Hidden Markov Models for Wavelet-Based Signal Processing

Matthew S. Crouse and *Richard G. Baraniuk* *

Dept. of Electrical and Computer Engineering
Rice University
Houston, TX 77251

Robert D. Nowak

Dept. of Electrical Engineering
Michigan State University
East Lansing, MI 48824

Abstract

Current wavelet-based statistical signal and image processing techniques such as shrinkage and filtering treat the wavelet coefficients as though they were statistically independent. This assumption is unrealistic; considering the statistical dependencies between wavelet coefficients can yield substantial performance improvements. In this paper, we develop a new framework for wavelet-based signal processing that employs hidden Markov models to characterize the dependencies between wavelet coefficients. To illustrate the power of the new framework, we derive a new signal denoising algorithm that outperforms current scalar shrinkage techniques.

1 Introduction

Wavelets have emerged as an exciting new tool for statistical signal and image processing. The wavelet transform is an atomic decomposition that represents a signal $z(t)$ in terms of shifted and dilated versions of a prototype bandpass wavelet function $\psi(t)$. For special choices of the wavelet, the atoms

$$\psi_{j,k}(t) \equiv 2^{-j/2}\,\psi\big(2^{-j}t-k\big), \quad j,k \in \mathbf{Z}$$

form an orthonormal basis, and we have the signal representation [1]

$$z(t) = \sum_{j,k\in\mathbf{Z}} w_{j,k}\,\psi_{j,k}(t), \quad w_{j,k} = \int z(t)\,\psi^*_{j,k}(t)\,dt.$$

For a wavelet centered at time t_0 and frequency f_0, the *wavelet coefficient* $w_{j,k}$ measures the content of the signal around the time $2^j t_0$ and frequency $2^{-j} f_0$ (see Figure 1). To analyze images, we employ 2-d wavelet systems. (Since the indexing system for higher dimensional wavelets rapidly gets out of hand, we will adopt an abstract single index system for wavelet atoms and coefficients: $\psi_{j,k} \to \psi_i$, $w_{j,k} \to w_i$.)

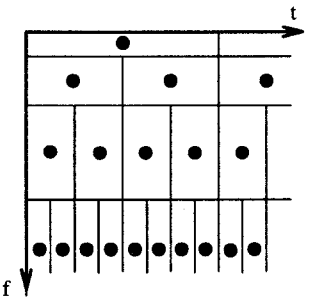

Figure 1. *Tiling of the time-frequency plane by the atoms of the wavelet transform. Each box depicts the idealized support of an atom ψ_i in time-frequency; the solid dot at the center corresponds to the wavelet coefficient w_i. Each different row of wavelet atoms corresponds to a different* scale or frequency band. *(We run the frequency axis down rather than up for later convenience.)*

The joint time-frequency analysis effected by the wavelet transform has some attractive properties that make it natural for statistical applications, including estimation [2, 3, 4], detection, and classification. We call these the *primary properties* of the wavelet transform:

Locality: Each wavelet atom ψ_i is localized simultaneously in time and frequency. Therefore, wavelets can match a wide range of different signal components, from transients to tones.

Multiresolution: Wavelet atoms compress and dilate to analyze at a nested set of scales. This allows the transform to match both short-duration and long-duration signal structures.

Compression: The wavelet transforms of real-world signals and images tend to be very sparse. This is due to the fact that wavelets form unconditional bases for most of the key function spaces [2].

Attention has focussed on *scalar* processing of the wavelet coefficients w_i. For example, independent,

*This work was supported by the National Science Foundation, grant no. MIP–9457438, and the Office of Naval Research, grant no. N00014–95–1–0849.

Email: mcrouse@rice.edu, richb@rice.edu, rnowak@egr.msu.edu
Web: http://www-dsp.rice.edu, http://www.egr.msu.edu/spc/

coordinate-wise thresholding of the wavelet coefficients has been demonstrated to suppress white Gaussian noise from a large class of real-world signals [2]. Scalar wavelet processing algorithms are based on the primary properties above plus an interpretation of the transform as a "decorrelator" that attempts to make each wavelet coefficient statistically independent of all others. If this were possible for all signals and images, then simple scalar processing in the wavelet domain would be optimal.

However, the wavelet transform cannot completely decorrelate real-world signals and images — a *residual dependency structure* always remains between the wavelet coefficients. In words, we have following *secondary properties* of the wavelet transform:

Clustering: If a particular wavelet coefficient is large/small, then adjacent coefficients are very likely to also be large/small.

Persistence across Scale: Large/small values of wavelet coefficients tend to propagate across scales.

Both of these empirical observations have been exploited with tremendous success by the compression community [5]. Our goal is to do the same for signal processing.

In this paper, we introduce the concept of *probabilistic graph models*, for characterizing the dependencies between the coefficients of the wavelet transform. Our marriage of wavelet transforms and these Hidden Markov models yields a flexible framework for statistical signal and image processing that both matches the properties of the wavelet transform and exploits the structure inherent in real-world signals and images. This framework provides a natural setting for signal estimation, detection, classification, and even synthesis.

Our modeling procedure for wavelet transforms will consist of two steps. First, in Section 2, we present a probabilistic model for an individual wavelet coefficient. Following [4], we employ a Gaussian mixture model. Then, in Section 3, we characterize the inter-coefficient dependencies using Hidden Markov models. We discuss a new Expectation Maximization (EM) algorithm for training the models on real (signal+noise) data in Section 4. In Section 5, we apply this powerful machinery to signal estimation and derive a new wavelet de-noising scheme that performs substantially better than current scalar approaches. We close in Section 6 with a discussion and conclusions.

2 Probabilistic Model for an Individual Wavelet Coefficient

Recall the compression property of the wavelet transform. The transform of a typical signal or image consists of a small number of large coefficients and a large number of small coefficients. Thus we can roughly model each coefficient as being in one of two states: "high" or "low." If we associate with each state a probability density — say a high-variance, zero-mean density for the "high" state and a low-variance, zero-mean density for the "low" state — the result is a two-state mixture model for each wavelet coefficient.

In general, an M-state mixture model for a random variable X consists of [1]

1. a discrete random state variable S taking the values $s \in 1, 2, \ldots, M$ with pmf $p_S(s)$, and
2. the conditional pdfs $f_{X|S}(x|S = s)$, $s \in 1, 2, \ldots, M$.

Typically, the conditional densities are chosen to be Gaussian or, more generally, a member of the exponential family of distributions [6]. To generate a realization of X using the model, we first generate a state value $S = s_0$ using $p_S(s)$ and then the value $X = x_0$ using the density $f_{X|S}(x|S = s_0)$. The pdf of X is given by

$$f_X(x) = \sum_{m=1}^{M} p_S(m)\, f_{X|S}(x|S = m).$$

In most applications of mixture models, the value $X = x_0$ is observed, but the value of the state variable S is not; we say that the value of S is *hidden*.

In this paper, we will model each wavelet coefficient as a random variable W_i with a two-state (zero-mean) Gaussian mixture density. Empirically, this model has proven both effective and convenient [3, 4]. As we see from Figure 2, this simple model is completely parameterized by the pmf of the state variable S_i, $p_{S_i}(1)$, $1-p_{S_i}(1)$, and the variances of the Gaussian pdfs corresponding to each state, $\sigma^2_{i,1}, \sigma^2_{i,2}$. In Figure 3, we demonstrate the close fit of this model with the distribution of wavelet coefficients of an actual signal. By increasing the number of states $M > 2$ and employing more general conditional pdfs, the fit can be made arbitrarily close.

3 Probabilistic Models for a Complete Wavelet Transform

Since a Gaussian mixture model can accurately characterize the pdf of a single wavelet coefficient, it seems logical to use Gaussian mixture models to characterize the joint pdf of the entire wavelet transform. The simplest approach would be to model the coefficients as independent with identical mixture distributions within each scale. We call

[1] Notation: We use $p_S(s)$ to denote the probability mass function (pmf) of the discrete random variable S. We use $f_X(x)$ to denote the probability density function (pdf) of the continuous random variable X.

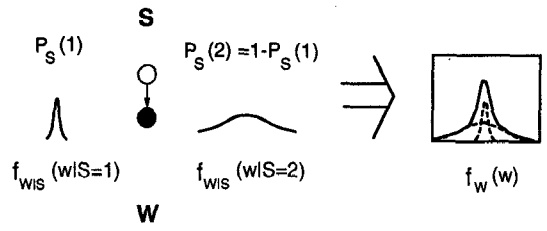

Figure 2. *A two-state Gaussian mixture model for a random variable W. We denote the state variable S with a white dot, the random variable W with a closed dot. Illustrated are the Gaussian conditional pdf's for $W|S$ as well as the overall mixture pdf for W. In our application, we model each wavelet coefficient W_i (each black dot in Figure 1) in this way.*

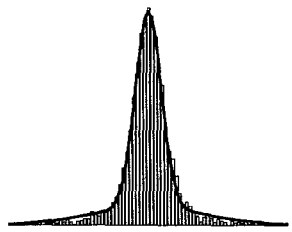

Figure 3. *A two-state Gaussian mixture model can closely fit real wavelet coefficient data. Here we compare the model pdf to the histogram of one scale of the wavelet transform of an image of fruit.*

this approach the *Independent Mixture Model*. As demonstrated by the de-noising results in [3, 4], the Independent Mixture Model is a substantial improvement over deterministic signal models.

Nevertheless, clustering and persistence of signal energy in the wavelet domain leads to local dependencies between wavelet coefficients. Characterization of these dependencies has resulted in significant performance gains in compression [5]. Ideally, we would like a model that both matches each individual coefficient's pdf and captures dependencies between coefficients.

We motivate our approach by extending the Gaussian mixture model for one wavelet coefficient to jointly model two adjacent wavelet coefficients. By the Clustering and Persistence properties, if one coefficient is in a high-variance (low-variance) state, then the other is very likely also in a high-variance (low-variance) state. Thus, the two adjacent wavelet coefficients can be modeled as Gaussian mixtures with *interdependent state variables*. This two-coefficient example suggests a natural generalization to the N coefficients in a wavelet transform: model each coefficient as a Gaussian mixture, but allow probabilistic dependencies between the state variables of each mixture.

What remains is to specify an appropriate model for these dependencies between the state variables. A complete joint pdf taking into account all possible dependencies is clearly intractable, since the number of different state variable combinations grows exponentially in the number of wavelet coefficients. Furthermore, in applications, the state variables will be hidden. Fortunately, we have the wavelet transform Locality and Multiresolution properties on our side. We propose to use probabilistic graph theory [7, 8] to both account for unobserved state variables and to focus in on the most relevant dependencies.

3.1 Probabilistic graph models

Probabilistic graphs, also known as Hidden Markov models, are convenient representations for modeling interdependent random variables. Graphs characterize the relevant local dependencies between random variables, while avoiding the clutter and intractability of a global joint probability model. To keep things clear, we will provide a simplified, intuitive sketch of the theory. Much more precise, complete treatments can be found in [7, 8].

A probabilistic graph models the joint statistics of N random variables $X_1, X_2, \ldots, X_N$ using N nodes and connections between nodes. A graph associates each X_i with a node i. Dependency between variables X_i and X_j is described by connecting nodes i and j. If nodes i and j are left unconnected, then X_i and X_j are independent. If i and j are not directly connected, then X_i and X_j are conditionally independent given the values of the random variables of the nodes that "separate" them. A set of nodes G separates i and j if all paths between i and j contain a node in G. The most obvious example of conditional independence and separation is the basic property of the temporal Markov-1 chain: given the present, the past and the future are independent.

Note that the separation property leads to global statistical modeling via simple local interactions between coefficients. For a random variable X_i, information about all other random variables is summarized by a local "neighborhood" of X_i — the random variables that are directly connected to X_i. This locality property is crucial for efficient signal processing using graphs.

3.2 Graph models for wavelet transforms

We seek a model that characterizes wavelet coefficients as Gaussian mixtures with mutually dependent hidden state variables. Using the wavelet Locality property, we expect the state variables to have local dependencies. Since graphs are efficient at expressing local dependencies, we use graphs for modeling the coefficients of a wavelet transform.

The Locality and Multiresolution properties of the wavelet transform suggest three simple ways to "connect the dots" representing the wavelet coefficients in Figure 1: (1) a graph with no dependencies between wavelet state variables, (2) a graph linking wavelet state variables across time using chains, and (3) a graph linking wavelet state variables across scale using trees. In Figure 4, we illustrate these three simple graphs.

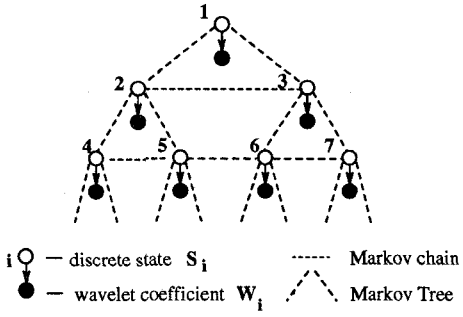

Figure 4. *Probabilistic graphs for modeling the statistical dependencies of the coefficients of a wavelet transform. Each black node represents a continuous wavelet coefficient W_i. Each white node represents the (hidden) mixture state variable S_i for W_i. Removing all dashed connections corresponds to the Independent Mixture Model. Connecting discrete nodes horizontally across time yields the Hidden Markov Chain Model. Connecting discrete nodes vertically across scale yields the Hidden Markov Tree Model.*

A theory exists for analyzing more complicated graphs [8], such as those obtained by linking hidden state variables across both time and scale. Since the analysis is beyond the scope of this paper, we focus more closely on the three simple graphs.

Independent Mixture Model: Removing all connections between state variables S_i in Figure 4 leads to the Independent Mixture Model presented in [3, 4] and discussed above. It treats wavelet state variables (and hence wavelet coefficients) as independent random variables.

Hidden Markov Chain Model: Connecting the state variables S_i horizontally in Figure 4 specifies a Markov chain dependency between the state variables *within each scale* [9]. This new model treats wavelet state variables as dependent within each scale, but independent from scale to scale.

Hidden Markov Tree Model: By connecting state variables vertically *across scales* in Figure 4, we obtain a graph with tree-structured dependencies between state variables. We call this new model the *Hidden Markov Tree Model* to emphasize the underlying dependencies between the wavelet state variables.

The Hidden Markov Tree Model matches both the Clustering and Persistence across Scale properties of the wavelet transform. Its structure is reminiscent of the zerotree wavelet compression system [5], which exploits tree-structured dependencies for substantial compression gains.

The Hidden Markov Tree Model has a natural parent-child dependency interpretation, which is defined formally by a directed tree graph [7, 8]. State variable dependencies are modeled via state transition probabilities from each parent state variable S_i to its "children," the two state variables connected to it from below (if they exist). For example, In Figure 4, state variables S_4 and S_5 are both children of S_2, and hence causally dependent on S_2. Dependency is not simply limited to parent-child interactions, however. State variables S_4 and S_5 may be highly dependent due to their joint interaction with S_2.

Let $S_{\rho}(i)$ denote the parent of S_i. Using a zero-mean Gaussian mixture model for each wavelet coefficient value W_i, the parameters for the Hidden Markov Tree Model are:

1. $p_{S_1}(m)$, the pmf for the root node S_1.
2. $\epsilon^{mr}_{i,\rho(i)} = p_{S_i|S_{\rho(i)}}(m|S_{\rho(i)} = r)$, the probability S_i is in state m given $S_{\rho(i)}$ is in state r.
3. $\sigma^2_{i,m}$, the variance of the wavelet coefficient W_i given S_i is in state m.

4 Model Training

We have defined three probabilistic graphs for capturing the structure in a wavelet transform. To use these graphs for signal processing, two operations are of interest:

Training: Given a set of training data, estimate the model parameters to achieve a maximum-likelihood (ML) fit.

Likelihood determination: Given a fixed model, calculate the probability of the observed wavelet data using the model.

Training is fundamental to any application. Once we have trained the model on a signal or class of signals, we can apply it to tasks such as estimation, classification, prediction (useful for compression), and synthesis. Likelihood determination not only is useful for tasks such as detection and classification, but also is a key component of training.

We train our models by choosing parameters that maximize the likelihood (or probability) of the training data given our model. These parameters are the state probabilities and conditional Gaussian variances. Unfortunately, the fact that we cannot observe the hidden state variables means that closed-form parameter estimates are unobtainable. We circumvent this obstacle using the Expectation Maximization (EM) algorithm.

4.1 EM algorithm

The EM algorithm is an iterative procedure for performing ML estimation in problems with incomplete data [10]. In our case, the unobserved state variables S_i are the incomplete data. Starting from an initial set of parameter estimates, the EM algorithm iterates between the following two steps:

Expectation: Estimate the likelihood (or probability) of the data given the current parameters.

Maximization: Select new parameters to maximize the estimated likelihood of the data.

For all three graphs discussed in Section 3.2, it can be shown the EM algorithm converges to a local maximum of the likelihood function [6, 9, 11]. Moreover, for these graphs, the Expectation step is equivalent to likelihood determination. We next investigate the specific expectation and maximization steps for the three different graphs.

Independent Mixture Model: The expectation step is Bayes rule. The maximization step is a weighted averaging of statistics. Both steps have simple closed-form expressions [6].

Hidden Markov Chain Model: The expectation step is the forward-backward algorithm [9]. Maximization is again a weighted averaging. The EM algorithm specialized to this case is known as the Baum-Welch algorithm [9].

Hidden Markov Tree Model: The expectation step essentially corresponds to the *upward-downward algorithm* for tree graphs derived in [11]. However, while [11] deals entirely with discrete variables, our tree has both discrete state variables (S_i) and continuous wavelet coefficients (W_i). Therefore, we must modify the upward-downward algorithm to allow the observed data to be continuous-valued. The maximization step is again a simple weighted averaging. Details of both steps are provided in [12].

4.2 Robust training

A key problem with training Hidden Markov models is that multiple iid observations of the entire set of wavelet coefficients are required for reliable model estimation. Often, however, only a single realization is observed.

Tying is the process of averaging over data expected to be statistically similar [9]. In the Independent Mixture Model, our assumption that the wavelet coefficients within each scale can be described by the same mixture density allows us to tie together all wavelet coefficients at the same scale and average their statistics.

Many more opportunities exist for tying in the Hidden Markov Tree Model. For example, by terminating the wavelet decomposition before reaching the lowest frequencies, the wavelet tree breaks into a *forest* of wavelet trees. In Figure 5(a) we tie across trees and average across nodes at identical locations in each tree. In Figure 5(b), we tie within a tree and averaged across nodes within a tree that are assumed statistically similar.

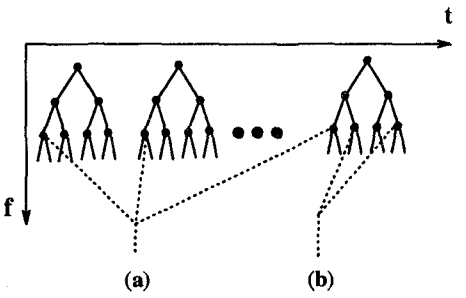

Figure 5. *Tying in the Hidden Markov Tree Model. (a) Tying across trees. (b) Tying within a tree.*

If the case where we have only one signal realization, we tie both across trees and within trees to estimate the parameters of the Hidden Markov Tree Model on a scale-by-scale, rather than a coefficient-by-coefficient, basis. This procedure is clearly less signal adaptive, but leads to more robust parameter estimates.

5 Application to Signal Estimation

To illustrate the power of the wavelet-Markov signal model, we will apply it to the problem of signal estimation in additive white Gaussian noise. Since the orthonormal wavelet transform of zero-mean white Gaussian noise of variance σ_n^2 is also zero-mean white Gaussian noise of variance σ_n^2, the estimation problem can be expressed in the wavelet domain as

$$w_i = \theta_i + n_i, \tag{1}$$

where w_i, θ_i, and n_i denote the wavelet coefficients of the observed data, the signal, and the noise, respectively. Our approach is to first estimate the statistical structure of the θ_i from the noisy data, and then use this structure as a prior to obtain a mean-square-error (MSE) optimal conditional mean estimate of the θ_i (and, hence, the signal).

We assume that only a single realization of the data is available. To stabilize our estimates, we terminate the wavelet expansion at K wavelet subtrees. (This leaves several low frequency wavelet coefficients, which we do not alter.) Denoting the data in each subtree with a superscript k, we can rewrite (1) as

$$w_i^k = \theta_i^k + n_i^k, \qquad k = 1, 2, \ldots, K.$$

For the graph structures considered above, the addition of the white Gaussian noise n_i^k to the θ_i^k has no effect other than increasing the mixture model variances $\sigma_{i,m}^2$ by σ_n^2. Hence, we obtain the parameters of the Hidden Markov Tree representing the θ_i coefficients in two steps: First, we estimate the parameters of the Hidden Markov Tree representing the observed w_i^k coefficients using the EM algorithm from Section 4.1 (for details, see [12]). Then, we decrease the mixture model variances by σ_n^2. We estimate the noise power σ_n^2 via the median estimate of [2] peformed on the finest scale wavelet coefficients (where the signal energy is expected to be negligible).

As in [4], we assume that the θ_i^k have zero mean. Therefore, if the state S_i^k corresponding to each signal wavelet coefficient θ_i^k is known, our estimation problem becomes one of estimating a zero-mean Gaussian random variable in zero-mean additive Gaussian noise, in which case the MSE-optimal conditional mean estimate is given by

$$E[\Theta_i^k | W_i^k = w_i^k, S_i^k = m] = \frac{\sigma_{i,m}^2}{\sigma_n^2 + \sigma_{i,m}^2} w_i^k. \quad (2)$$

We apply the forward-backward algorithm to find the conditional probability $P(S_i^k = m|O)$, with O denoting the noisy signal training data (for details, see [12]). The conditional mean estimate for each signal wavelet coefficient is then obtained using the chain rule for conditional expectation

$$E[\Theta_i^k|O] = \sum_m P(S_i^k = m|O) \frac{\sigma_{m,i}^2}{\sigma_n^2 + \sigma_{m,i}^2} w_i^k.$$

The inverse wavelet transform of these estimated signal wavelet coefficients gives us the final signal estimate.

Table 1 compares the estimation performance of the Independent Mixture and the Hidden Markov Tree models with two state-of-the-art scalar algorithms. Donoho's SureShrink algorithm [2] performs scalar soft thresholding in the wavelet domain. Chapman, Kolaczyk, McCulloch's Bayesian mixture algorithm [4] bases an estimate similar to (2) on independent Gaussian mixture models. We provide the MSE results for de-noising Donoho's test signals Bumps, Blocks, Doppler, and Heavisine [2] in additive white Gaussian noise of power $\sigma_n^2 = 1$.[2]

The Independent Mixture Model algorithm resembles the Bayesian algorithm and offers similar performance.

[2]For each estimation algorithm, Bumps was transformed using the Daubechies-4 wavelet, Blocks using the Haar wavelet, and Doppler and Heavisine using the Daubechies-8 most-nearly-symmetric wavelet. The SureShrink and Bayesian algorithms used the maximum possible number of wavelet decomposition levels (within the resolution limits of the wavelet filter). The IMM and Markov tree algorithms used a seven-level wavelet decomposition. Error results for all signals were obtained by averaging over 1000 trials.

Table 1. *De-noising results for Donoho's test signals [2].*

Method	Mean-squared error			
	Bump	Block	Dopp	Hsine
SureShrink [2]	0.683	0.222	0.228	0.095
Bayesian [4]	0.350	0.099	0.165	0.087
IMM	0.335	0.105	0.170	0.080
Markov Tree	0.268	0.079	0.132	0.081

Since both algorithms assume independence between signal wavelet coefficients, they serve as a benchmark for analyzing the effect of exploiting wavelet-domain dependency. Only the Hidden Markov Tree algorithm exploits dependencies in the wavelet decomposition. Comparing the Hidden Markov Tree and the benchmark algorithms, we observe that significant gains can be achieved by exploiting wavelet-domain dependencies. (The only exception is the Heavisine signal. Its lack of high-frequency signal content limits the performance of the Hidden Markov Tree approach.)

Using the zerotree structure of [5], we can apply a Hidden Markov Tree Model to the wavelet transforms of images. Preliminary results indicate that exploiting 2-d dependencies provides similar performance improvements for image estimation.

6 Conclusions

The primary properties of the wavelet transform — Locality, Multiresolution, and Compression — have lead to a new approach to statistical signal processing based on simple scalar processing of the wavelet coefficients. However, the wavelet transforms of real-world signals and images have a residual structure that can be exploited to improve the performance of these algorithms. In this paper, we have modeled the dependencies between wavelet coefficients that stem from the secondary properties of the wavelet transform — Clustering and Persistence across Scale. We can interpret our approach in the following way: The wavelet transform "almost decorrelates" the signal, removing all but the most local dependencies for the probabilistic graph model to handle. It is the fact that the wavelet transform can almost decorrelate so many signals that makes our approach feasible.

We feel that the graph-theoretic framework presented here could serve as a powerful new tool for wavelet-based statistical signal and image processing, with applications in signal estimation, detection, classification, compression, and even synthesis. Work remains in (1) characterizing accurate, practical graph structures for wavelet modeling (which coefficients to connect?), and (2) formulating robust, efficient algorithms for analyzing and estimating these structures. A key to future work is tapping into the knowledge base that has already accumulated in statistics, speech recognition, artificial intelligence, and related fields.

References

[1] I. Daubechies, *Ten Lectures on Wavelets*. New York: SIAM, 1992.

[2] D. Donoho and I. Johnstone, "Adapting to unknown smoothness via wavelet shrinkage," *J. Amer. Stat. Assoc.*, vol. 90, pp. 1200–1224, Dec. 1995.

[3] J.-C. Pesquet, H. Krim, and E. Hamman, "Bayesian approach to best basis selection," in *IEEE Int. Conf. on Acoust., Speech, Signal Proc. — ICASSP '96*, (Atlanta), pp. 2634–2637, 1996.

[4] H. Chapman, E. Kolaczyk, and E. McCulloch, "Signal de-noising using adaptive Bayesian wavelet shrinkage," in *Proc. IEEE-SP Int. Symp. Time-Frequency and Time-Scale Analysis*, (Paris), pp. 225–228, June 1996.

[5] J. Shapiro, "Embedded image coding using zerotrees of wavelet coefficients," *IEEE Trans. Signal Proc.*, vol. 41, pp. 3445–3462, Dec. 1993.

[6] R. Redner and H. Walker, "Mixture densities, maximum likelihood and the EM algorithm," *SIAM Review*, vol. 26, pp. 195–239, Apr. 1994.

[7] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Francisco: Morgan Kaufmann, 1988.

[8] P. Smyth, D. Heckerman, and M. Jordan, "Probabilistic independence networks for hidden markov probability models," *Neural Comp.*, vol. 9, no. 1, To appear.

[9] L. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, pp. 257–285, Feb. 1989.

[10] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Royal Stat. Soc.*, vol. 39, pp. 1–38, 1977.

[11] O. Ronen, J. Rohlicek, and M. Ostendorf, "Parameter estimation of dependence tree models using the em algorithm," *IEEE Signal Proc. Lett.*, vol. 2, pp. 157–159, Aug. 1995.

[12] M. C. Crouse, R. D. Nowak, and R. G. Baraniuk, "Hidden Markov models for wavelet-based signal processing," Tech. rep., Dept. ECE, Rice Univ., 1996.