

# Multiscale Recursive Estimation, Data Fusion, and Regularization

Kenneth C. Chou, *Member, IEEE*, Alan S. Willsky, *Fellow, IEEE*, and Albert Benveniste, *Fellow, IEEE*

**Abstract**—A current topic of great interest is the multiresolution analysis of signals and the development of multiscale signal processing algorithms. In this paper, we describe a framework for modeling stochastic phenomena at multiple scales and for their efficient estimation or reconstruction given partial and/or noisy measurements which may also be at several scales. In particular multiscale signal representations lead naturally to pyramidal or tree-like data structures in which each level in the tree corresponds to a particular scale of representation. Noting that scale plays the role of a time-like variable, we introduce a class of multiscale dynamic models evolving on dyadic trees. The main focus of this paper is on the description, analysis, and application of an extremely efficient optimal estimation algorithm for this class of models. This algorithm consists of a fine-to-coarse filtering sweep, followed by a coarse-to-fine smoothing step, corresponding to the dyadic tree generalization of Kalman filtering and Rauch-Tung-Striebel smoothing. The Kalman filtering sweep consists of the recursive application of three steps: a measurement update step, a fine-to-coarse prediction step, and a fusion step, the latter of which has no counterpart for time- (rather than scale-) recursive Kalman filtering. We illustrate the use of our methodology for the fusion of multiresolution data and for the efficient solution of “fractal regularizations” of ill-posed signal and image processing problems encountered, for example, in low-level computer vision.

## I. INTRODUCTION

MULTIRESOLUTION signal and image analysis methods have been investigated for some time under a variety of names including multirate filters [27], subband coding [22], Laplacian pyramids [7], and “scale-space” image processing [30]. It is the emerging theory of wavelet transforms, [11], [12], [15], [17]–[19], [23], that has sparked much of the recent flurry of activity in this area, in part because of its rich mathematical foundation and in part because of the evocative examples suggesting the possibility of developing efficient, optimal processing algorithms. The development of such algorithms (e.g., for the reconstruction of noise degraded signals) and the evaluation of their performance requires the

Manuscript received December 10, 1991; revised February 7, 1993. Recommended by Past Associate Editor W. S. Wong. This work was supported in part by the Air Force Office of Scientific Research under Grant AFOSR-92-J-0002, by the National Science Foundation under Grants MIP-9015281 and INT-9002393, and by the Office of Naval Research under Grant N00014-91-J-1004.

K. C. Chou is with SRI International, Menlo Park, CA 94025.

A. S. Willsky is with the Laboratory for Information and Decision Systems and Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139.

A. Benveniste is with the Institut de Recherche en Informatique et Systemes Aleatoires (IRISA), Campus de Beaulieu, 35042 Rennes, CEDEX, FRANCE. The research of this author was also supported in part by Grant CNRS G0134. IEEE Log Number 9213718.

development of a corresponding multiscale theory of stochastic processes that allows us both to model phenomena and to develop effective analysis tools. The research presented in this paper and in several others, [2]–[5], [8] has the development of such a theory as its objective.

In this paper we examine a class of multiscale state space models. Standard time domain state space models have proven to be of considerable value both because of the extremely efficient algorithms they admit (e.g., the Kalman filter) and because rich classes of stochastic phenomena can be well modeled using such descriptions. As we will see, both of these are also true for our multiscale state models, leading to the possibility of devising novel and extremely efficient algorithms for a variety of signal and image analysis problems. The key to our development is the observation that multiscale representations, whether for one-dimensional (1-D) time series or multidimensional images, have a natural, time-like variable, namely *scale*. Essentially all methods for representing and processing signals at multiple scales involve pyramidal data structures, where each level in the pyramid corresponds to a particular scale and each node at a given scale is connected both to a parent node at the next coarser scale and to several descendent nodes at the next finer scale.

In 1-D, the usual scale-to-scale decimation by a factor of two leads directly to a dyadic tree data structure. The simplest example of this is the Haar wavelet representation in which the representation of a signal  $f(x)$  at the  $m$ th scale is given by its average values  $f(m, n)$  over successive intervals of length  $2^{-m}$ , i.e.,

$$f(m, n) = k_m \int_{n2^{-m}}^{(n+1)2^{-m}} f(x) dx \quad (1.1)$$

where  $k_m$  is a normalizing constant. In this case each node  $(m, n)$  is connected to a single “parent” node  $(m-1, [n/2])$ , where  $[y]$  = the integer part of  $y$ , and to two “descendent” nodes  $(m+1, 2n)$ ,  $(m+1, 2n+1)$ , where the fine-to-coarse relationship among the  $f(m, n)$  values is given by an interpolation plus the adding of higher resolution detail not available at the coarser level.

An important aspect of this example is that the signal representations at different scales are related by a local scale-to-scale recursion on the dyadic tree structure of the  $(m, n)$  index set. Also, while the fine-to-coarse recursion corresponds to the multiresolution analysis of signals, the coarse-to-fine recursion, in which we add higher resolution detail at each scale, corresponds to the multiresolution *synthesis* of signals. It is the latter of these that is appropriate for the multiresolution

modeling of signals and phenomena. In doing this, however, we wish to consider a far broader class of models than the Haar transform. In particular we choose to view each scale of such a representation more abstractly, much as in the notion of state, as capturing the features of signals up to that scale that are relevant for the “prediction” of finer scale approximations. For example, the Haar transform can naturally be thought of as a first-order recursion in scale. As we know from time series analysis, however, a considerably broader class of models is obtained if we allow higher-order dynamics, i.e., by introducing additional memory in scale.

In this paper we develop such an extension by considering vector state models on the dyadic tree, providing a framework that allows us to model phenomena with multiscale features and to develop extremely efficient, parallelizable algorithms. By adopting this perspective, we provide a natural setting not only for dealing with multiscale phenomena and algorithms but also with multiscale data. In particular, a problem of considerable importance (e.g., in remote sensing) is the fusion of data from heterogeneous suites of sensors (multicolor IR, visual, microwave, etc.) providing information in different spectral bands and at different resolutions. The framework we describe allows the modeling of such multiresolution data simply as measurements at different levels in the dyadic tree, resulting in data fusion algorithms that are no more complex than algorithms for filtering single resolution data. As we discuss, the same cannot be said for standard estimation formulations. Since the key to our models and algorithms are recursions in scale, we obtain essentially the same algorithmic structures for two- or higher-dimensional image processing. For example in two-dimensions (2-D) our dyadic tree would be replaced by a *quadtree* in which each node has four descendants rather than two, resulting in the same order of complexity per data point as in 1-D. This stands in stark contrast to other optimal 2-D estimation formulations which have per-point complexities that grow with the size of the data array to be processed. The use of a simple quadtree model of the type described in this paper has been explored in the context of image coding and reconstruction in [9], [25].

In the next section we introduce our model and perform some elementary statistical analysis. In Section III we then investigate the problem of optimal estimation for this model and develop the generalization of the Rauch–Tung–Striebel (RTS) algorithm [20], consisting of a fine-to-coarse filtering sweep followed by a coarse-to fine smoothing sweep. The fine-to-coarse sweep, corresponding to a generalization of the Kalman filter to multiscale models on trees, consists of a three-step recursion of measurement updating, fine-to-coarse prediction, and the fusion of information as we move from fine-to-coarse scales. The last of these three steps has no counterpart in standard Kalman filtering, and this in turn leads to a new scale-recursive Riccati equation. In Section IV, we illustrate the application of our methodology for the estimation of both fractal,  $1/f$ -like processes and standard Gauss–Markov processes, both based on single-scale measurements and based on the fusion of multiresolution data. In addition we demonstrate the use of our methodology for the efficient solution, via “fractal regularization,” of a motion

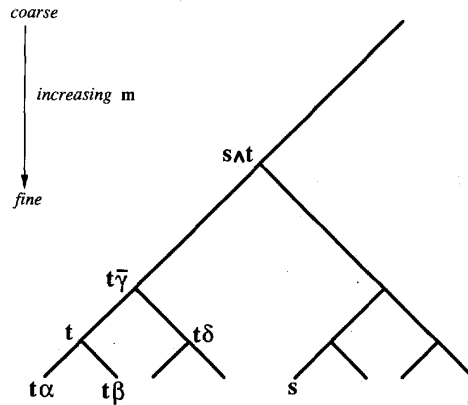


Fig. 1. Illustrating the multiscale structure of the dyadic tree and some notation used in the paper.

estimation problem typical of many ill-posed image processing problems encountered, for example in low-level computer vision. For simplicity our entire development is carried out in the context of the dyadic tree which corresponds to the representation and processing of 1-D signals. As we have indicated the extension to higher dimensions introduces only notational rather than analytical or computational complexity, offering the possibility of substantial computational savings for many image processing problems.

## II. STATE-SPACE MODELS ON DYADIC TREES

In this section, we introduce dynamic models on the dyadic tree of scale/translation pairs  $(m, n)$  where each value of  $m$  corresponds to a particular scale of resolution and there is a factor of two decimation from scale to scale. As shown in Fig. 1, for convenience we denote each node of the tree by a single abstract index  $t$ , i.e.,  $t = (m, n)$ , where  $T$  denotes the set of all nodes, and  $m(t)$  denotes the scale or  $m$ -component of  $t$ . We also introduce the basic shift operators on  $T$ , which play roles analogous to forward and backward shifts for temporal systems. In particular, with increasing  $m$  (i.e., coarse-to-fine) denoting the forward direction, there is a unique backward shift  $\bar{\gamma}$  and two forward shifts  $\alpha$  and  $\beta$  (see Fig. 1).<sup>1</sup> If  $t = (m, n)$ , then  $t\alpha = (m + 1, 2n)$ ,  $t\beta = (m + 1, 2n + 1)$ , and  $t\bar{\gamma} = (m - 1, \lceil n/2 \rceil)$ .

The structure of  $T$  admits a class of scale-recursive linear dynamic models defined locally on  $T$  and evolving from coarse to fine scales:

$$x(t) = A(t)x(t\bar{\gamma}) + B(t)w(t) \quad (2.1)$$

$$y(t) = C(t)x(t) + v(t) \quad (2.2)$$

where  $w(t)$  and  $v(t)$  are independent, zero-mean white noise processes with covariances  $I$  and  $R(t)$ , respectively, and  $x(t)$  is an  $n$ -dimensional, zero-mean stochastic process. The term  $A(t)x(t\bar{\gamma})$  represents a coarse-to-fine prediction or interpolation,  $B(t)w(t)$  represents the higher resolution detail added

<sup>1</sup>The other operator  $\delta$ , which maps  $t$  to its nearest neighbor,  $t\delta$ , at the same scale, is used only in Appendix B.

in going from one scale to the next finer scale, and  $y(t)$  is the measured variable (if any) at the particular scale  $m$  and location  $n$  represented by  $t$ . Thus (2.1) represents a generalization of the coarse-to-fine synthesis form of the wavelet transform in that we allow additional memory, captured by  $x(t)$ , at each scale, together with a general scale-to-scale linear recursion rather than the particular synthesis recursion of the Haar transform.

The general model (2.1)–(2.2) allows full  $t$ -dependence of all the system matrices, e.g.,  $A(t)$  can vary with both scale and translational position. An important special case is that in which the parameters are constant at each scale but may vary from scale to scale, in which case we abuse notation by writing  $A(t) = A(m(t))$ , etc. Such a model is useful for capturing a variety of scale-dependent effects such as  $1/f$ -like, fractal behavior as shown in [28], [29]. Also, as we illustrate in Sections IV-B and IV-C, the general case of  $t$ -varying parameters also has a number of potential uses.

We assume throughout this paper that  $A(t)$  is invertible for all  $t$ . The analysis and results we present extend to the case when this is not true, but our discussion is simplified if we make this assumption. In addition we assume that  $w(t)$  is independent of the “past” of  $x$ , i.e.,  $\{x(\tau) | m(\tau) < m(t)\}$ . In this case, (2.1) not only describes a scale-to-scale Markov process, but it in fact specifies a Markov random field on  $T$  in that conditioned on  $x(t\bar{\gamma})$ ,  $x(t\alpha)$ , and  $x(t\beta)$   $x(t)$  is independent of  $x$  at all other nodes.<sup>2</sup> Also, if we wish to consider representations of signals of unbounded extent, we must deal with the full infinite tree  $T$ , i.e.,  $\{(m, n) | -\infty < m, n < \infty\}$ . If we are concerned with a compact interval of data, the index set of interest represents a finite version of the tree of consisting of  $M + 1$  levels beginning with the coarsest scale represented by a unique root node, denoted by 0, and  $M$  subsequent levels, the finest of which has  $2^M$  nodes. In this case, we simply assume that  $w(t)$  is independent of the initial condition  $x(0)$  which is assumed to have zero mean and covariance  $P_x(0)$ .

Straightforward calculation shows that the covariance  $P_x(t) = E[x(t)x^T(t)]$  evolves according to a Lyapunov equation on the tree

$$P_x(t) = A(t)P_x(t\bar{\gamma})A^T(t) + B(t)B^T(t). \quad (2.3)$$

Let  $K_{xx}(t, s) = E[x(t)x^T(s)]$ . Let  $s \wedge t$  denote the least upper bound of  $s$  and  $t$ , i.e., the finest scale node that is a predecessor of both  $t$  and  $s$ . Then

$$K_{xx}(t, s) = \Phi(t, s \wedge t)P_x(s \wedge t)\Phi^T(s, s \wedge t) \quad (2.4)$$

where  $\Phi(t_1, t_2)$  is the state transition matrix on the tree

$$\Phi(t_1, t_2) = \begin{cases} I & t_1 = t_2 \\ A(t_1)\Phi(t_1\bar{\gamma}, t_2) & m(t_1) > m(t_2) \end{cases} \quad (2.5)$$

Equation (2.4) differs from the formula for standard state models in which the transition matrix appears on either the left or right side (but not both).

<sup>2</sup>Indeed this fact is used in [33] to describe a multigrid-like iterative algorithm for the solution of the multiscale estimation problem studied in this paper.

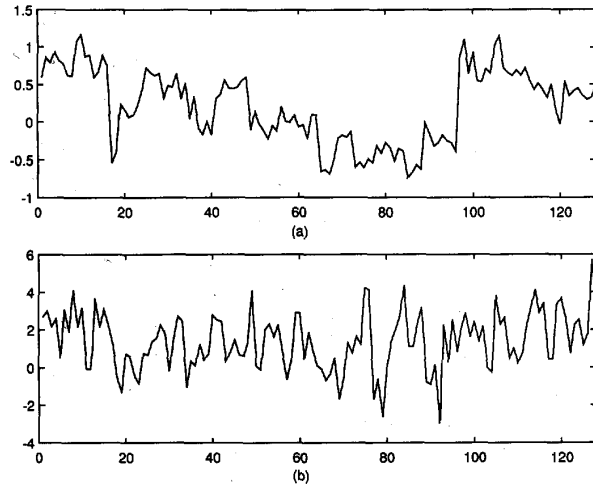


Fig. 2. Sample paths for two scalar multiscale models: (a) Process with  $A = .9$ ,  $B(m) = 2^{-m/2}$ ,  $P_x(0) = 1$ ; (b) Process with  $A = .9$ ,  $B = P_x(0) = 1$ .

In the case in which the model parameters vary in scale only, if at some scale  $P_x(t)$  is constant, then this holds at each scale, so that by an abuse of notation  $P_x(t) = P_x(m(t))$ , and we have a scale-to-scale Lyapunov equation

$$P_x(m+1) = A(m)P_x(m)A^T(m) + B(m)B^T(m) \quad (2.6)$$

(this is always true for a finite subtree with single root node 0), and

$$K_{xx}(t, s) = \Phi(m(t), m(s \wedge t))P_x(m(s \wedge t)) \cdot \Phi^T(m(s), m(s \wedge t)) \quad (2.7)$$

where  $\Phi(m_1, m_2)$  is the state transition matrix for  $A(m)$ . Fig. 2(a) depicts the sample path of a scalar model with  $A = .9$  and  $B(m) = 2^{-m/2}$ ,  $m = 1, \dots, 7$ , and  $P_x(0) = 1$ . The use of a geometrically-varying noise gain allows us to capture fractal characteristics with commensurately-scaled fluctuations at all scales.

If we further specialize our model to the case in which  $A$  and  $B$  are constant, and  $A$  is stable, then (2.6) admits a steady-state solution,  $P_x$ , which satisfies the usual algebraic Lyapunov equation. In this case if  $P_x(t) = P_x$  for all  $t$ , we have what is referred to as a *stationary model* in [35]

$$K_{xx}(t, s) = A^{d(t, s \wedge t)}P_x(A^T)^{d(s, s \wedge t)} = K_{xx}(d(t, s \wedge t), d(s, s \wedge t)) \quad (2.8)$$

where  $d(t_1, t_2)$  denotes the distance between  $t_1$  and  $t_2$ , i.e., the number of branches on the path from  $t_1$  to  $t_2$ . Fig. 2(b) depicts the sample path of a scalar model with  $A = .9$ ,  $B = P_x(0) = 1$ . The covariance in (2.8) only depends on the distances of  $s$  and  $t$  to their common “parent” node  $s \wedge t$ . This yields a notion of self similarity since, roughly speaking (2.8) states that the correlation between  $x(t)$  and  $x(s)$  depends on the differences in scale and in temporal offset of the nodes  $t$  and  $s$ . A stronger notion of shift invariance for stochastic

processes on trees is *isotropy* in which  $K_{xx}(t, s)$  depends only on  $d(s, t)$ . Note also that (2.8) represents an isotropic covariance if  $AP_x = P_x A^T$ , which points to the connection to the class of reversible stochastic processes [1]. Note also that since  $d(s, t) = d(s, s \wedge t) + d(t, s \wedge t)$ , any isotropic process is stationary, but the reverse implication is not always true. We refer the reader to [5], [6], [35] for a detailed analysis of isotropic processes.

Finally, in the next section we encounter the need for fine-to-coarse prediction and recursion, i.e., a model representing  $x(t\bar{\gamma})$  in terms of  $x(t)$  and a noise that is uncorrelated with  $x(t)$ . To do this, we can directly apply the results of [26]

$$x(t\bar{\gamma}) = F(t)x(t) - A^{-1}(t)B(t)\tilde{w}(t) \quad (2.9)$$

$$\begin{aligned} F(t) &= A^{-1}(t)[I - B(t)B^T(t)P_x^{-1}(t)] \\ &= P_x(t\bar{\gamma})A^T(t)P_x^{-1}(t) \end{aligned} \quad (2.10)$$

$$\tilde{w}(t) = w(t) - E[w(t)|x(t)] \quad (2.11)$$

$$\begin{aligned} E[\tilde{w}(t)\tilde{w}^T(t)] &= I - B^T(t)P_x^{-1}(t)B(t) \\ &\triangleq \tilde{Q}(t). \end{aligned} \quad (2.12)$$

For standard temporal models the noise process  $\tilde{w}(t)$  is white in time. In our case the situation is a bit more complex. In particular  $\tilde{w}(t)$  is white along all upward paths on the tree, i.e.,  $\tilde{w}(s)$  and  $\tilde{w}(t)$  are uncorrelated if  $s \wedge t = s$  or  $t$ . Otherwise it is not difficult to check that  $\tilde{w}(s)$  and  $\tilde{w}(t)$  are not uncorrelated.<sup>3</sup>

### III. A TWO-SWEEP ESTIMATION ALGORITHM FOR MULTISCALE PROCESSES

In this section we derive an extremely efficient algorithm for the smoothing of (possibly) multiscale measurement data for our dynamic system (2.1), (2.2). Recall that the standard RTS algorithm involves a forward Kalman filtering sweep followed by a backward sweep to compute the smoothed estimates. The generalization to our models on trees has the same structure, with several important differences. First for the standard RTS algorithm the procedure is completely symmetric with respect to time, i.e., we can start with a reverse-time Kalman filtering sweep followed by a forward smoothing sweep. For processes on trees, the Kalman filtering sweep must proceed from fine-to-coarse (i.e., in the reverse direction from that in which the model (2.1) is defined) followed by a coarse-to-fine smoothing sweep.<sup>4</sup> Furthermore, one full step of the Kalman filter recursion involves a measurement update, two parallel backward predictions (corresponding to backward prediction along both of the paths descending from a node), and the fusion of these predicted estimates. This last step

<sup>3</sup>In fact  $\tilde{w}(t)$  is a martingale difference for a martingale defined on the partially-ordered tree [31].

<sup>4</sup>The reason for this is not very complex. To allow the measurement on the tree at one point to contribute to the estimate at another point on the same level of the tree, one must use a recursion that first moves up and then down the tree.

has no counterpart for state models evolving in time and is one of the major reasons for the differences between the analysis of temporal Riccati equations and that presented in the sequel [34] to this paper. Note also that our algorithm has a pyramidal structure consistent with that of the tree and thus has considerable parallelism. To begin, let us define some notation

$$Y_t = \{y(s)|s = t \text{ or } s \text{ is a descendant of } t\} \quad (3.1)$$

$$\begin{aligned} Y_t^+ &= \{y(s)|s \text{ is a descendant of } t\} \\ &= Y_t - \{t\} = Y_{t\alpha} \cup Y_{t\beta} \end{aligned} \quad (3.2)$$

$$\hat{x}(\cdot|t) = E[x(\cdot)|Y_t] \quad (3.3)$$

$$\hat{x}(\cdot|t+) = E[x(\cdot)|Y_t^+]. \quad (3.4)$$

Suppose that we have computed  $\hat{x}(t|t+)$  and the corresponding error covariance,  $P(t|t+)$ . Then, standard estimation results yield

$$\hat{x}(t|t) = \hat{x}(t|t+) + K(t)[y(t) - C(t)\hat{x}(t|t+)] \quad (3.5)$$

$$K(t) = P(t|t+)C^T(t)V^{-1}(t) \quad (3.6)$$

$$V(t) = C(t)P(t|t+)C^T(t) + R(t) \quad (3.7)$$

$$P(t|t) = [I - K(t)C(t)]P(t|t+). \quad (3.8)$$

Suppose now that we have computed  $\hat{x}(t\alpha|t\alpha)$  and  $\hat{x}(t\beta|t\beta)$ . Note that  $Y_{t\alpha}$  and  $Y_{t\beta}$  are disjoint, and these estimates can be calculated in parallel. We then compute

$$\hat{x}(t|t\alpha) = F(t\alpha)\hat{x}(t\alpha|t\alpha) \quad (3.9)$$

$$\hat{x}(t|t\beta) = F(t\beta)\hat{x}(t\beta|t\beta) \quad (3.10)$$

with corresponding error covariances given by

$$P(t|t\alpha) = F(t\alpha)P(t\alpha|t\alpha)F^T(t\alpha) + Q(t\alpha) \quad (3.11)$$

$$Q(t\alpha) = A^{-1}(t\alpha)B(t\alpha)\tilde{Q}(t\alpha)B^T(t\alpha)A^{-T}(t\alpha) \quad (3.12)$$

$$P(t|t\beta) = F(t\beta)P(t\beta|t\beta)F^T(t\beta) + Q(t\beta) \quad (3.13)$$

$$Q(t\beta) = A^{-1}(t\beta)B(t\beta)\tilde{Q}(t\beta)B^T(t\beta)A^{-T}(t\beta). \quad (3.14)$$

These equations follow directly from the backward model (2.9)–(2.12).

The third and final step of the Kalman filtering recursion is to merge the estimates (3.9) and (3.10), to form  $\hat{x}(t|t+)$

$$\begin{aligned} \hat{x}(t|t+) &= P(t|t+)[P^{-1}(t|t\alpha)\hat{x}(t|t\alpha) \\ &\quad + P^{-1}(t|t\beta)\hat{x}(t|t\beta)] \end{aligned} \quad (3.15)$$

$$P(t|t+) = [P^{-1}(t|t\alpha) + P^{-1}(t|t\beta) - P_x^{-1}(t)]^{-1}. \quad (3.16)$$

The interpretation of (3.15), (3.16) is rather simple:  $\hat{x}(t|t\alpha)$  ( $\hat{x}(t|t\beta)$ ) is the best estimate of  $x(t)$  based on the prior statistics of  $x(t)$  (i.e., it is zero-mean, with covariance  $P_x(t)$ ) and the measurements in  $Y_{t\alpha}$  ( $Y_{t\beta}$ ); consequently the fusion of these estimates, as captured in (3.15) and (3.16), must avoid a double-counting of prior information. A brief proof of (3.15) and (3.16) is given in Appendix A.

Equations (3.5)–(3.16) define the coarse-to-fine Kalman filter for our multiscale stochastic model. The on-line calculations consist of an update step (3.5), a pair of parallel prediction steps (3.9), (3.10), and the fusion step (3.15). The associated Riccati equation consists of three corresponding steps (3.6)–(3.8), (3.11)–(3.14), (3.16) the first two of which correspond to the usual Riccati equation (in reverse time). The third step (3.16) has no counterpart in the standard case.

Let us now consider the computation of  $\hat{x}_s(t)$ , the optimal smoothed estimate of  $x(t)$  based on all available data on a finite subtree with root node 0 and  $M$  scales below it. The initialization of the Kalman filter in this case at scale  $m(t) = M$  is given by  $\hat{x}(t|t+) = 0$ ,  $P(t|t+) = P_x(t)$ . Once the Kalman filter has reached the root node at the top of the tree, we have computed  $\hat{x}_s(0) = \hat{x}(0|0)$ , which serves as the initial condition for the coarse-to-fine smoothing sweep which also has a parallel, pyramidal structure. Specifically, suppose that we have computed  $\hat{x}_s(t\bar{\gamma})$ . This is then combined with the fine-to-coarse filtered estimate  $\hat{x}(t|t)$  to produce  $\hat{x}_s(t)$

$$\hat{x}_s(t) = \hat{x}(t|t) + J(t)[\hat{x}_s(t\bar{\gamma}) - \hat{x}(t\bar{\gamma}|t)] \quad (3.17)$$

where

$$J(t) \triangleq P(t|t)F^T(t)P^{-1}(t\bar{\gamma}|t). \quad (3.18)$$

We also have a coarse-to-fine recursion for the smoothing error covariance, initialized with  $P_s(0) = P(0|0)$

$$P_s(t) = P(t|t) + J(t)[P_x(t\bar{\gamma}) - P(t\bar{\gamma}|t)]J^T(t). \quad (3.19)$$

Equations (3.17)–(3.19) are of the same form as the usual RTS smoothing sweep, although the derivation, given in Appendix B, is somewhat more involved.

Note that all of the coarse-to-fine and fine-to-coarse processing steps can be performed in parallel with only nearest neighbor communication required (e.g., the fusion step). Such tree-like structures map directly onto hypercube architectures which can achieve logarithmic speed-up: for example, the estimates at the  $2^M$  nodes at the finest level can be computed in time proportional to the number of levels in the tree, i.e., in  $O(M)$  steps. This can be compared to standard temporal RTS smoothers, which, for a signal of length  $2^M$ , require  $O(2^M)$  sequential steps.

#### IV. EXAMPLES AND APPLICATIONS

In this section we provide several illustrations of the application of the estimation framework described in this paper. The purpose of the first subsection is to demonstrate that the highly parallelizable estimation structure we have developed can be successfully applied to a rather rich class of processes

extending well beyond those that are precisely of the form generated by our tree models. In the second subsection we present examples of the fusion of multiresolution data which the algorithm of Section III accomplishes with essentially no increase in computational complexity as compared to the processing of single scale data. Finally in Section IV-C we examine a 1-D version of the computer vision problem of motion estimation, illustrating that the use of a slight variation on standard formulations, involving a “fractal prior” obtained from a tree model, yields excellent results while offering substantial computational advantages. In particular, while all of the examples in this section are 1-D, they all can be extended to 2-D, in which the potential computational savings can be quite dramatic.

##### A. Smoothing Gauss–Markov and 1/f Processes

In this section we consider smoothing problems for two stochastic processes, neither of which exactly fits the tree model used in this paper.

1) *Smoothing for a First-Order Gauss–Markov Process:* The first of these is a discrete-time first-order Gauss–Markov process, i.e., a stationary time series given by the standard first-order difference equation

$$x_{n+1} = \alpha x_n + w_n \quad (4.1)$$

where, for simplicity, we normalize the variance of  $x_n$  to a value of one, so that the variance of the white noise sequence  $w_n$  is  $(1 - \alpha^2)$ . For our example we take  $\alpha = .9006$ , a value arrived at by sampling and aliasing considerations [33]. We consider the following measurements of  $x_n$ .

$$y_n = x_n + v_n, \quad n = 0, \dots, N - 1 \quad (4.2)$$

where  $v_n$  has variance  $R$  so that the SNR in (4.2) is  $R^{-1/2}$ . In the examples that follow we take  $N = 128$ .

As developed in [6], [8], [14] while the wavelet transform of  $\{x_n | 0 \leq n \leq N - 1\}$  does not yield a completely whitened sequence of wavelet coefficients, it does accomplish a substantial level of decorrelation, i.e., the detail added to the approximation at each scale is only weakly correlated with the coarser-scale approximation. Furthermore, decorrelation is improved by using wavelets of larger support, and in particular those with increasing numbers of vanishing moments [6], [11], suggesting a wavelet-transform-based estimation algorithm developed in [8], [33]. It also provides the motivation for the example presented here in which we smooth the data in (4.2) using a model for  $x_n$  as the finest level process generated by a scalar model of the form (2.1), (2.2), where in this case the data (4.2) corresponds to measurements only at the finest level of the tree. In particular we have considered two scalar tree models, the first being a constant parameter model in steady-state

$$x(t) = ax(t\bar{\gamma}) + w(t) \quad (4.3)$$

with  $w(t)$  having variance equal to  $p(1 - a^2)$ , where  $p$  is the variance of  $x(t)$ . In our example in which the length of the signal  $x_n$  is 128, the model (4.3) evolves from the root node

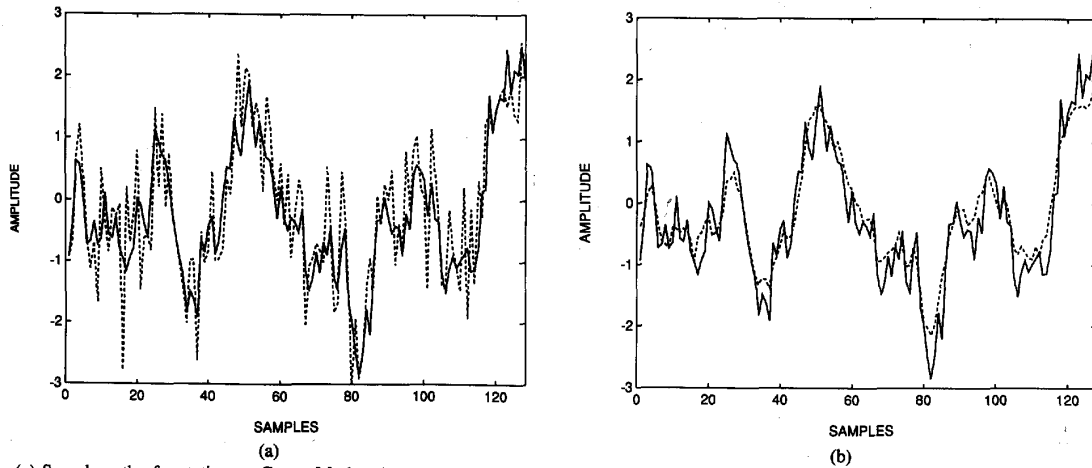


Fig. 3. (a) Sample path of a stationary Gauss-Markov Process (solid) and its noisy measurement with SNR = 1.4142 (dashed); (b) the same Gauss-Markov sample path (solid) and its smoothed version using the optimal smoother for the Gauss-Markov model.

to level  $m = 7$ , at which we obtain our approximate model for  $x_n$  and at which we incorporate the measurements (4.2).

A second model that we considered has the property that its fine-scale variations have smaller variance than its coarser-scale variations. In particular, as developed in [13], [24], [28], fractal-like processes which have spectra that vary as  $1/f^\beta$  for some value of  $\beta$ , have wavelet variances that decrease geometrically with  $m$ . Thus we consider a variation on the model (4.3)

$$x(t) = ax(t\bar{\gamma}) + 2^{-\delta m(t)/2}w(t) \quad (4.4)$$

where  $\delta$  controls the scale-to-scale geometric decay in the variance of the noise term,  $w(t)$  has unit variance, and the variance of the initial condition,  $x(0)$ , at the top of the tree is  $p_0$ .

Models of this type were used to design suboptimal estimators for the Gauss-Markov process (4.1) based on the data (4.2) at several different SNR's. Fig. 3(a) illustrates a sample path of (4.1) and data (4.2) at SNR =  $\sqrt{2}$ , while Fig. 3(b) compares the Gauss-Markov sample path to the optimal estimate using a standard optimal smoothing algorithm for the model (4.1). To compare this optimal performance to that of algorithms based on the tree models (4.3), (4.4) we define a metric for comparison based on error variance reduction. Specifically if  $p_i$  is the prior variance of a process, and  $p_{opt}$  and  $p_{sub}$  denote the error variances of the optimal and suboptimal estimates, respectively, then

$$\Delta = \frac{p_{sub} - p_{opt}}{p_i - p_{opt}} \quad (4.5)$$

is the loss in error variance reduction (from prior to estimate) due to using the suboptimal rather than the optimal estimate, where this loss is expressed as a fraction of the error variance reduction achieved by the optimal estimate.<sup>5</sup>

For each SNR considered and for each of the models (4.3), (4.4), we chose the model parameters ( $a$  and  $p$  for (4.3)

<sup>5</sup>For example, if  $p_i - p_{opt} = .5$  and  $p_i - p_{sub} = .4$ , then there is a 20% loss in variance reduction if we use the suboptimal estimate.

TABLE 1  
PERFORMANCE DEGRADATION COMPARISON FOR TREE  
SMOOTHERS BASED ON MODELS (4.3) AND (4.4)

	Model (4.3)	Model (4.4)
SNR = 2.8284	1.11%	1.08%
SNR = 1.412	3.55%	3.31%
SNR = .7071	7.59%	6.88%
SNR = .5	10.52%	9.15%

and  $a$ ,  $p_0$ , and  $\delta$  for (4.4) to minimize the corresponding value of  $\Delta$ . Table 1 presents the resulting values of  $\Delta$ , expressed in percentage form, indicating that estimators based on these models achieve variance reduction levels within a few percent of the optimal smoother.<sup>6</sup> This is further illustrated in Fig. 4 in which we compare in (a) the optimal estimate at SNR =  $\sqrt{2}$  for a tree smoother based on the model (4.4) with the original sample path. In Fig. 4(b) we compare the estimates produced by the optimal filter (Fig. 4(b)) and that of Fig. 4(a), providing further evidence of the excellent performance of our estimators. One should ask why this is significant, since we already have optimal smoothing algorithms for Gauss-Markov processes. There are at least three reasons: 1) the tree algorithms, with their pyramidal structure, are highly parallelizable; 2) as we will see in the next section, these algorithms directly allow us to fuse multiresolution data; and 3) perhaps most importantly, these same ideas extend to 2-D data where the computational savings over previously known algorithms are substantial.

Note that as part of this experiment we were faced with the issue of choosing a multiresolution state model that is "close" to the Gauss-Markov process in (4.1), (4.2) where the specific measure of "closeness" that we have used here is  $\Delta$ , namely the closeness in error variance performance between

<sup>6</sup>Note that the increasing size of  $\Delta$  with decreasing SNR is due for the most part to the decrease in the denominator of (4.5), i.e., at low SNR's only minimal variance is achieved by any estimator.

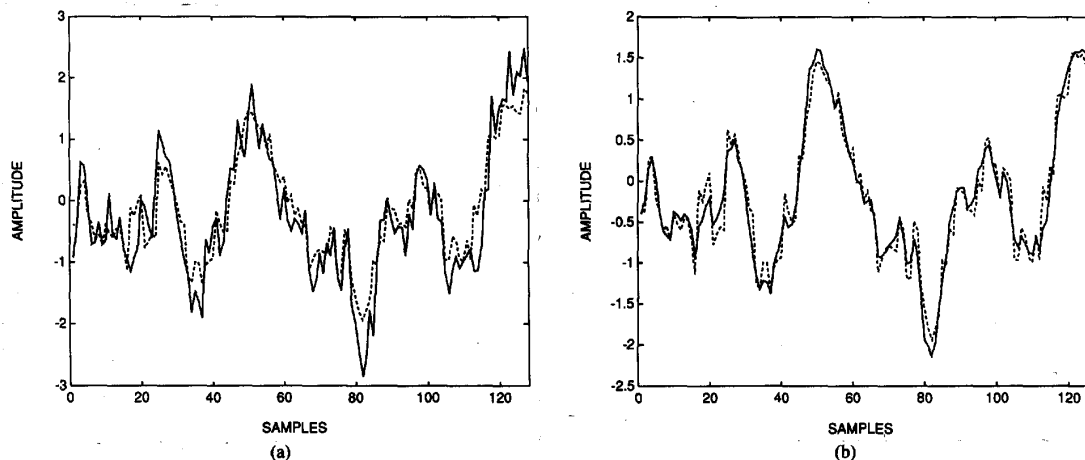


Fig. 4. (a) Stationary Gauss-Markov process (solid) versus the result of smoothing of the noisy data of Fig. 3(a) (dashed) based on the model (4.4) with parameters  $a = .9464$ ,  $b = 1$ ,  $p_0 = 7.7462$ ,  $\delta = .5059$ ; and (b) comparison of the optimal smoothing estimate of Fig. 3(b) (solid) versus the estimate in Fig. 4(a) (dashed).

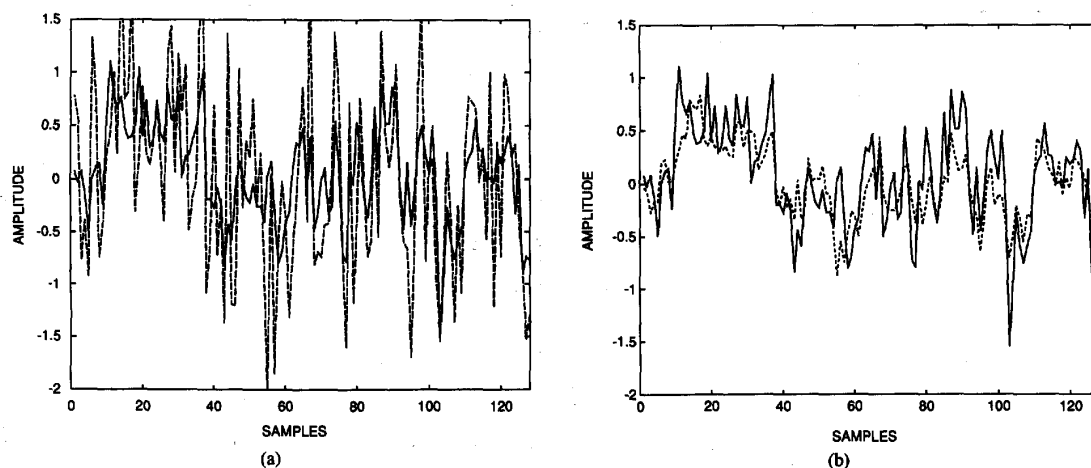


Fig. 5. (a) A  $1/f$  process (solid) generated using the method of [36] and its noisy measurement (dotted) with  $\text{SNR} = \sqrt{2}$ ; and (b) comparison of the  $1/f$  sample path (solid) versus the optimal estimate (dashed) based on the noisy data of part (a).

that of a suboptimal estimator based on a multidimensional model and the performance of the optimal estimator based on (4.1), (4.2). Obviously one could also consider model selection using any of the many other metrics that have been developed for measuring distance between models. For example, one such measure is the Bhattacharya distance used to bound the probability of error in deciding, based on noisy observations as in (4.2), if a given stochastic process corresponds to one of two models, and we refer the reader to [33] for the use of this metric for multiresolution models. As for any model class, the development of efficient methods for constructing models and assessing the goodness of fit is an important topic whose full development for our multiresolution models remains for the future. We also point to [5] in which some of the first steps in developing a realization theory for such models are described. The development of a full theory of stochastic realization and system identification for multiresolution models is still far

from completion, but the results presented here, we believe, provide motivation for its continued investigation.

2) *Smoothing for  $1/f$  Processes:* As a second example, we consider smoothing for a  $1/f$ -like fractal process of the type developed in [28]. Specifically as shown in [28], we can use the synthesis form of the wavelet transform to construct processes possessing self-similar statistical properties and with spectra which are nearly  $1/f$  by taking the wavelet coefficients to be white and with variances that decay geometrically at finer scales. In Fig. 5(a) we illustrate the sample path of such a process using the four-tap wavelet filter of Daubechies [11], as well as noisy data with an SNR of  $\sqrt{2}$ . In this case the scale-dependent geometric decay in wavelet coefficient variances is  $2^{-m}$ , yielding a  $1/f^\beta$  spectrum with  $\beta = 1$ . In Fig. 5(b) we illustrate the optimal estimate in this case. As developed in [8], [29] this optimal estimator can be implemented by taking the (four-tap) wavelet transform of the data and then filtering

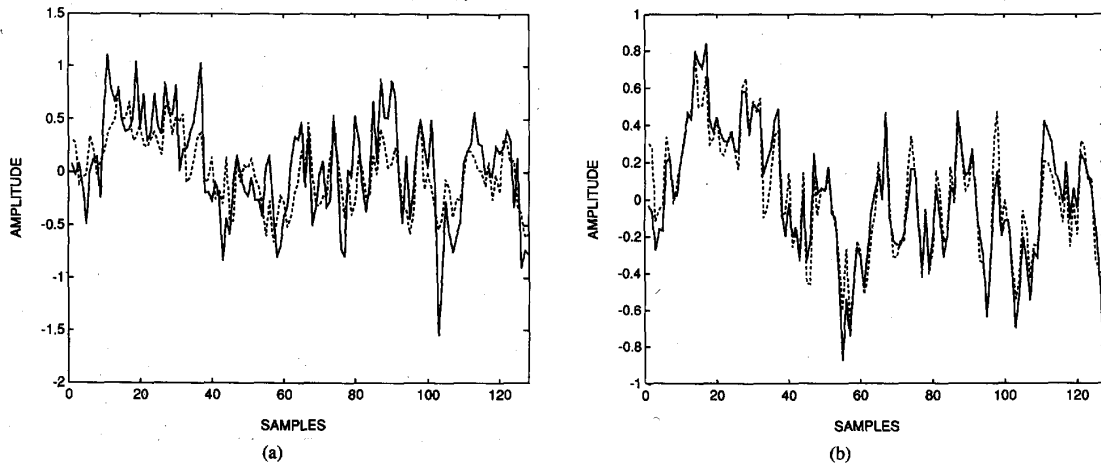


Fig. 6. (a) The  $1/f$  sample path of Fig. 5 versus the optimal estimate (dashed) based on the model (4.4) with  $a = .6694$ ,  $b = 1$ ,  $p_0 = 1.0131$ ,  $\delta = .4843$ ; and (b) comparison of the optimal estimates of Fig. 5(b) and 6(a) (dashed).

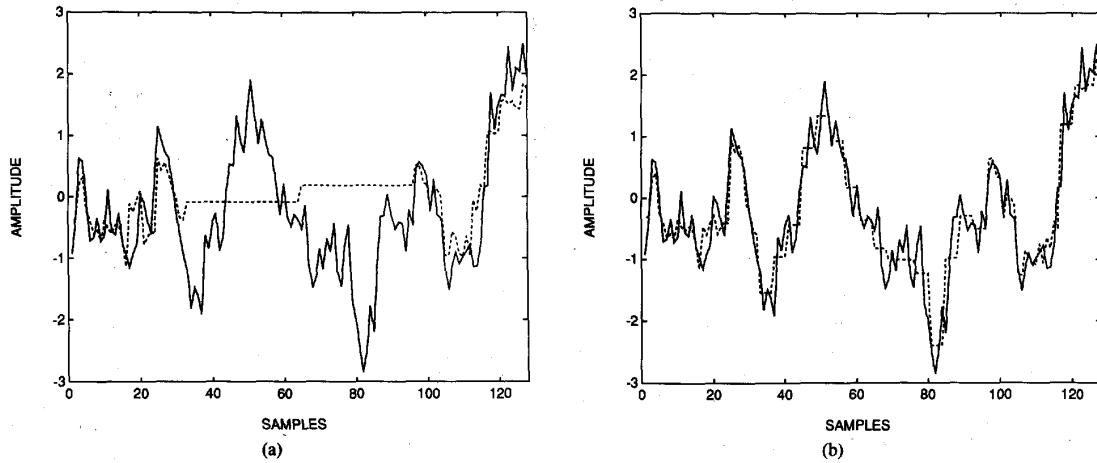


Fig. 7. (a) Sample path of the stationary Gauss–Markov Process (solid) of Fig. 3(a) and the result of using the tree smoother based on (4.4) and using sparse data of  $\text{SNR} = 1.4142$  (dashed); (b) comparing the Gauss–Markov sample path (solid) with the result (dashed) of using the tree smoother to fuse high quality coarse data (at two scales above the finest scale) of  $\text{SNR} = 100$  with the sparse fine data of  $\text{SNR} = 1.4142$ .

each wavelet coefficient separately (taking advantage of the fact that the wavelet transform yields the Karhunen–Loeve expansion for the data). In Fig. 6(a) we illustrate the result of smoothing the noisy data of Fig. 5(a) based on a tree model of the form of (4.4), and in Fig. 6(b) we compare this estimate to the optimal estimate of Fig. 5(b). The average performance degradation of the tree smoother is only 2.76%, although, as indicated in Fig. 6, the tree smoother appears to do *better* over certain portions of the signal, indicating that our models are well-suited to capturing and tracking  $1/f$ -like behavior. Again given the existence of effective optimal algorithms [8], [29], there is the natural question of why we should bother with the tree-based algorithm in this case, especially since these methods are very efficient, the methods of [8] can be applied to multiresolution data, and they can be extended to 2-D. The reason is that, for these wavelet-transform-based methods to apply, it is necessary that the identical type and quality of measurement be available along each scale (i.e.,  $C(t)$  in (2.2) must only depend on  $m(t)$ ). In the next two sections we encounter important examples in which this is not the case,

and thus the methods of [8], [29] are not applicable, but our tree-based methods are.

### B. Multiresolution Data Fusion

In this section, we illustrate the application of our tree-based smoothing algorithms to the fusion of multiresolution data. While we could provide examples in which data are available at all nodes at several levels of the tree—and such examples are presented in [8], [33] using wavelet-transform-based algorithms—we focus here on examples to which wavelet-based methods do not apply. In particular we consider an example in which we wish to fuse fine-scale data of limited coverage with full coverage coarse-scale data. Problems of this type arise, for example, in atmospheric or oceanographic sensing in which both broad coverage satellite data and sparse point measurement data (e.g., from surface measurement stations) are available.

Consider first the Gauss–Markov process of Fig. 3(a) where we assume that only 50% of the data in this figure are available, corresponding to the 32 data points at each end of the data interval. In Fig. 7(a) we illustrate the optimal



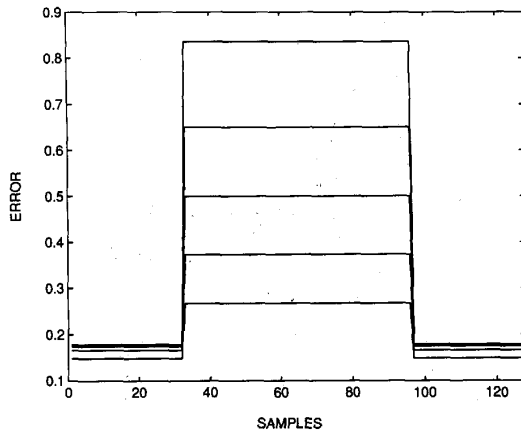


Fig. 8. Plots of performance for the case of full coverage coarse data fused with sparse fine data both of SNR = 1.4142. Five plots correspond to coarse data 1) one level coarser than fine data 2) two levels coarser. . . 5) five levels coarser.

tree-smoothed estimate based on the model of (4.4) and using only this limited data set. In contrast, in Fig. 7(b) we display the optimal tree-smoothed estimate based on the fusion of the limited fine-scale data with a full set of high quality measurements at a level two scales coarser (i.e., where there are 32 points rather than 128). Note that in addition to providing information that guides the interpolation of the fine-scale data, the coarse data also provides information that allows us to remove offset errors in regions in which noisy fine scale data are available. By using the multiscale covariance equations of Section III we can quantify the value of such coarse data, as illustrated in Fig. 8 in which we display the variation of performance with the level at which the coarse data is available. We refer the reader to [33] for other examples, including, for example, the  $1/f$  process of Fig. 5.

Note further that the algorithm used to compute the estimates based on this multiresolution data is of exactly the same form as that used in the preceding section, i.e., there is no increase in computational complexity as a result of including multiresolution data. This stands in sharp contrast to what happens in the standard Kalman filtering framework. In particular consider the model (4.1) based on fine-scale measurements as in (4.2) (for some sets of values of  $n$ ) together with coarse scale measurements, each of which has the form

$$z_n = \frac{1}{N}(x_n + x_{n-1} + \dots + x_{n-N+1}) + \mu_m \quad (4.6)$$

where, for the example shown here,  $N = 4$ . In this case the application of Kalman filtering techniques require the use of an augmented state, corresponding to a window of values of  $x_n$  of size corresponding to the coarsest scale measurement to be incorporated, i.e., equal to the largest value of  $N$ . Thus in the example here,  $N = 4$ , so the resulting Kalman filter would be four-dimensional, involving the solution of a  $4 \times 4$  Riccati equation. Furthermore the situation is worse if even coarser

measurements are to be fused. For example, if we have a measurement of the average value of  $x_n$  over the entire 128-point interval, then a 128-dimensional Kalman filter would, in principle, be required. In contrast, the dimension of our scale-recursive filter is always one, independent of the scales of the measurements to be processed.

Finally, let us comment on the measurement model for the coarse-level data in the example depicted in Fig. 7(b). In particular, the coarse data used in the example of Fig. 7 were noise-corrupted measurements of four-point averages of the fine-level process  $x_n$ . Note that using the model (4.4) we can view these as measurements two levels higher in the tree. For example, from (4.4) it is straightforward to see that the average of  $x(t\alpha^2)$ ,  $x(t\alpha\beta)$ ,  $x(t\beta\alpha)$  and  $x(t\beta^2)$  is a scaled version of  $x(t)$  corrupted by the noises  $w(t\alpha)$ ,  $w(t\beta)$ ,  $w(t\alpha^2)$ ,  $w(t\alpha\beta)$ ,  $w(t\beta\alpha)$ , and  $w(t\beta^2)$ . From this it is straightforward to obtain a model for the coarse measurements in which the measurement noise also reflects finer-scale process fluctuations captured by the  $w$ 's. Note that to be precise, this noise is correlated with the finer level data, and thus a truly optimal tree-based smoother would need to account for this. The result presented in Fig. 7(b), however, was generated using a tree-based smoother that neglected this correlation. As the quality of the result in the figure indicates, excellent performance is achieved with this simplification.

### C. Motion Estimation

In this section we illustrate the use of our tree model in order to provide statistical regularization for a 1-D version of a representative problem arising in computer vision, namely that of estimating *optical flow*, i.e., motion in the image frame that is apparent from observation of the temporal fluctuations in the intensity patterns in an image sequence. Regularization methods are common in many computer vision problems such as optical flow estimation and in other contexts including many inverse problems of mathematical physics, and the approach described in this section should prove to be of considerable value in many of these as well as in full 2-D and three-dimensional image analysis problems. In particular the usual "smoothness" regularization methods employed in many problems lead to computationally intensive variation problems solvable only by iterative methods. By adopting a statistical perspective and by modifying the regularization slightly, we obtain a formulation exactly of the form of our tree smoothing problem, leading to non-iterative, highly parallelizable and efficient scale-recursive solutions.

The results in this section focus on a 1-D version of the problem as formulated in Horn and Schunck [16]. Let  $f(x, t)$  denote a 1-D "image sequence", i.e., for each  $t$ ,  $f(x, t)$  is a function of the 1-D spatial variable  $x$ , representing image intensity variations with space. The constraint typically used in gradient-based approaches to motion estimation is referred to in the literature as the "brightness constraint" [16], and it amounts to assuming that the brightness of each point in the image, as we follow its movement, is constant. In other words, at each time  $t$  the total derivative of the image intensity is

zero, i.e.,

$$\frac{Df}{dt} = 0. \quad (4.7)$$

This equation can be rewritten as follows:

$$\frac{\partial f}{\partial t} + \frac{\partial f}{\partial x} \frac{\partial x}{\partial t} = 0. \quad (4.8)$$

The quantity  $(\partial x/\partial t)$  is referred to as the optical flow, which we denote as the spatial function,  $v(x)$ , and it is this quantity which we would like to determine from observing  $f(x, t)$  sequentially in time. That is, at each  $t$  we assume that we can extract measurements of the temporal and spatial derivatives of  $f(x, t)$  and we then want to use these, together with (4.8), to estimate the velocity field  $v(x)$  at this instant in time.

We rewrite (4.8) as

$$y(x) = c(x)v(x) \quad (4.9)$$

$$y(x) = -\frac{\partial f}{\partial t} \quad (4.10)$$

$$c(x) = \frac{\partial f}{\partial x}. \quad (4.11)$$

The following is the optimization problem formulation of Horn and Schunk [16] for determining  $v(x)$

$$\hat{v}(x) = \operatorname{argmin}_v \int \left\{ \mu \|y(x) - c(x)v(x)\|^2 + \left\| \frac{dv}{dx} \right\|^2 \right\} dx. \quad (4.12)$$

The second term in the cost function,  $\|dv/dx\|^2$ , is referred to as the ‘‘smoothness’’ constraint as it is meant to penalize large derivatives of the optical flow, constraining the solution to have a certain degree of smoothness. In the standard 2-D image processing problem  $v(x)$  is a 2-D velocity vector field,  $(\partial f/\partial x)$  is the 2-D gradient of image intensity, and, the brightness constraint (4.8) provides a scalar constraint at each spatial location for the 2-D vector  $v(x)$ . In this case the optical flow problem is ill-posed, and the smoothness penalty makes it well posed by, in essence, providing another (noisy) constraint in 2-D. The other purposes of the smoothness constraint is to reduce the effects of noise in the measurement of the temporal and spatial gradients and to allow interpolation of motion over regions of little or no contrast, i.e., regions where  $c(x)$  is zero or nearly zero. In our 1-D problem, we do not have the issue of ill-posedness, so that the role of the smoothness constraint is solely for the purposes of noise rejection and interpolation in areas of zero or near-zero intensity variations. Note that computing  $\hat{v}(x)$  in (4.12) is potentially daunting as the dimension of  $\hat{v}(x)$  is equal to the number of pixels in the image.

As in Rougee, *et al.* [21], the optimization problem in (4.12) can be interpreted as a stochastic estimation problem. In particular, the smoothness constraint can be interpreted as the following prior model on  $v(x)$ .

$$\frac{dv}{dx} = w(x) \quad (4.13)$$

where  $w(x)$  is a 1-D white noise process with unit intensity. The estimation problem equivalent to (4.12) consists of estimating  $v(x)$  based on the following observation equation

$$y(x) = c(x)v(x) + r(x) \quad (4.14)$$

where  $r(x)$  is white noise with intensity  $\mu^{-1}$ . Henceforth, we refer to (4.13), (4.14) as the *standard model*.

Let us now examine the particular prior model (4.13) associated with the smoothness constraint. In particular we see that  $v(x)$  is in fact a *Brownian motion process*, i.e., a self-similar, fractal process with a  $1/f^2$  spectrum. For this reason (4.12), (4.13) is sometimes referred to as a ‘‘fractal prior.’’ Given that the introduction of this prior is simply for the purposes of regularization, we are led directly to the idea of replacing the smoothness constraint model (4.13) by one of our tree models. Since in 2-D, solution of (4.12) corresponds to solving coupled partial differential equations [16], the use of a tree model and the resulting scale-recursive algorithm offer the possibility of substantial computational savings.

Let us consider a simplified, discretized, version of this problem, i.e., where  $f(x, t)$  is observed only at integer values of  $x$  and  $t$ . In this case we need to approximate  $(\partial f/\partial t)$  and  $(\partial f/\partial x)$ . For simplicity in our discussion, we consider the following finite difference approximations of these partial derivatives.

$$\frac{\partial f}{\partial t} \Big|_{x,t} \approx f(x, t+1) - f(x, t) \quad (4.15)$$

$$\frac{\partial f}{\partial x} \Big|_{x,t} \approx (f(x+1, t) - f(x-1, t))/2 \quad (4.16)$$

Obviously these rather crude approximations will lead to distortions (which we will see here), and more sophisticated methods can be used. In particular, as discussed in [32], the highest velocity that can be estimated depends both on the temporal sampling rate and the spatial frequency content of  $f(x, t)$ . In particular, low-pass spatial filtering of  $f(x, t)$  not only is useful for the reduction in noise in the derivatives (4.15), (4.16) but also in allowing the estimation of larger inter-frame spatial displacements via the differential brightness constraint (4.8). As we are simply interested in demonstrating the promise of our alternate formulation, we confine ourselves here to the simple approximation (4.15), (4.16).

We assume that our image is available at two time instants,  $t$  and  $t+1$ , where each image is uniformly sampled over a finite interval in space, i.e., we have  $f(i, t)$  and  $f(i, t+1)$  where  $i \in \{0, 1, 2, \dots, N-1\}$ . The discretized smoothing problem for our standard model is as follows:

$$v(i+1) - v(i) = w(i) \quad (4.17)$$

$$y(i) = -[f(x, i+1) - f(x, i)] = c(i)v(i) + r(i) \quad (4.18)$$

$$c(i) = f(i+1, t) - f(i, t) \quad (4.19)$$

where the white noises  $w(i)$  and  $v(i)$  have intensities 1 and  $\mu^{-1}$ , respectively. The solution to this smoothing problem is

given by

$$\hat{v} = (\mathcal{L} + \mu^{-1}C^T C)^{-1}(\mu^{-1}C^T)\bar{y} \quad (4.20)$$

where  $\hat{v}$  is the vector of velocity estimates for all  $i$ ,  $\bar{y}$  is the vector of measurements, and

$$C = \begin{bmatrix} c(0) & 0 & \dots & \dots & 0 \\ 0 & c(1) & 0 & \dots & 0 \\ \vdots & & \ddots & & \vdots \\ 0 & \dots & & c(N-3) & 0 \\ 0 & \dots & & 0 & c(N-2) \end{bmatrix} \quad (4.21)$$

and

$$\mathcal{L} = \begin{bmatrix} 1 & -1 & 0 & \dots & \dots & \dots & 0 \\ 0 & -1 & 2 & -1 & \dots & \dots & 0 \\ \vdots & & & \ddots & \ddots & & \vdots \\ 0 & \dots & \dots & 0 & -1 & 2 & -1 \\ 0 & \dots & \dots & & 0 & -1 & 1 \end{bmatrix} \quad (4.22)$$

Note that the matrix  $\mathcal{L}$ , which is a discrete approximation of the operator  $(d^2/dx^2)$ , is a result of the prior model we have chosen.

The multiscale model we use as our prior model for  $v(x)$ , as an alternative to our discretized standard model, is the tree model, (4.4) where  $t$  indexes the nodes of a finite tree with  $N$  points at the bottom level. Thus, the covariance  $P(\theta)$  of this zero-mean process at the bottom level of the tree is specified entirely by the parameter vector  $\theta = [a, p_0, \gamma, b]$ . To choose the parameters of our tree model so as to yield an approximation to the standard model, we fit the information matrix of our tree process,  $P^{-1}(\theta)$ , to  $\mathcal{L}$  by minimizing the matrix two-norm of the difference between  $\mathcal{L}$  and  $P^{-1}(\theta)$ .

$$\theta_{\text{fitted}} = \underset{\theta}{\text{argmin}} \|\mathcal{L} - P^{-1}(\theta)\|_2 \quad (4.23)$$

In comparing the performance of the multiscale smoother with the performance of the standard regularization method, we need a way of normalizing the problem. We define the following quantity, which can be thought of as the ratio between the information due to measurements and the information due to the model.

$$\Gamma \triangleq \frac{\text{trace}(\mu C^T C)}{\text{trace}(\mathcal{I})} \quad (4.24)$$

where  $\mathcal{I}$  is either  $\mathcal{L}$  or  $P^{-1}(\hat{\theta})$ . For our examples, we vary  $\Gamma$  by varying  $\mu$ .

Fig. 9 shows snapshots of the image of a translating sinusoid at times  $t$  and  $t+1$ , while Fig. 10(a) shows the result of estimating  $v$  based on standard regularization for  $\Gamma = 1, .1, .01$ , and Fig. 10(b) shows the result of estimating  $v$  based on our tree smoother for  $\Gamma = 1, .1, .01$ . The true value of  $v$  is a constant equal to three. The substantial deviations from the value of three are due to the inaccuracy of the approximations (4.15), (4.16). Note that the two approaches yield similar results. In fact, for all three values of  $\Gamma$  our tree smoother actually performs better. As we would expect by decreasing  $\Gamma$ , i.e., decreasing the weight  $\mu$  of the measurement

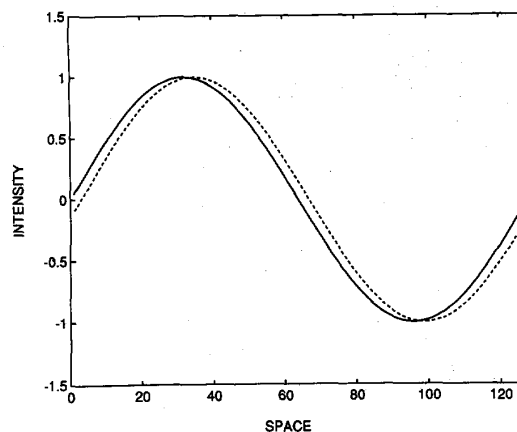


Fig. 9. Translating sinusoid at time  $t$  (solid) and at time  $t+1$  (dashed).

term in the cost function, the effect of the approximation error in (4.15), (4.16) is reduced, and smoother estimates result. Figs. 11(a) and (b) show similar behavior when the data of Fig. 9 is observed in noise. Again the performance is somewhat better for the tree model, but this is not our main point. Rather, what we have shown is that comparable results can be obtained with tree-model-based algorithms, and, given their considerable computational advantages, they offer an attractive alternative.

## V. CONCLUSION

In this paper we have developed a new method for multiresolution modeling of stochastic processes based on describing their scale-to-scale construction using dynamic models defined on dyadic trees. This framework allows us to describe a rich class of processes and also leads to an extremely efficient and highly parallelizable scale-recursive optimal estimation algorithm generalizing the Rauch-Tung-Striebel smoothing algorithm to the dyadic tree. This algorithm involves a variation on the Kalman filter in that, in addition to the usual measurement update and (fine-to-coarse) prediction steps, there is also a data fusion step. This in turn leads to a new Riccati equation, which we analyze in detail in [34] using several system-theoretic concepts for systems on trees. We have illustrated the potential of this framework in providing highly parallel algorithms for the smoothing of broad classes of stochastic processes, for the fusion of multiresolution data, and for the efficient solution of statistically regularized problems such as arise in computer vision.

We believe that this framework has considerable promise, and numerous directions for further work suggest themselves. In particular the extension and application of these ideas in 2-D offers numerous possibilities such as for the motion estimation problem described in Section IV. Also, there are several reasons to believe that our tree models can be used to describe a surprisingly rich class of stochastic processes. For example, a recent extension of wavelet transforms is the class of so-called wave packet transforms [10] in which both the fine and coarse resolution features are subject to further

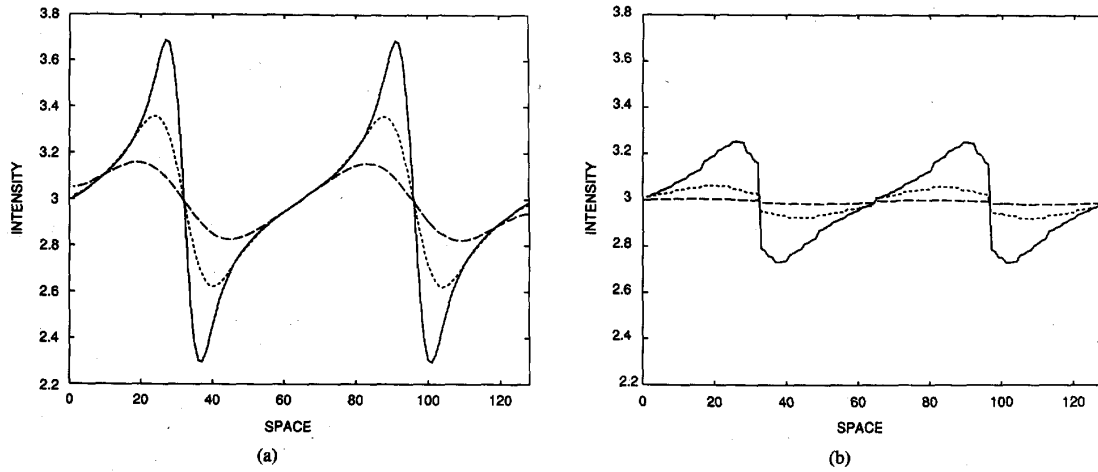


Fig. 10. (a) Estimate of  $v(x)$  using standard regularization for  $\Gamma = 1$  (solid), for  $\Gamma = .1$  (dashed), and for  $\Gamma = .01$  (long-dash); and (b) Estimate of  $v(x)$  using the tree smoother for  $\Gamma = 1$  (solid),  $\Gamma = .1$  (dashed), and  $\Gamma = .01$  (long-dash).

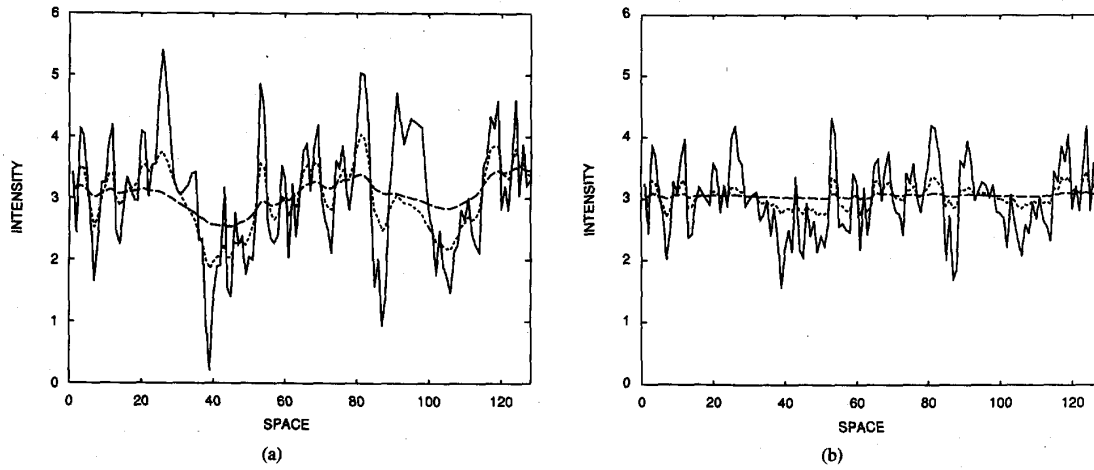


Fig. 11. (a) Estimate of  $v(x)$  for the translating sinusoid based on noisy data and using standard regularization for  $\Gamma = 1$  (solid), for  $\Gamma = .1$  (dashed), and for  $\Gamma = .01$  (long-dash); (b) Analogous estimates using the tree smoother.

decomposition. These correspond, in essence, to higher-order models in scale and thus to higher-dimensional versions of our state models. These and a variety of other examples offer intriguing possibilities for the future.

APPENDIX A

In this appendix we verify the formulae (3.15), (3.16) for the fusion of the estimates  $\hat{x}(t|t\alpha)$  and  $\hat{x}(t|t\beta)$  to produce  $\hat{x}(t|t+)$ . By definition

$$\hat{x}(t|t+) = E[x(t)|Y_{t\alpha}, Y_{t\beta}]. \tag{A.1}$$

By abuse of notation, let  $Y_{t\alpha}$  and  $Y_{t\beta}$  also denote vectors obtained by ordering the corresponding sets of random variables defined as in (3.1). Then, from our model (2.1), (2.2) we can decompose  $Y_{t\alpha}$  and  $Y_{t\beta}$  in the following way:

$$Y_{t\alpha} = M_{t\alpha}x(t) + \xi_1 \tag{A.2}$$

$$Y_{t\beta} = M_{t\beta}x(t) + \xi_2 \tag{A.3}$$

where the matrices  $M_{t\alpha}$  and  $M_{t\beta}$  contain products of  $A(s)$ ,  $m(s) > m(t)$ , and the vectors  $\xi_1$  and  $\xi_2$  are functions of the driving noises  $w(s)$  and the measurement noises  $v(s)$  for  $s$  in the subtrees strictly below  $t\alpha$  and  $t\beta$ , respectively, the latter fact implying  $\xi_1 \perp \xi_2$ . Let  $R_{t\alpha}$  and  $R_{t\beta}$  denote the covariances of  $\xi_1$  and  $\xi_2$ , respectively. Then, rewriting (A.2), (A.3) as

$$\mathcal{Y} = \mathcal{H}x(t) + \Xi \tag{A.4}$$

where

$$\mathcal{H} = \begin{bmatrix} M_{t\alpha} \\ M_{t\beta} \end{bmatrix}, \quad \Xi = \begin{bmatrix} \xi_1 \\ \xi_2 \end{bmatrix}, \quad \mathcal{R} = E[\Xi\Xi^T] \tag{A.5}$$

and  $x(t) \perp \Xi$ , we can write the optimal estimate of  $x(t)$  given

$\mathcal{Y}$  in the following way:

$$\begin{aligned} P(t|t+) &= [P_x^{-1}(t) + \mathcal{H}^T \mathcal{R}^{-1} \mathcal{H}]^{-1} \\ &= [P_x^{-1}(t) + M_{t\alpha}^T R_{t\alpha}^{-1} M_{t\alpha} + M_{t\beta}^T R_{t\beta}^{-1} M_{t\beta}]^{-1} \end{aligned} \quad (\text{A.6})$$

$$\begin{aligned} \hat{x}(t|t+) &= P(t|t+) \mathcal{H}^+ \mathcal{R}^{-1} \mathcal{Y} \\ &= P(t|t+) [M_{t\alpha}^T R_{t\alpha}^{-1} Y_{t\alpha} + M_{t\beta}^T R_{t\beta}^{-1} Y_{t\beta}]. \end{aligned} \quad (\text{A.7})$$

Similarly

$$\hat{x}(t|t\alpha) = P(t|t\alpha) M_{t\alpha}^T R_{t\alpha}^{-1} Y_{t\alpha} \quad (\text{A.8})$$

where

$$P(t|T\alpha) = [P_x^{-1}(t) + M_{t\alpha}^T R_{t\alpha}^{-1} M_{t\alpha}]^{-1} \quad (\text{A.9})$$

and analogous equations hold for  $\hat{x}(t|t\beta)$  and  $P(t|t\beta)$ . Equations (3.15), (3.16) then follow immediately from (A.6)–(A.9).

#### APPENDIX B

In this appendix, we verify the RTS smoothing recursions (3.17)–(3.19). The key to this is the following orthogonal decomposition of  $Y_0$  (i.e., the full set of all measurements at every node on the tree). Specifically, for each  $t$ ,  $Y_t$ , is given by (3.1), and we let  $Y_{\bar{t}}$  denote all the remaining measurements. Viewing these as vectors we define

$$\nu_{\bar{t}|t} = Y_{\bar{t}} - E[Y_{\bar{t}}|Y_t] \quad (\text{B.1})$$

so that  $\nu_{\bar{t}|t} \perp Y_t$  and the linear span of the set of all measurements, is given by

$$\text{span } Y_0 = \text{span } \{Y_t, Y_{\bar{t}}\} = \text{span } \{Y_t, \nu_{\bar{t}|t}\}. \quad (\text{B.2})$$

Then

$$\begin{aligned} \hat{x}_s(t) &= E[x(t)|Y_t, \nu_{\bar{t}|t}] \\ &= \hat{x}(t|t) + E[x(t)|\nu_{\bar{t}|t}]. \end{aligned} \quad (\text{B.3})$$

If we write  $x(t)$  as

$$x(t) = \tilde{x}(t|t) + \hat{x}(t|t) \quad (\text{B.4})$$

and note that

$$\hat{x}(t|t) \perp \nu_{\bar{t}|t} \quad (\text{B.5})$$

then we can write the following:

$$\hat{x}_s(t) = \hat{x}(t|t) + E[\tilde{x}(t|t)|\nu_{\bar{t}|t}]. \quad (\text{B.6})$$

Using the same argument on  $\hat{x}_s(t\bar{\gamma})$  allows us to write

$$\hat{x}_s(t\bar{\gamma}) = \hat{x}(t\bar{\gamma}|t) + E[\tilde{x}(t\bar{\gamma}|t)|\nu_{\bar{t}|t}]. \quad (\text{B.7})$$

Suppose the following equality were to hold

$$E[\tilde{x}(t|t)|\nu_{\bar{t}|t}] = L(E[\tilde{x}(t\bar{\gamma}|t)|\nu_{\bar{t}|t}]) \quad (\text{B.8})$$

where  $L$  is a matrix. Then (B.6) and (B.7) could be combined to yield

$$\hat{x}_s(t) = \hat{x}(t|t) + L[\hat{x}_s(t\bar{\gamma}) - \hat{x}(t\bar{\gamma}|t)]. \quad (\text{B.9})$$

We now proceed to show that (B.8) indeed holds and compute explicitly the matrix  $L$ . We begin with the following iterated expectation

$$E[\tilde{x}(t|t)|\nu_{\bar{t}|t}] = E[E[\tilde{x}(t|t)|\tilde{x}(t\bar{\gamma}|t), \nu_{\bar{t}|t}]] \nu_{\bar{t}|t}. \quad (\text{B.10})$$

We now examine the inner expectation,  $E[\tilde{x}(t|t)|\tilde{x}(t\bar{\gamma}|t), \nu_{\bar{t}|t}]$ , in detail. In particular note that  $Y_{\bar{t}}$  corresponds to measurements at all nodes outside of the subtree with root node  $t$ , i.e., at all nodes including  $t\bar{\gamma}$  and extending from it in either of the two directions other than toward  $t$  (toward  $t\bar{\gamma}^2$  or toward  $t\delta$ —see Fig. 1). Any such node is connected to  $t\bar{\gamma}$  by a backward segment (of possibly zero length) moving back to  $t\bar{\gamma}^r$  for some  $r > 0$ , followed by a forward segment moving down the other portion of the tree (i.e., a path of the form  $t\bar{\gamma}, t\bar{\gamma}^2, \dots, t\bar{\gamma}^{r-1}, t\bar{\gamma}^r, t\bar{\gamma}^{r-1}\delta, t\bar{\gamma}^{r-2}\delta, \dots$ ). By using the backward dynamics (2.9)–(2.12) for the backward portion of any such path and the forward model (2.1) for the forward segment, we can express  $x(s)$  for any  $s$  outside the tree as a linear combination of  $x(t\bar{\gamma})$ , backward white noises  $\tilde{W} = \{\tilde{w}(t\bar{\gamma}^r)|r > 0\}$  and forward white noises  $W = \{w(s)|s \text{ a descendant of } t\bar{\gamma}^r \delta \text{ for some } r \geq 1\}$ . Then, letting  $V$  denote the vector of measurement noises for  $Y_{\bar{t}}$ , we have that  $Y_{\bar{t}}$  has the following form

$$Y_{\bar{t}} = L_1 x(t\bar{\gamma}) + f(\tilde{W}, W) + V \quad (\text{B.11})$$

where  $f$  is a linear function of its arguments. Note that by construction  $\tilde{W}$  and  $W$  are orthogonal to  $x(t\bar{\gamma})$  and to everything beneath  $x(t\bar{\gamma})$ . The same is trivially true of  $V$ . Thus, in particular  $f(\tilde{W}, W) + V \perp Y_t$ , and substituting (B.11) into (B.1) then yields

$$\nu_{\bar{t}|t} = L_1 \tilde{x}(t\bar{\gamma}|t) + f(\tilde{W}, W) + V. \quad (\text{B.12})$$

This, together with the fact that  $f(\tilde{W}, W) + V \perp \tilde{x}(t|t)$ ,  $\tilde{x}(t\bar{\gamma}|t)$ , yields

$$E[\tilde{x}(t|t)|\tilde{x}(t\bar{\gamma}|t), \nu_{\bar{t}|t}] = E[\tilde{x}(t|t)|\tilde{x}(t\bar{\gamma}|t)]. \quad (\text{B.13})$$

But by using (2.9) and (3.9) (in the latter of these with  $t\alpha \mapsto t$  and  $t \mapsto t\bar{\gamma}$ ) we find that

$$E[\tilde{x}(t|t)|\tilde{x}(t\bar{\gamma}|t)] = J(t)\tilde{x}(t\bar{\gamma}|t) \quad (\text{B.14})$$

where  $J(t)$  is given by (3.18). This, together with (B.10) then yields

$$E[\tilde{x}(t|t)|\nu_{\bar{t}|t}] = J(t)E[\tilde{x}(t\bar{\gamma}|t)|\nu_{\bar{t}|t}]. \quad (\text{B.15})$$

This together with (B.8) and (B.9) then yields  $t\bar{\gamma}$  (3.17).

Finally, we can now easily derive a recursion for the smoothing error. Let

$$\tilde{x}_s(t) \triangleq x(t) - \hat{x}_s(t). \quad (\text{B.16})$$

Subtracting  $x(t)$  from both sides of (3.17) and rearranging terms we get

$$\tilde{x}_s(t) - J(t)\tilde{x}_s(t\bar{\gamma}) = \tilde{x}(t|t) - J(t)\tilde{x}(t\bar{\gamma}|t). \quad (\text{B.17})$$

By multiplying both sides of (B.17) on the right by its transpose and taking expectations, we get

$$P_s(t) + J(t)E[\hat{x}_s(t\bar{\gamma})\hat{x}_s^T(t\bar{\gamma})]J^T(t) = P(t|t) + J(t)E[\hat{x}(t\bar{\gamma}|t)\hat{x}^T(t\bar{\gamma}|t)]J^T(t) \quad (B.18)$$

$$P_s(t) = E[\hat{x}_s(t)\hat{x}_s^T(t)] \quad (B.19)$$

where we have relied on the fact that

$$E[\hat{x}_s(t)\hat{x}_s^T(t\bar{\gamma})] = 0 \quad (B.20)$$

$$E[\hat{x}(t|t)\hat{x}^T(t\bar{\gamma}|t)] = 0. \quad (B.21)$$

And finally, since

$$E[\hat{x}_s(t\bar{\gamma})\hat{x}_s^T(t\bar{\gamma})] = P_x(t\bar{\gamma}) - P_s(t\bar{\gamma}) \quad (B.22)$$

$$E[\hat{x}(t\bar{\gamma}|t)\hat{x}^T(t\bar{\gamma}|t)] = P_x(t\bar{\gamma}) - P(t\bar{\gamma}|t) \quad (B.23)$$

we obtain (3.19).

REFERENCES

[1] B. Anderson and T. Kailath, "Forwards, backwards, and dynamically reversible Markovian models of second-order processes," *IEEE Trans. Circuits Syst.*, vol. 26, no. 11, pp. 956-965, 1978.

[2] M. Basseville, A. Benveniste, and A. S. Willsky, "Multiscale autoregressive processes, Part I: Schur-Levinson parameterizations," *IEEE Trans. Sign. Proc.*, vol. 40, no. 8, pp. 1915-1934, Aug. 1992.

[3] —, "Multiscale autoregressive processes, part II: Lattice structures for whitening and modeling," *IEEE Trans. Sig. Proc.*, vol. 40, no. 8, pp. 1935-1954, Aug. 1992.

[4] M. Basseville, A. Benveniste, A. S. Willsky, and K. C. Chou, "Multiscale statistical processing: Stochastic processes indexed by trees," in *Proc. of Int'l Symp. on Math Theory Networks Syst.*, Amsterdam, June 1989.

[5] A. Benveniste, R. Nikoukchah, and A. S. Willsky, "Multiscale system theory," in *Proc. 29th IEEE Conf. Decis. Contr.*, Honolulu, HI, Dec. 1990.

[6] G. Beylkin, R. Coifman, and V. Rokhlin, "Fast wavelet transforms and numerical algorithms I," *Comm. Pure and Appl. Math.*, vol. 44, pp. 141-183, Mar. 1991.

[7] P. Burt and E. Adelson, "The Laplacian pyramid as a compact image code," *IEEE Trans. Comm.*, vol. 31, pp. 482-540, 1983.

[8] K. C. Chou, S. Golden, and A. S. Willsky, "Modeling and estimation of multiscale stochastic processes," in *Proc. Int'l Conf. Acoustics, Speech, and Sig. Proc.*, Toronto, Apr. 1991.

[9] S. C. Clippingdale and R. G. Wilson, "Least squares image estimations on a multiresolution pyramid," in *Proc 1989 Int'l Conf. on Acoustics, Speech, and Sig. Proc.*

[10] R. R. Coifman and M. V. Wickenhauser, "Entropy-based algorithms for best basis selection," *IEEE Trans. Info. Theory*, vol. 38, no. 2, pp. 713-718, Mar. 1992.

[11] I. Daubechies, "Orthonormal bases of compactly supported wavelets," *Comm. Pure and Applied Math.*, vol. 91, pp. 909-996, 1988.

[12] I. Daubechies, "The wavelet transform, time-frequency localization and signal analysis," *IEEE Trans. Info. Theory*, vol. 36, pp. 961-1005, 1990.

[13] P. Flandrin, "Wavelet analysis and synthesis of fractional Brownian motion," *IEEE Trans. Info. Theory*, vol. 38, no. 2, pp. 910-917, Mar. 1992.

[14] S. Golden, "Identifying multiscale statistical models using the wavelet transform," S.M. Thesis, MIT Dep. Elec. Eng. Comp. Sci., Cambridge, MA, May 1991.

[15] A. Grossman and J. Morlet, "Decomposition of Hardy functions into square integrable wavelets of constant shape," *SIAM J. Math. Anal.*, vol. 15, pp. 723-736, 1984.

[16] B. Horn and B. Schunck, "Determining optical flow," *Artificial Intell.*, vol. 17, pp. 185-203, 1981.

[17] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Mach. Intel.*, vol. 11, pp. 674-693, Jul. 1989.

[18] S. G. Mallat, "Multifrequency channel decompositions of images and wavelet models," *IEEE Trans. ASSP*, vol. 37, pp. 2091-2110, Dec. 1989.

[19] Y. Meyer, "L'analyse par ondelettes," *Pour la Science*, Sep. 1987.

[20] H. E. Rauch, F. Tung, and C. T. Striebel, "Maximum likelihood estimates of linear dynamic systems," *AIAA J.*, vol. 3, no. 8, pp. 1445-1450, Aug. 1965.

[21] A. Rougee, B. Levy, and A. Willsky, "An estimation-based approach to the reconstruction of optical flow," MIT, Cambridge, MA, Lab. Info. Decision Systems Tech. Rep., LIDS-P-1663, Apr. 1987.

[22] M. J. Smith and T. P. Barnwell, "Exact reconstruction techniques for tree-structured subband coders," *IEEE Trans. ASSP*, vol. 34, pp. 434-441, 1986.

[23] G. Strang, "Wavelets and dilation equations: A brief introduction," *SIAM Rev.*, vol. 31, no. 4, pp. 614-627, Dec. 1989.

[24] A. H. Tewfik and M. Kim, "Correlation structure of the discrete wavelet coefficients of fractional Brownian motions," *IEEE Trans. Info. Theory*, vol. 38, no. 2, pp. 904-909, Mar. 1992.

[25] M. Todd and R. Wilson, "An anisotropic multi-resolution image data compression algorithm," in *Proc. 1989 Int'l Conf. on Acoustics, Speech, and Sig. Proc.*, 1989.

[26] G. C. Verghese and T. Kailath, "A further note on backwards Markovian models," *IEEE Trans. on Info. Theory*, vol. 25, pp. 121-124, 1979.

[27] M. Vetterli and C. Herley, "Wavelet and filter banks: Relationships and new results," in *Proc. ICASSP*, Albuquerque, NM, 1990.

[28] G. W. Wornell, "A Karhunen-Loeve-like expansion for 1/f processes via wavelets," *IEEE Trans. Info. Theory*, vol. 36, no. 9, pp. 859-861, Jul. 1990.

[29] G. W. Wornell and A. V. Oppenheim, "Estimation of fractal signs from noisy measurements using wavelets," *IEEE Trans. Sig. Proc.*, vol. 40, no. 3, pp. 611-623, Mar. 1992.

[30] A. Witkin, D. Terzopoulos and M. Kass, "Signal matching through scale space," *Int. J. Comp. Vision*, vol. 1, pp. 133-144, 1987.

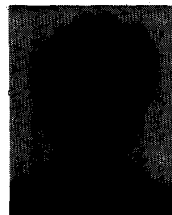
[31] R. B. Washburn and A. S. Willsky, "Optional sampling of submartingales on partially-ordered sets," *Annals of Prob.*, vol. 9, no. 6, pp. 957-970, 1981.

[32] T. M. Chin, "Dynamic estimation in computational vision," Ph.D. dissertation, MIT Dept. of EECS, Cambridge, MA, Oct. 1991.

[33] K. C. Chou, "A stochastic modeling approach to multiscale signal processing," MIT Dept. Elec. Eng. Comp. Sci., Cambridge, MA, Ph.D. dissertation, May 1991.

[34] K. C. Chou, A. S. Willsky, and R. Nikoukchah, "Multiscale systems, Kalman filters, and Riccati equations," to be submitted for publication.

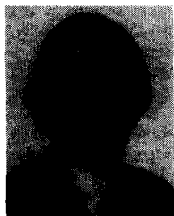
[35] M. Basseville, A. Benveniste, K. C. Chou, S. A. Golden, R. Nikoukchah, and A. S. Willsky, "Modeling and estimation of multiresolution stochastic processes," *IEEE Trans. Info. Theory*, vol. 38, no. 2, pp. 766-784, Mar. 1992.



**Kenneth C. Chou** (S'90-M'91) received the S.B., S.M., E.E., and Ph.D. degrees in electrical engineering from the Massachusetts Institute of Technology, (MIT) Cambridge, MA in 1985, 1987, 1987, and 1991, respectively.

He is currently a research engineer at SRI International in the Acoustics and Radar Technology Laboratory. His current research interests include detection and estimation theory involving multiscale stochastic processes, multiscale system identification, and applications of multiscale signal processing to machine monitoring and radar imaging. His interests also include MIMO robust control of active noise control systems.

Dr. Chou is a member of Tau Beta Pi, Eta Kappa Nu, and Sigma Xi.



**Alan S. Willsky** (S'70, M'73, SM'82, F'86) received both the S.B. degree and the Ph.D. degree from the Massachusetts Institute of Technology (MIT), Cambridge, MA, in 1969 and 1973, respectively.

He joined the MIT faculty in 1973 and his present position is Professor of Electrical Engineering. From 1974 to 1981 he served as Assistant Director of the MIT Laboratory for Information and Decision Systems. He is also a founder and member of the board of directors of Alphatech, Inc. In 1975

he received the Donald P. Eckman Award from the American Automatic Control Council. He has held visiting positions at Imperial College, London, L'Université de Paris-Sud, and the Institut de Recherche en Informatique et Systèmes Aléatoires in Rennes, France. He was program chairman for the 17th IEEE Conference on Decision and Control, has been an associate editor of several journals including the IEEE TRANSACTIONS ON AUTOMATIC CONTROL, has served as a member of the Board of Governors and Vice President for Technical Affairs of the IEEE Control Systems Society, was program chairman for the 1981 Bilateral Seminar on Control Systems held in the People's Republic of China, and was special guest editor of the 1992 special issue of the IEEE TRANSACTIONS ON INFORMATION THEORY on wavelet transforms and multiresolution signal analysis. Also in 1988 he was made a Distinguished Member of the IEEE Control Systems Society. In addition, he has given several plenary lectures at major scientific meetings including the 20th IEEE Conference on Decision and Control, the 1991 IEEE International Conference on Systems Engineering, the SIAM Conference on Applied Linear Algebra, 1991, and the 1992 Inaugural Workshop for the National Centre for Robust and Adaptive Systems, Canberra, Australia.

Dr. Willsky is the author of the research monograph *Digital Signal Processing and Control and Estimation Theory* and is co-author of the undergraduate text *Signals and Systems*. He was awarded the 1979 Alfred Noble Prize by the American Society of Civil Engineers and the 1980 Browder J. Thompson Memorial Prize Award by the IEEE for a paper excerpted from his monograph. His present research interests are in problems involving multidimensional and multiresolution estimation and imaging, discrete-event systems, and the asymptotic analysis of control and estimation systems.



**Albert Benveniste** (M'81, SM'89, F'91) was born May 8, 1949 in Paris, France. In 1971 he graduated from Ecole des Mines de Paris. He performed his Thèse d'Etat in Mathematics, probability theory, in 1975.

From 1976 to 1979 he was Associate Professor in mathematics at Université de Rennes I. From 1979 to now he has been Directeur de Recherche at INRIA.

In 1980, Benveniste was co-winner of the IEEE TRANSACTIONS ON AUTOMATIC CONTROL Best Trans-

action Paper Award for his paper on blind deconvolution in data communications. In 1990 he received the CNRS silver medal, and in 1991 he was elected IEEE fellow. From 1986 to 1990 he was vice-chairman of the IFAC Committee on Theory and was chairman of this committee from 1991-1993. From 1987 to 1990 he was associate editor for IEEE TRANSACTIONS ON AUTOMATIC CONTROL. He is currently associate editor for *International Journal of Adaptive Control and Signal Processing*, *International Journal of Discrete Event Dynamical Systems* and Associate Editor at Large for IEEE TRANSACTIONS ON AUTOMATIC CONTROL. He has coauthored with M. Métivier and P. Priouret the book *Adaptive Algorithms and Stochastic Approximations* and has been an editor, jointly with Michèle Basseville, of the collective monograph *Detection of Abrupt Changes in Signals and Systems*. His current interests include computer science in the area of real-time languages and systems.