SOM layer
h_{Jj} modulates the amount of learning for the jth unit in the SOM layer, decreasing exponentially with distance to the J^{th} unit
w_- and w_+ are the lower and upper bounds for the initial weights w_{ij}
β_0 is the initial value for β
β_1 is the final value for β
σ_0 is the initial value for σ
σ_1 is the final value for σ
t_1 is the number of training set inputs needed for β and σ to decrease to β_1 and σ_1
n is the total number of training set inputs

In all simulations below: SOM is a 1D grid with $N = 64$, $w_- = -0.001$, $w_+ = 0.001$, $\alpha = 0.1$, $\beta_0 = 0.5$, $\beta_1 = 0.01$, and $\sigma_1 = 0.1$.

Algorithm:

```
0.  Set $t = 1$, distribute weights $w_{ij}$ uniformly in $[w_-, w_+]$, and set all $c_{sk} = 0$
1.  Decrease $\beta$:
        $\beta = \begin{cases} \beta_0(\beta_1/\beta_0)^{\frac{t-1}{t_1-1}} & \text{if } 1 \leq t < t_1 \\ \beta_1 & \text{if } t \geq t_1 \end{cases}$
2.  Decrease $\sigma$:
        $\sigma = \begin{cases} \sigma_0(\sigma_1/\sigma_0)^{\frac{t-1}{t_1-1}} & \text{if } 1 \leq t < t_1 \\ \sigma_1 & \text{if } t \geq t_1 \end{cases}$
3.  Get the $t^{\text{th}}$ input vector $\mathbf{A}$ and output vector $\mathbf{B}$
4.  Find the winning SOM unit:
        $J = \arg\min_j ||\mathbf{A} - \mathbf{w}_j||$
```

```
5.  Compute the activity of the SOM layer:
    $y_J = 1$ and $y_j = 0, j \neq J$
6.  Compute the activity of the wavelet layer:
    $\mathbf{\Phi}_s = \mathbf{D}\mathbf{y}' = d_{Js}$
7.  Compute the output: $\hat{B}_k = \sum_{s=1}^{N} c_{sk}\Phi_s$
8.  Adjust $c_{sk}$ according to: $\Delta c_{sk} = \alpha\Phi_s(B_k - \hat{B}_k)$
9.  Set: $h_{Jj} = \exp(-\|\mathbf{X}_j - \mathbf{X}_J\|^2/\sigma^2)$
10. Adjust $w_{ij}$ according to: $\Delta w_{ij} = \beta h_{Jj}(A_i - w_{ij})$
11. If $t = n$ then stop. Else add 1 to $t$ and go to 1
```

During testing the same algorithm is applied with $\alpha = 0$, $\beta = 0$, and output $\hat{\mathbf{B}}$ for all inputs $\mathbf{A}$. If the task is categorical, the maximum $\hat{B}_k$ value chooses the output class.

III. SIMULATIONS

This section illustrates WSOM's capabilities with a 1D function approximation task and a speaker-independent vowel recognition task. The discrete wavelet transform implemented by the weights d_{js} from the SOM layer to the wavelet layer is computed using [13].

A. 1D FUNCTION APPROXIMATION

This task is to estimate a fifth-order chirp function (Figure 2). Simulations consider five different input probability distributions $p_\lambda(x)$. Each distribution takes the form:

$$p_\lambda(x)dx = \begin{cases} x^\lambda dx & 0 < x < 1 \\ 0 & \text{otherwise} \end{cases}$$

where $p_\lambda(x)dx$ is the probability of generating a number between x and $x + dx$ and λ is a nonnegative integer. In simulations λ varied from 0 (uniform distribution) to 4 (equal to the degree of the chirp function frequency). Larger values of λ bias the input distribution towards higher input values in the interval $[0, 1]$. The training set for each distribution consisted of 7000 input points A in the interval [0,1], with output $B = 0.5 + 0.5\sin(\omega(A)A)$, $\omega(A) = 40\pi A^4$.

In order to illustrate the advantages of adaptive wavelets for non-uniformly sampled data, performance comparisons were made between the discrete Haar wavelet basis (DWT) and an adaptive Haar wavelet basis (WSOM). Performance was measured by computing the root mean squared error RMSE$= \sqrt{\frac{1}{n}\sum_{t=1}^{n}(\mathbf{B}(t) - \hat{\mathbf{B}}(t))^2}$ on a test set containing $n = 3000$ observations drawn from the same distribution as the training set. The simulations used $\sigma_0 = 20$ and $t_1 = 2000$ for all distributions.

Table I summarizes network performance for the different distributions for two stages in the training process. As the input distribution moves away from the uniform case (λ increases) WSOM's performance compared to DWT improves considerably. For less uniform distributions the adaptive wavelet basis selected by WSOM displays increased resolution in areas where the input density is larger (Figure 2).

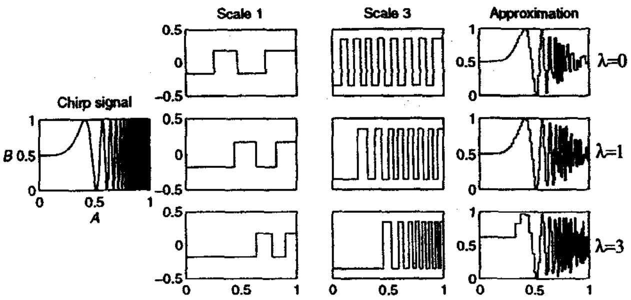

Fig. 2. Function approximation after training WSOM with 7000 observations for simulation input distributions with $\lambda = 0, 1, 3$. Scale 1 graphs show the sum of two coarse-scale wavelet node activations for inputs $A \in [0, 1]$; and scale 3 graphs show the sum of eight finer-scale wavelet node activations. The wavelets for $\lambda = 0$ (uniform distribution) are close to those for the discrete wavelet transform.

TABLE I
ROOT MEAN SQUARED ERROR FOR THE CHIRP TASK WITH DIFFERENT INPUT DISTRIBUTIONS (INDEXED BY λ) AND WITH 3000 AND 7000 TRAINING SET INPUTS.

	RMS Error			
	3000 Observations		*7000 Observations*	
λ	*DWT*	*WSOM*	*DWT*	*WSOM*
0	0.199	0.185	0.192	0.179
1	0.261	0.214	0.255	0.209
2	0.295	0.230	0.291	0.228
3	0.312	0.235	0.305	0.229
4	0.330	0.220	0.331	0.218

B. VOWEL RECOGNITION

Performance of WSOM was compared to the performance of seven other models, reported by [14], on a speaker-independent vowel recognition task. The vowel examples were collected by [15], who recorded eleven steady-state English vowels from 15 speakers, 7 female and 8 male. A word containing each vowel was spoken once by each speaker. The speech signals were low-pass filtered at 4.7 kHz and then digitized to 12 bits with a 10 kHz sampling rate. Twelfth-order linear predictive analysis was carried out on six 512-sample Hamming windowed segments from the steady part of the vowel, and the reflection coefficients were used to calculate 10 log area parameters, giving a 10-dimensional input space. Each speaker thus yielded six samples of speech from each of the eleven vowels. The data were partitioned into 528 samples for training, from four male and four female speakers, and 462 samples for testing, from the remaining four male and three female speakers. The data set is archived in the CMU connectionist benchmark collection [16].

TABLE II

PERFORMANCE ON THE TEST SET FOR THE SPOKEN VOWEL CLASSIFICATION PROBLEM. WITH THE EXCEPTION OF THOSE FOR WSOM, RESULTS ARE FROM [14]. RESULTS FOR WSOM ARE FOR 70 TRAINING EPOCHS AND *Units* STANDS FOR NUMBER OF HIDDEN UNITS.

Classifier	*Units*	*% Correct*
Single-layer perceptron	-	33
Multi-layer perceptron	88	51
Modified Kanerva model	528	50
Radial Basis Function	528	53
Gaussian node network	528	55
Square node network	88	55
Nearest neighbor	528	56
WSOM	**64**	**55**

WSOM was trained during 70 epochs. The simulation used $\sigma_0 = 10$ and $t_1 = 1000$. Although WSOM used a completely unsupervised approach for placing the hidden units in feature space, its results were comparable to the best results reported by Robinson for other supervised classifiers (Table II). WSOM also used fewer hidden units.

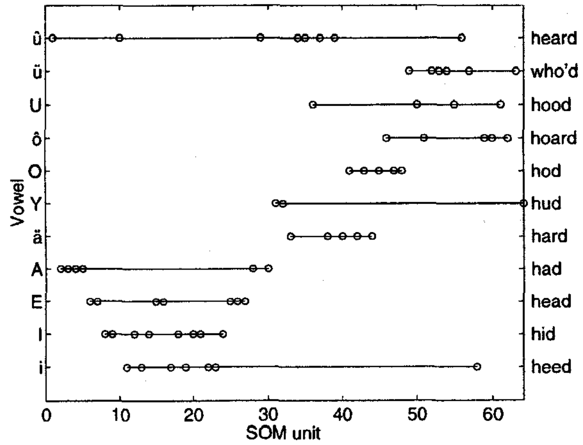

Fig. 3. Decision regions for the different vowels.

This example shows that WSOM can construct a high-dimensional wavelet basis from 1D wavelets. One advantage of WSOM compared to other approaches that combine wavelets and neural networks is the data visualization capability inherited from the SOM algorithm. Because the SOM layer is usually implemented as a one- or two-dimensional grid, it can represent the structure of high-dimensional data in graphical form. For the vowel recognition task, each SOM unit in the 1D grid represents a region in the ten-dimensional input space, and each unit has associated with it a prototypical input vector, equal to the centroid of the region. In Figure 3, circles plot which vowel is most frequently associated with a given region, projected onto an SOM unit. The horizontal bars indicate how spread out a given vowel (output class) is in the feature space. Figure 3 shows that similar vowels are grouped together and that the regions in feature space associated with similar vowels overlap significantly.

IV. CONCLUSION

WSOM is a neural network model for building wavelets that are capable of adapting to non-uniformly distributed data and constructing high-dimensional wavelet bases from low-dimensional components. This new approach can be implemented on-line and has good data visualization capabilities. The primary contribution of the model is the use of a self-organizing map to implement a coordinate transformation from the input space to a regular grid. Basis functions (wavelets) are then defined on the new coordinate system (grid). This method can also be adapted to other basis functions, such as gaussians.

Acknowledgments

This research was supported in part by the National Science Foundation (NSF IRI 94-011659) and the Office of Naval Research (ONR N00014-95-10409 and ONR N00014-95-0657).

REFERENCES

[1] A. Grossman and J. Morlet, "Decomposition of Hardy functions into square integrable wavelets of constant shape," *SIAM Journal of Mathematical Analysis*, vol. 15, pp. 723–736, 1984.

[2] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Communications in Pure and Applied Mathematics*, vol. 41, pp. 909–996, 1988.

[3] Y. Meyer, *Ondelettes et Opérateurs*. Paris: Hermann, 1990.

[4] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Learning*, vol. 11, pp. 674–693, 1989.

[5] H. Szu, B. Telfer, and S. Kadambe, "Neural network adaptive wavelets for signal representation and classification," *Journal of Optical Engineering*, vol. 31, no. 9, pp. 1907–1916, 1992.

[6] Q. Zhang and A. Benveniste, "Wavelet networks," *IEEE Transactions on Neural Networks*, vol. 3, no. 6, pp. 889–898, 1992.

[7] Y. Patti and P. Krishnaprasad, "Analysis and synthesis of feedforward neural networks using discrete affine wavelet transforms," *IEEE Transactions on Neural Networks*, vol. 4, no. 1, pp. 73–85, 1993.

[8] J. Echauz and G. Wachtsevanos, "Elliptic and radial wavelet neural networks," in *Proc. Second World Automation Congress*, vol. 5, pp. 173–179, Montpellier, France. TSI Press, 1996.

[9] B. R. Bakshi and G. Stephanopoulos, "Wave-net: A multi-resolution, hierarchical neural network with localized learning," *AIChE Journal*, vol. 39, no. 1, pp. 57–81, 1993.

[10] T. I. Boubez and R. L. Peskin, "Multiresolution neural networks," in *SPIE*, vol. 2242, pp. 637–660, 1994.

[11] S. Grossberg, "Adaptive pattern classification and universal recoding, I: Parallel development and coding of neural feature detectors," *Biological Cybernetics*, vol. 23, pp. 121–134, 1976.

[12] T. Kohonen, *Self-Organization and Associative Memory*. New York: Springer-Verlag, second ed., 1988.

[13] Wavelab v.701, "Public domain MATLAB toolbox." Available through anonymous ftp: playfair.stanford.edu/pub/wavelab, 1996.

[14] A. J. Robinson, *Dynamic Error Propagation Networks*. Ph.D. dissertation, Cambridge University, 1989.

[15] D. H. Deterding, *Speaker Normalization for Automatic Speech Recognition*. Ph.D. dissertation, Cambridge University, 1989.

[16] S. E. Fahlman, *CMU benchmark collection for neural net learning algorithms*. Pittsburgh: Carnegie Mellon University, School of Computer Science [machine-readable data repository], 1993.