Multiscale Methods for the Segmentation and Reconstruction of Signals and Images

Michael K. Schneider, *Student Member, IEEE*, Paul W. Fieguth, *Member, IEEE*, William C. Karl, *Member, IEEE*, and Alan S. Willsky, *Fellow, IEEE*

***Abstract*—This paper addresses the problem of both segmenting and reconstructing a noisy signal or image. The work is motivated by large problems arising in certain scientific applications, such as medical imaging. Two objectives for a segmentation and denoising algorithm are laid out: it should be computationally efficient and capable of generating statistics for the errors in the reconstruction and estimates of the boundary locations. The starting point for the development of a suitable algorithm is a variational approach to segmentation [1]. This paper then develops a precise statistical interpretation of a one-dimensional (1-D) version of this variational approach to segmentation. The 1-D algorithm that arises as a result of this analysis is computationally efficient and capable of generating error statistics. A straightforward extension of this algorithm to two dimensions would incorporate recursive procedures for computing estimates of inhomogeneous Gaussian Markov random fields. Such procedures require an unacceptably large number of operations. To meet the objective of developing a computationally efficient algorithm, the use of recently developed multiscale statistical methods is investigated. This results in the development of an algorithm for segmenting and denoising which is not only computationally efficient but also capable of generating error statistics, as desired.**

***Index Terms*—Denoising, multiscale statistical models, segmentation.**

I. Introduction

MUMFORD and Shah have developed a theoretical framework in which to address the problem of simultaneous image denoising and segmentation [2], [3]. In this framework, the goal is to decompose a given noisy image into piecewise smooth regions bounded by contours on which the image intensity is allowed to change abruptly. This is accomplished by minimizing a particular functional jointly over the image boundaries and a reconstruction of the image. The minimizing reconstruction of the original image has been denoised through smoothing everywhere except along the detected boundary contours. The Mumford and Shah functional has many nice mathematical and psychovisual properties [4]; however, it is difficult to compute minimizers because of the discrete nature of the image boundary terms.

Various people have tried to address these computational difficulties by making small alterations in the Mumford and Shah framework [5]–[7]. In this revised setting, there are also two objects computed by the algorithm: a reconstruction of the original image and a continuous-valued edge-strength function. The reconstruction is a denoised version of the unprocessed, noisy image which does not suffer from the edge blurring effects of some simple linear reconstruction algorithms (e.g., low-pass filtering). The edge-strength function provides information about the optimal spatially-varying amount of smoothing that should be applied to produce the reconstruction. The function varies between the values 0 and 1, taking on the value 1 where no smoothing should be done and 0 in areas where full smoothing is performed. While the edge-strength function itself is not an explicit estimate of image edges, it is demonstrably a more robust indicator of edge likelihood than standard gradient maps. In particular, the edge-strength function displays substantial robustness to noise and automatically avoids problems of dynamic range exhibited by gradient maps of noisy images (see Section VI). As a consequence, generating explicit edge contours from edge-strength functions can be accomplished robustly as shown in [1] using thresholding and [8] using curve evolution.

The approach to segmentation and denoising taken in this paper begins with a novel Bayesian interpretation of the revised Mumford and Shah variational approach to segmentation. This interpretation follows in the footsteps of some recent work in which close connections are made between certain variational and statistical approaches to image processing [9], [10]. The principal advantage of the Bayesian framework is that it provides a theoretical structure for the interpretation and computation of error statistics. Error statistics provide a quantitative measure of the quality of the reconstruction and estimate of the edge-strength function. Such a measure of quality is very important in certain scientific applications. One of the main contributions of this paper is the development of a segmentation algorithm which produces not only a reconstruction of the image and an estimate of the edge-strength function but also error statistics. Furthermore, this paper contains a careful evaluation of the nature and quality of the information provided by these error statistics.

While the Bayesian interpretation can be equally well applied to both one and two-dimensional (2-D) signals, there is a significant difference in computational complexity in solving the

Manuscript received January 13, 1997; revised August 25, 1999. This material is based upon work supported by a National Science Foundation Graduate Research Fellowship, by ONR under Grant N00014-91-J-1004, by AFOSR under Grants F49620-98-1-0349 and F49620-96-1-0028, by Boston University under Grant GC123919NGN, and by NIH under Grant NINDS 1 R01 NS34189. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Jeffrey J. Rodriguez.

M. K. Schneider and A. S. Willsky are with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: mikesch@mit.edu).

P. W. Fieguth is with the Department of Systems Design Engineering, University of Waterloo, Waterloo, Ont., Canada N2L 3G1.

W. C. Karl is with the Department of Electrical and Computer Engineering and the Department of Biomedical Engineering, Boston University, Boston, MA 02215 USA.

Publisher Item Identifier S 1057-7149(00)01512-8.

resulting problems. As a consequence, a thorough analysis of the problem is first presented in a one-dimensional (1-D) setting. Markov random field prior (MRF) models appear in the corresponding 2-D Bayesian estimation problems. Unfortunately, algorithms for solving estimation problems involving 2-D MRF priors require a large number of computations to generate exact estimates and a prohibitively large computational load to calculate error statistics. Our objective here is to develop an algorithm with constant per-pixel complexity that also produces useful error statistics.

There are two possible approaches to achieving such an objective, namely *approximating the solution* (i.e., replacing the solution to the estimation problem with one which is easier to compute) or *approximating the problem* (i.e., replacing the estimation problem with one which has similar characteristics but which can be solved exactly using an efficient algorithm). An approach of the former type is described in [11]. In this paper, an approach of the latter type is developed by altering the prior model appearing in the problem formulation. In particular, this paper examines the usefulness of multiscale prior models for image segmentation. Multiscale models, which were introduced and studied in [9], admit algorithms with constant per-pixel complexity for the calculation of both estimates and error variances. They have also been shown to be useful in defining alternative approaches to problems in computer vision which are often posed in a variational context [9], [10], [12], [13]. These previous investigations, however, dealt with problems that resulted in linear estimation algorithms. In contrast, image segmentation is fundamentally a nonlinear problem, and thus, this paper represents the first work on using multiscale stochastic models to solve a nonlinear problem in image processing and computer vision. The algorithm that results not only has a modest computational load but also yields good performance.

The work presented in this paper builds primarily upon the two areas of variational methods for segmentation and multiscale methods for image processing. Section II summarizes the relevant material on variational methods for segmentation. Section III provides an overview of the multiscale modeling framework. The overview is not meant to be comprehensive, however, and many details are not discussed but may be found in the literature [9], [10], [12]–[15]. The subsequent Sections IV and V discuss the specifics of the 1-D scenario, and Section VI is devoted to the 2-D scenario.

II. Variational Methods in Image Segmentation

A family of functionals proposed by Ambrosio and Tortorelli for image segmentation and denoising [5], [6] lies at the core of a segmentation and denoising algorithm developed by Shah [1] and extended by Pien and Gauch [7] among others. A member of this family of functionals, parameterized by ρ, is of the form

$$E(f,s) = \int \int_{\Omega} \left\{ r^{-1}(g-f)^2 + \lambda |\nabla f|^2 (1-s)^2 + \frac{\beta}{2}\left(\rho |\nabla s|^2 + \frac{s^2}{\rho}\right)\right\} dx\, dy \tag{1}$$

where $\Omega \subset \mathbf{R}^2$ is the image domain, $g\colon \Omega \to \mathbf{R}$ is the image data, $f\colon \Omega \to \mathbf{R}$ is a piecewise smooth approximation to g, and $s\colon \Omega \to [0,1]$ is an estimate of the edge-strength function, indicating high probability for the presence of an edge where it takes values close to one. The first and second terms constrain the approximating surface f to match the data as best as possible and also to be smooth in those places where s is close to zero. The third term ensures that s remain reasonably smooth and does not tend to 1 everywhere. As shown in [6], the minima of (1) converge to a minimum of the Mumford and Shah functional as $\rho \to 0$. The general approach Shah and Pien use to minimize (1) is coordinate descent: one alternates between fixing s and minimizing

$$\int \int_{\Omega} (r^{-1}(g-f)^2 + \lambda |\nabla f|^2 (1-s)^2)\, dx\, dy \tag{2}$$

over possible f, and fixing f and minimizing

$$\int \int_{\Omega} \left(\lambda |\nabla f|^2 (1-s)^2 + \frac{\beta}{2}\left(\rho |\nabla s|^2 + \frac{s^2}{\rho}\right)\right) dx\, dy \tag{3}$$

over possible s. Based on empirical evidence, Shah [1] and Pien and Gauch [7] have noted that this coordinate descent scheme converges to a reasonable solution and that the results are not significantly affected by the initial condition or whether one starts by estimating f or s.

III. Multiscale Models

Consider a prototypical quadratic minimization problem: minimize

$$E(f) = r^{-1}||g-f||^2 + \lambda ||Lf||^2 \tag{4}$$

where f and g are vectors consisting of a lexicographic ordering of pixels in an image and L is a matrix chosen to ensure that the minimizer of (4) is smooth (e.g., L could take first differences of nearest neighbors as an approximation of a derivative). Minimization of (4) is equivalent to maximization of

$$p(f,g) = e^{-(r^{-1}||g-f||^2 + \lambda ||Lf||^2)}. \tag{5}$$

One can now view $p(f,g)$ as a joint Gaussian probability distribution. In fact $p(f,g)$ is the joint distribution for f and g given by the measurement and model equations

$$g = f + \sqrt{r}v \quad \sqrt{\lambda} Lf = w \tag{6}$$

where v and w are independent zero-mean Gaussian random vectors with identity covariance. For a given value of g, the maximum of $p(f,g)$ over f occurs at the conditional mean of the Gaussian, $\mathrm{E}[f|g]$. What's more, $\mathrm{E}[f|g]$ is the Bayes least-squares estimate of f, according to the distribution induced by the modeling equations (6). Thus, one can view the problem of minimizing (4) from the perspective of optimization or of statistical estimation [9], [10], [16].

The main advantage of the Bayesian interpretation is that it casts the problem into a probabilistic framework in which it is natural to examine the accuracy of the resulting estimates. This is especially relevant in scientific applications such as remote

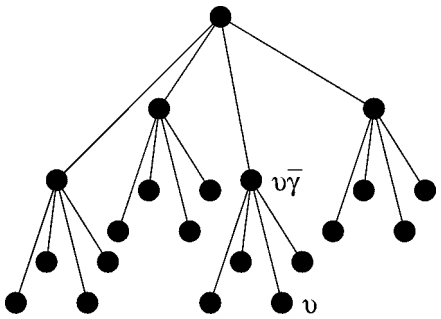


Fig. 1. Nodes of a tree, such as the quad-tree pictured here (typically one uses a binary tree for 1-D applications and a quad-tree for 2-D applications), are the index sets of the multiscale processes discussed in this paper. The operation of $\overline{\gamma}$ on an index is described as follows: if ν is the index of some node, then $\nu\overline{\gamma}$ denotes the parent of that node.

sensing, in which one may be, for example, assessing if features in a reconstruction are meaningful or statistically insignificant artifacts. In addition, this statistical formulation brings into focus the role played by the regularization term as a prior model, opening up the possibility of using alternate models that offer certain advantages.

Specifically, one can consider modeling a 1-D or 2-D phenomenon as the finest scale of a stochastic process on a tree. Doing so provides important computational advantages when performing estimation. The tree modeling framework used in this paper was introduced in [14] and further developed in [9], [10], [12], [13], [15]. In these references, one can find more detailed discussions of the following concepts.

The multiscale processes that are of interest in this paper are specified in terms of an autoregression on a tree (see Fig. 1). The root-to-leaf recursion takes the form

$$x(\nu) = A(\nu)x(\nu\overline{\gamma}) + B(\nu)w(\nu) \tag{7}$$

where the notation $\nu\overline{\gamma}$ refers to the parent of node ν. This recursion is a generalization of the standard state-space recursion for modeling a phenomenon evolving in time. In (7), the $w(\nu)$ and the state $x(\text{root})$ at the root node are independent zero-mean Gaussian random vectors, the w's with identity covariance and $x(\text{root})$ with prior covariance P_{root}. The A and B matrices are deterministic quantities which define the statistics of the process on the tree. Observations $y(\nu)$ of the state variables have the form

$$y(\nu) = C(\nu)x(\nu) + v(\nu) \tag{8}$$

where the $v(\nu)$ are independent Gaussian random vectors, and the matrices $C(\nu)$ are deterministic. The least-squares estimates of process values at all nodes on the tree given all observations and the associated error variances can be calculated with an $O(N)$ recursive algorithm [14], [17], where N is the number of finest scale nodes on the tree. The algorithm consists of a fine-to-coarse recursion in which data in successively larger subtrees are fused up to the root node of the tree, and a subsequent coarse-to-fine recursion which produces both the optimal estimates and their error covariances. This algorithm generalizes the standard Kalman filter and Rauch–Tung–Striebel smoother.

The class of processes representable as the finest scale of a tree process includes some very important processes. In particular, Luettgen [18] has shown that any 1-D Gauss–Markov process can be represented on a tree using a model with a three-dimensional (3-D) state. The 1-D results in this paper make use of this model. The 2-D results make use of and extend the $1/f$-like model used previously for the computation of optical flow and ocean surface reconstruction [10], [13]. This $1/f$-like model and its extension are presented in Section VI.

IV. Statistical Interpretation of Segmentation in One Dimension

The first step in applying the multiscale modeling framework to segmentation in 1-D is to develop a statistical interpretation of the discretized version of (1). One possibility for discretizing this functional is to replace the functions $f(x), g(x)$, and $s(x)$ with regularly spaced collections of samples $f(i), g(i)$, and $s(i)$; the integrals with sums over i; and the derivatives with first differences. The result in 1-D is the discrete functional

$$\begin{aligned} E(f,s) = {} & r^{-1}\sum_{i=1}^{n}(f(i)-g(i))^2 \\ & + \lambda\sum_{i=1}^{n-1}(1-s(i))^2(f(i+1)-f(i))^2 \\ & + \frac{\beta}{2}\left(\rho\sum_{i=1}^{n-2}(s(i+1)-s(i))^2 + \frac{1}{\rho}\sum_{i=1}^{n-1}s(i)^2\right) \end{aligned} \tag{9}$$

where n denotes the number of data points $g(i)$[1]. As was done in [1], [7], one can use coordinate descent to minimize (9), thereby decomposing this complex problem into two simpler ones.

The problem of fixing s and finding the f that minimizes (9) is equivalent to finding the f that minimizes the discrete functional

$$\begin{aligned} E_s(f) = {} & r^{-1}\sum_{i=1}^{n}(f(i)-g(i))^2 \\ & + \sum_{i=1}^{n-1}\lambda(1-s(i))^2(f(i+1)-f(i))^2. \end{aligned} \tag{10}$$

A slightly more compact form can be written by collecting the samples $f(i), g(i)$, and $s(i)$ into vectors f and $g \in \mathbf{R}^n$ and $s \in \mathbf{R}^{n-1}$. Specifically, let L_n be the the $(n-1)\times n$ matrix that takes first differences of n samples, and let $S = \text{diag}\ (1-s_1, \cdots, 1-s_{n-1})$. Then, (10) simplifies to $E_s(f) = \|f-g\|^2_{r^{-1}I} + \lambda\|L_n f\|^2_{S^T S}$, where $\|x\|^2_W = x^T W x$. Finding the minimum of E_s for fixed invertible S is then equivalent to finding the least-squares estimate of f assuming the following measurement and prior model

$$g = f + \sqrt{r}v_f \quad L_n f = \frac{1}{\sqrt{\lambda}}S^{-1}w_f \tag{11}$$

[1]In this discretization, s has a length which is one sample less than that of f. This is because the samples of the discretized s lie between the samples of the discretized f. More details are provided in [19].

where v_f and w_f are independent zero-mean Gaussian random vectors with identity covariance. Notice that for i such that $s(i) \approx 1$, the multiplier of $w_f(i), 1/(1-s(i))$, is very large. Thus, at these locations, the variance of $f(i+1)-f(i)$ in the prior model is high, and a least-squares estimator will allow big jumps to occur in the estimate of f. This is exactly what one wants the estimator to do at edge locations.

The problem of fixing f and finding s that minimizes (9) is equivalent to minimizing

$$\lambda \sum_{i=1}^{n-1} (f(i+1)-f(i))^2(1-s(i))^2 + \frac{\beta\rho}{2} \sum_{i=1}^{n-2} (s(i+1)-s(i))^2 + \frac{\beta}{2\rho} \sum_{i=1}^{n-1} s(i)^2. \quad (12)$$

Defining $a(i) = \lambda(f(i+1)-f(i))^2$, $b = \beta/2\rho, c = \beta\rho/2$, and $\gamma(i) = a(i)/(a(i)+b)$, one finds that, after completing the square, (12) can be rewritten

$$\sum_{i=1}^{n-1}((a(i)+b)(\gamma(i)-s(i))^2 + a(i)(1-\gamma(i)^2) - b\gamma(i)^2) + c \sum_{i=1}^{n-2} (s(i+1)-s(i))^2. \quad (13)$$

Ignoring terms which do not depend on s, one observes that minimizing (12) over s is equivalent to minimizing

$$E_f(s) = \sum_{i=1}^{n-1} (a(i)+b)(\gamma(i) - s(i))^2 + c \sum_{i=1}^{n-2} (s(i+1)-s(i))^2. \quad (14)$$

By defining the diagonal matrix

$$A = \text{diag}\,(\sqrt{\lambda(L_n f)_1 + b}, \cdots, \sqrt{\lambda(L_n f)_{(n-1)} + b})$$

and the vector

$$\gamma = \left(\frac{\lambda(L_n f)_1^2}{\lambda(L_n f)_1^2 + b} \cdots \frac{\lambda(L_n f)_{(n-1)}^2}{\lambda(L_n f)_{(n-1)}^2 + b} \right)^T$$

where $(L_n f)_k$ corresponds to the kth row of $L_n f$, one can rewrite (14) as

$$E_f(s) = \|\gamma - s\|^2_{A^T A} + c\|L_{n-1}s\|^2. \quad (15)$$

In the original functional (1), s is constrained to lie within [0,1]. If one removes this constraint, the problem of finding the s that minimizes (15) is equivalent to the problem of estimating s given the following measurement and prior model:

$$\gamma = s + A^{-1}v_s \quad L_{n-1}s = \frac{1}{\sqrt{c}} w_s \quad (16)$$

where v_s and w_s are independent zero-mean Gaussian random vectors with identity covariance. Notice that γ plays the role of an observation of s and that its components take on values near one where the difference between consecutive samples of f is large and near zero where the difference is small. Observe also that $\gamma(i)$ lies within [0,1); thus, the first term in (15) provides an increased penalty for s that does not stay within [0,1]. This is desirable because a solution to the unconstrained minimization of (15) that lies within [0,1] is an optimal solution of the constrained problem. As it turns out, this is often the case, as discussed in Section V.

As an aside, we note that one of the benefits of formulating a minimization problem in terms of statistics is that it yields natural interpretations of the parameters. These interpretations, in turn, can be used to form a loose set of guidelines for picking parameter values suitable for a particular segmentation application. Specifically, simple calculations lead to the following observations which can be used to pick the signal-dependent parameters λ, b, c, and r.

- λ is inversely proportional to the variability in the reconstructed signal f at locations where no edges are present. In particular, $\mathrm{E}[(f(i+1)-f(i))^2 | s(i)=0] = 1/\lambda$.
- r is the measurement noise variance term, determined in many applications by sensor specifications.
- c is proportional to the width of the edges. Specifically, the model for the edge-strength function s implies $\mathrm{E}(s(i+1)-s(i))^2 = 1/c$.
- b controls the degree of edginess. In particular, choosing a value for it is related to the issue of defining what level of variability in f we wish to call an edge. Since the observations appearing in the model for the edge-strength function s have the form

$$\gamma(i) = \frac{\lambda(f(i+1)-f(i))^2}{\lambda(f(i+1)-f(i))^2 + b}$$

one desires that for i at edge locations

$$b \gg \lambda\, \mathrm{E}[(f(i+1)-f(i))^2 | s(i)=0] = 1,$$
$$b \ll \lambda(f(i+1)-f(i))^2.$$

These rules guided the choice of parameter values used for the numerical results presented in the subsequent section. More discussion concerning the parameters, including Monte Carlo results, can be found there and in [19].

V. Numerical Results

Based on the Gauss-Markov estimation problem formulations specified by (11) and (16), one can compute estimates $\hat{f}$ and $\hat{s}$ using any one of a variety of efficient methods. These include direct methods for solving the associated normal equations and Kalman filter smoothing. For the simulation results that follow, estimates $\hat{f}$ and $\hat{s}$ as well as error variances P_f and P_s were computed by a multiscale recursive estimation algorithm [13], [17], [18] (see also Section III). One detail concerning the implementation of this and other algorithms is that they require the specification of prior variances P_0 on $f(0)$ and $s(0)$, the first samples of the probabilistic models for f and s. However, the precise interpretation of the variational formulation as an estimation problem corresponds to viewing the initial value as unknown, which is equivalent to an infinitely large prior variance. While it is possible to accommodate this in the estimation formulation with no effect on algorithmic complexity, it is common to use an alternate approach in which one closely approximates the solution to the original problem by setting the prior covariance P_0 to a relatively large number. For the purposes of this

TABLE I
DESCRIPTION OF PARAMETERS IN 1-D SEGMENTATION ALGORITHM

Parameter	*Description*	*Value for the Results in Figs.* 2 and 4	3
λ	λ adjusts the smoothness in f away from edges. See (16).	1	25000
b	b affects the edginess of the estimate of the edge-strength function. See (16).	10	25
c	c sets the allowed variability in s. See (16).	100	1
r	r is the assumed noise variance in the data	1	$(0.2)^2$
P_0	P_0 is the prior covariance for initial process values.	100	
ϵ	Estimates of s are clipped to lie within $[0, 1-\epsilon]$.	1.0×10^{-4}	
Δ	The algorithms stops after the percent change of the functional (9) falls below Δ.	0.01%	

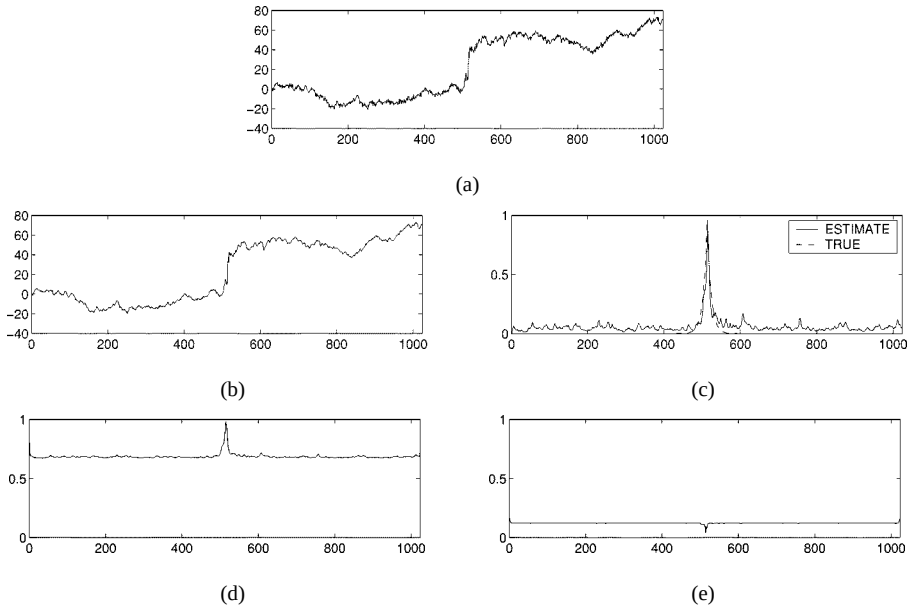


Fig. 2. The results for segmenting the data pictured in (a), a noisy observation of a process whose statistics are dictated by (11) for the true edge-strength function given by (17). The measurement noise is white and Gaussian with unit variance. All parameters are set as in Table I. (a) Data g, (b) reconstruction $\hat{f}$, (c) estimate of the edge-strength function $\hat{s}$; (d) reconstruction error standard deviations $\sqrt{P_f}$; and (e) estimate of edge-strength function error standard deviations $\sqrt{P_s}$.

paper, P_0 is set large relative to $1/\lambda$ and $1/c$. This segmentation algorithm requires the specification of two parameters in addition to that of P_0:

- ϵ: Since estimating f requires that $1/(1-s)^2$ be well-behaved, we must also enforce a constraint on the range of s. A simple solution is to clip each estimate of the edge-strength function so that for some small ϵ, $s \in [0, 1-\epsilon]$. This solution proves adequate.
- Δ: As the segmentation algorithm is iterative in nature, one must specify a stopping criterion. For all of the results in this section, the algorithm stops when the percentage change in the functional (9) falls below Δ.

Our experience has been that a single pair of values of ϵ and Δ can be used to produce good segmentations of a variety of signal types. The parameter Δ specifies when to stop, but one also needs an initial guess with which to start the iterative algorithm. For the results in this section, the algorithm starts by estimating f using an initial estimate of the edge-strength function $s^0 = 0$. A list of all of the algorithm's parameters are listed in Table I.

To illustrate the operation of the algorithm, some examples follow. These, in turn, are followed by some Monte Carlo experiments designed to assess quantitatively the performance of the algorithm.

A. Examples

Fig. 2 illustrates a segmentation of a synthetic signal. The data g in Fig. 2(a) consists of a signal f to which unit intensity white Gaussian measurement noise has been added. The signal f is a realization of a Gaussian process described by (11). The process is started with initial condition $f(0) = 0$ and generated using for s the double-sided exponential function

$$0.97 \times \exp(-|i - 512|/10). \tag{17}$$

The values of the parameters used are listed in Table I. Now, recall that where the edge-strength function is approximately one,

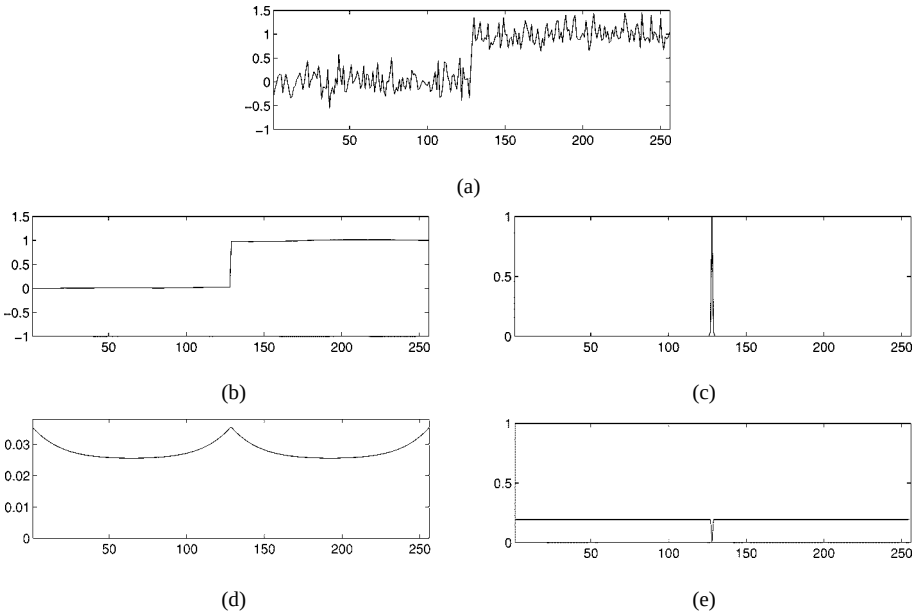


Fig. 3. Segmentation of a noisy unit step as plotted in part (a). The measurement noise is white and Gaussian with standard deviation 0.2. All parameters are given in Table I. (a) Data g, (b) reconstruction $\hat{f}$, (c) estimate of the edge-strength function $\hat{s}$, (d) reconstruction error standard deviations $\sqrt{P_f}$, and (e) estimate of edge-strength function error standard deviations $\sqrt{P_s}$

the variance of the increment in the model of f increases. This is clearly evident in Fig. 2(a) in which the particular realization of f displays a clear jump in its value in the vicinity of the edge-strength function's peak.

The data in Fig. 2(a) are then fed into the iterative segmentation and denoising algorithm. The results displayed in the remaining parts of Fig. 2 are after five iterations of the algorithm, at which point, the value of the functional (9) was changing by less than $\Delta = 0.01\%$. No clipping was necessary during the course of the run, and thus, the results are true to the discrete form of the variational formulation (9). The final reconstruction is a smoother version of the data, but the edge has not been smoothed away. The final estimate of the edge-strength function has a strong peak at the location of the edge. These are the desired results of the segmentation algorithm. In addition, the estimation error variance for f in Fig. 2(d) displays the characteristic one would expect: away from the expected edge, considerable lowpass filtering is effected, reducing the noise variance. However, in the vicinity of the edge, one expects greater variability and, in essence, the estimator performs less noise filtering, resulting in a larger error variance. Note also that the variance in the estimate of the edge process is almost constant, with a slight drop in the vicinity of the edge, i.e., where f changes abruptly, reflecting greater confidence that an edge is present in this vicinity. The results for the preceding example are good, but not completely convincing by themselves since f is matched to the algorithm by its construction. Consider a prototypical signal not matched to the model, namely a step edge. Fig. 3 displays results for data consisting of a unit step embedded in white Gaussian noise. The estimates are shown after 12 iterations. Once again, no clipping was necessary in the iterative process. The results demonstrate that the algorithm works as desired. It removes almost all of the noise away from the edge, while accurately preserving the discontinuity. Note that P_f increases near the outer ends because there are fewer measurements in the vicinity and λ is large. As in the case of the first example, however, the error statistics reflect the fact that, near the edge, one expects less noise reduction in estimating f and has higher confidence in the estimate of the edge process because of the abrupt change in value of f.

B. Monte Carlo Experiments

Two different sets of Monte Carlo experiments are presented here. The first set provides some information for interpreting the error statistics. The second set shows that the algorithm is robust to parameter settings.

1) Error Statistics: In this section, a more careful look is taken at the error statistics provided by the segmentation algorithm in order to assess their accuracy and utility. Since the full iterative algorithm is nonlinear, the exact error variances in estimating f and s are not easily computed. The statistics calculated by our algorithm represent approximations that result from the linear estimation problems for each of the two separate coordinate descent steps for f and s. Fig. 4 presents Monte Carlo results comparing the error statistics computed by the segmentation algorithm with the actual error variances. Each experiment in this simulation corresponds to (a) generating a realization f of the process described by (11) for the fixed edge-strength function s given by (17) and with the initial value $f(0)$ set to 0; (b) adding white Gaussian measurement noise with unit intensity; and (c) applying the segmentation algorithm using the parameters in Table I to obtain the estimates $\hat{f}$ of the realization f and $\hat{s}$ of the edge-strength function s as well as P_f and P_s, the error variances for these estimates that the algorithm generates.

The quantities of interest for each run are $e_f = (\hat{f} - f)$, $e_s = (\hat{s} - s)$, and the error statistics P_f and P_s computed by the algorithm. From 100 independent runs, the following quantities are estimated: E e_f^2, Var(e_f), E P_f, E e_s^2, Var(e_s), and E P_s. These are plotted in Fig. 4 along with Monte Carlo error bars

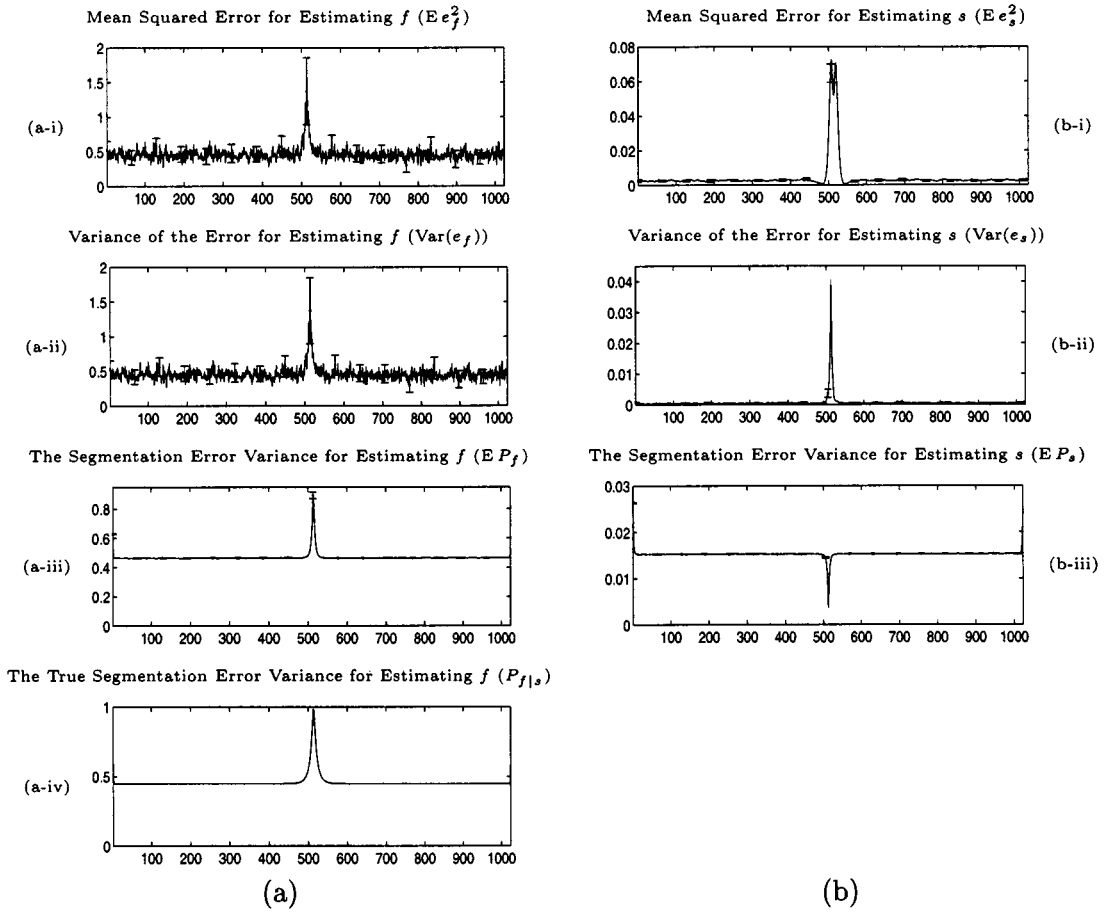


Fig. 4. Comparison of various error statistics compiled using Monte Carlo techniques for segmenting synthetic data. The data are realizations of a process whose statistics are given by (11) for the exponential edge-strength function given by (17). Part (a) of this figure displays statistics concerning the reconstruction errors, $e_f = (\hat{f} - f)$, and the reconstruction error standard deviations generated by the algorithm, P_f. Part (a-iv) displays the optimal error standard deviations for estimating f given that the true edge-strength function given by (17) is known. Part (b) of this figure displays the statistics concerning the errors in estimating the edge-strength function, $e_s = (\hat{s} - s)$. (a-i) Mean squared error for estimating f (Ee_f^2), (a-ii) variance of the error for estimating f $(\mathrm{Var}(e_f))$, (a-iii) segmentation error variance for estimating f $(E\, P_f)$, (a-iv) true segmentation error variance for estimating f (P_{fs}), (b-i) mean squared error for estimating s (Ee_s^2), (b-ii) variance of the error for estimating s $(\mathrm{Var}(e_s))$, and (b-iii) segmentation error variance for estimating s $(E\, P_s)$.

set at 2 standard deviations. Comparing Fig. 4(a)-i for E e_f^2 and Fig. 4(a)-ii for Var (e_f), one sees that these are quite close in value, indicating that the estimate produced by our algorithm is essentially unbiased. Comparing these two figures with the plot of E P_f, one observes that the error variance computed by our algorithm has essentially the same shape, reflecting the fact that it accurately captures the nature of the errors in estimating f. Fig. 4(a)-iv shows a plot of the error variance for an estimator that is given perfect knowledge of the edge process. Comparing this to Fig. 4(a)-iii, one notices that the segmentation algorithm performs nearly as well as if s were known perfectly and did not have to be estimated.

The error statistics for the edge-strength function are depicted in Fig. 4(b). E e_s^2 and Var(e_s) being small relative to one indicate that the estimate of the edge-strength function is quite accurate and that the error does not vary considerably from sample path to sample path. In addition, the shapes of these plots have several interesting features related to the behavior of the estimator in the vicinity of the edge. Note first that, as can be seen in Fig. 2, the algorithm tends to estimate edge-strength functions that are slightly narrower than the actual edge-strength function. This is actually preferable for segmentation, for which the peak locations in the estimates of the edge strength-functions are more important than the estimates' shapes. Because of this bias toward tighter edge localization, $\hat{s}$ is a slightly biased estimate of s given by (17), as evidenced by the broader peak of E e_s^2 as compared to Var(e_s).

A second interesting point is that E e_s^2 increases slightly in the vicinity of the edge, while the variance computed by the estimation algorithm, E P_s, decreases. The reason for this can be explained as follows. Specifically, the estimator believes that it is has more information about s when the gradient of f is large. Thus, in the vicinity of an edge, the estimator indicates a reduction in error variance for estimating s. However, if the estimate of the *location* of the edge is in error, then the difference $e_s = (\hat{s} - s)$ will exhibit large, very localized errors, both positive and negative (just as one would see in the difference of two discrete-time impulses whose locations are slightly different). Thus, rather than providing an accurate estimate of the size of the estimation error variance in this vicinity, this dip in the error variance should be viewed as a measure of confidence in the presence of an edge in the vicinity.

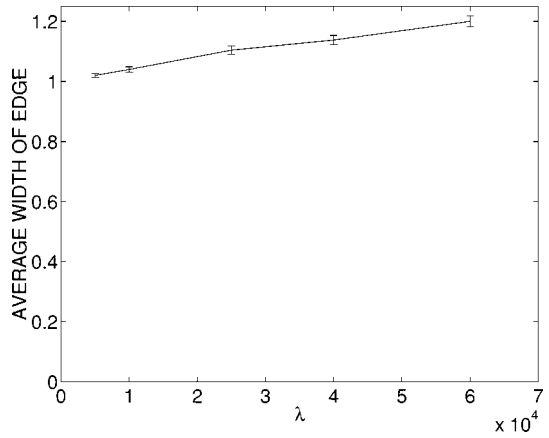


Fig. 5. Average value of W for the step edge example of Fig. 3 but for different values of λ.

2) Parameter Selection: Fig. 5 presents results from a Monte Carlo simulation designed to characterize how well the algorithm can segment signals for different values of the parameter λ. In this experiment, the algorithm is segmenting a unit step edge. The quantity computed from each segmentation is W, the number of values of the estimated edge-strength function that lie above a given threshold set close to one. This corresponds roughly to the sum of the widths of the edges in the segmentation. In the case of the step edge, the desired value of W is one. Except for λ, the algorithm's parameters take on the values listed in Table I for Fig. 3. The threshold is set to 0.9. The results in Fig. 5 are for 500 runs, and the error bars are set at one standard deviation.

Recall that the amount of smoothness the algorithm expects in f where there is no edge is directly related to λ. For $\lambda = 2.5 \times 10^3$, the algorithm generates the very flat step estimate of Fig. 3. However, one can not set λ too high because, as Fig. 5 shows, the average value of W increases with λ. Remember that W is a measure of how many points are very likely to be edges. If λ is set too high, the algorithm will put edges in many places and set the estimate of the function f almost constant between edges. Although the results of the algorithm depend on λ, the slope of the curve in Fig. 5 is not very steep. This indicates that small perturbations in the value of λ will not severely diminish the performance of the algorithm.

Similar experiments have been performed in which other parameters were adjusted. These can be found in [19]. To summarize, the effect of these parameters on the results coincide with the simple guidelines presented in Section IV. Furthermore, these Monte Carlo experiments indicate that the results are robust to small changes in the parameter settings.

VI. A Multiscale Method for Image Segmentation

As discussed in [19], solving the exact 2-D counterparts to the estimation problems in the 1-D segmentation and denoising algorithm is not an easy task. In particular, the calculations in 2-D correspond to solving estimation problems with prior models that are 2-D Markov random fields (MRF) or, equivalently, to solving elliptic partial differential equations and computing the diagonal elements of the inverses of elliptic operators [9]. There exist no known algorithms which can compute the necessary quantities for this general problem with fewer than $O(n^3)$ operations, where n is the linear dimension of the image.

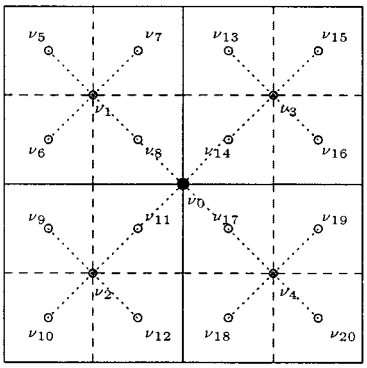


Fig. 6. How discontinuities are incorporated into the $1/f$-like multiscale model.

Since the objective is to generate estimates and error variances with constant computational complexity per pixel (i.e., with $O(n^2)$ operations), one is confronted with the need to develop approximations. The approach taken here is to alternately estimate a reconstruction and an edge-strength function assuming multiscale prior models. When the prior is a multiscale model, there is an efficient algorithm for producing the estimates, and the associated error variances of the reconstruction and edge-strength function [14], [17] (see also Section III). These two estimation steps replace the two coordinate descent step in Shah and Pien's variational approach to segmentation as discussed in Section II. The estimation is performed in an overlapped domain, as described in [20]. Thus, the multiscale models do not directly model f and s but the corresponding lifted versions $\mathbf{f}$ and $\mathbf{s}$. The two models, one for each of the lifted fields $\mathbf{f}$ and $\mathbf{s}$, are described in Section VI-A. Note that there are two principal distinctions between our multiscale modeling approach and some other quad-tree approaches to solving image processing problems. The first is that the variables on the tree are states that decorrelate regions of the image, and the second is that the associated estimation algorithm passes information everywhere along the tree so as to produce globally optimal estimates of the lifted fields.

A. Multiscale Models for Segmentation

Consider first, the model used for the estimation of $\mathbf{s}$. As discussed in [9], [10], [12], [13], [16], the smoothness penalty associated with the gradient, used for example in (2) and (3), corresponds to a fractal penalty in that it is roughly equivalent to a $1/f$-like prior spectrum for the random field being modeled. This type of spectrum has a natural scaling law; namely, the variances of increments at finer and finer scales decrease geometrically. In [9], [10], [12], [13], it was demonstrated that a very simple multiscale model having this same scaling property leads to estimates that are very similar to those produced using the original smoothness penalty. A model of this type is used for the lifted version $\mathbf{s}$ of the edge-strength function. Specifically

$$\mathbf{s}(\nu) = \mathbf{s}(\nu\bar{\gamma}) + d(\nu)B_s w_s(\nu) \tag{18}$$

TABLE II
DESCRIPTION OF PARAMETERS, IN THE MULTISCALE METHOD

Parameter	*Description*	*Value for the Results in Figs.* 7	8	9
b	b affects the edginess of the estimate of the edge-strength function.	30		10
λ	λ affects the determination of what is an edge and what isn't.	15		50
B_s	B_s adjusts the multiscale smoothness penalty placed on s. See (18).	1		0.87
B_f	B_f adjusts the multiscale smoothness penalty placed on f. See (19).	1/40		0.75
r	r is the assumed noise variance in image data.	1	6	1
P_{root}	P_{root} is the multiscale model prior covariance for the process value at the root node.	1×10^6		
$\mathcal{O}$	Each component of $\mathcal{O}$ specifies the amount of overlap at a particular scale.	(16 10 / 7 4 2 / 2 0 0)		(50 31 / 18 11 7 / 4 2 1 / 0 0)
ϵ	Estimates of s are clipped to lie within $[0, 1-\epsilon]$.	0.01		
I	I is the number of iterations of estimating f and s.	2		

where B_s is a constant, and the $w_s(\nu)$ are independent unit variance Gaussian random variables. As described in [12], [15], the $d(\nu)$ are constants that decrease from one scale to the next finer scale and depend on the amount of overlap used. The measurements and measurement error variances used in conjunction with the model for $\mathbf{s}$ in (18) are exactly analogous to those specified by γ and A^{-2} in 1-D. The only difference in 2-D is that a sum of the squares of the first differences in each direction replaces the square of the first difference in one direction. Details are specified in [19].

The multiscale model for the lifted version of the reconstruction $\mathbf{f}$ is, to some extent, similar to the one for the edge-strength function. However, significant modification to this model is needed in order to capture the presence of discontinuities, as indicated by the edge estimates. In particular, in the 1-D case, as captured in (11), the increments of f have a variance which is inversely proportional to the corresponding value of $(1-s)^2$. Thus, the variance of the increment of f is large near an edge (i.e., where s is approximately one in value). In a similar manner, one needs to capture the idea that increments of $\mathbf{f}$, as one moves to finer scales, should have variances that reflect the presence of edges (i.e., that are again inversely proportional to $(1-s)^2$). This is done as follows. Note that each node on the tree can be thought of as representing the center of a subregion of the image domain. A 2-D example is depicted in Fig. 6. The dots in this figure correspond to the center points of the regions associated with different nodes on the tree. The dots are shaded according to the scale of the corresponding node on the tree; the darker the dot, the coarser the scale. Thus, for example, the node ν_0 represents the entire large square region, while the node ν_3 at the next finest scale represents the upper-right quadrant of this large square. Now, if there is an edge located between ν_0 and ν_3 (i.e., if the values of s at image domain pixels between these nodes indicate the presence of an edge), the variance of the scale-to-scale increment of $\mathbf{f}$ between these two nodes should increase. More precisely, the model for $\mathbf{f}$ is specified by the recursion

$$\mathbf{f}(\nu) = \mathbf{f}(\nu\overline{\gamma}) + \eta(\nu)d(\nu)B_f w_f(\nu) \tag{19}$$

where B_f is a constant, $w_f(\nu)$ are independent unit variance Gaussian random variables, and $\eta(\nu)$ is the sum of $1/(1-\hat{s}(i,j))^2$ for estimates of the edge-strength function values which fall on the line connecting ν and $\nu\overline{\gamma}$. In this manner, additional uncertainty is put into the recursion for $\mathbf{f}$ at the appropriate locations.

B. Numerical Results

This section presents numerical results on two test images, a synthetic image of a circle and an MRI brain scan. Table II lists the algorithm's parameters and values[2]. The execution times for each example were on the order of minutes when run using MATLAB on a Sparc Ultra.

The segmentation of a synthetic 64 × 64 image of a circle is presented in Fig. 7. The circle image provides a simple example for which one can observe the desired outputs of the algorithm. The reconstruction contains very little of the noise present in the original data. Furthermore, the edges of the circle have not been smoothed over in the reconstruction. The estimate of the edge-strength function is close to zero everywhere except at the edge of the circle, where the estimate tends to unity. This indicates the algorithm has successfully identified edges and has ignored the spurious changes in intensity in the data due to noise.

[2]The parameters b, λ, B_s, B_f, and r have interpretations analogous to those presented in Section IV for the parameters b, λ, c, and r appearing in the 1-D problem. The interpretations were used to guide the choice of parameters used in the numerical examples of the multiscale method for segmentation.

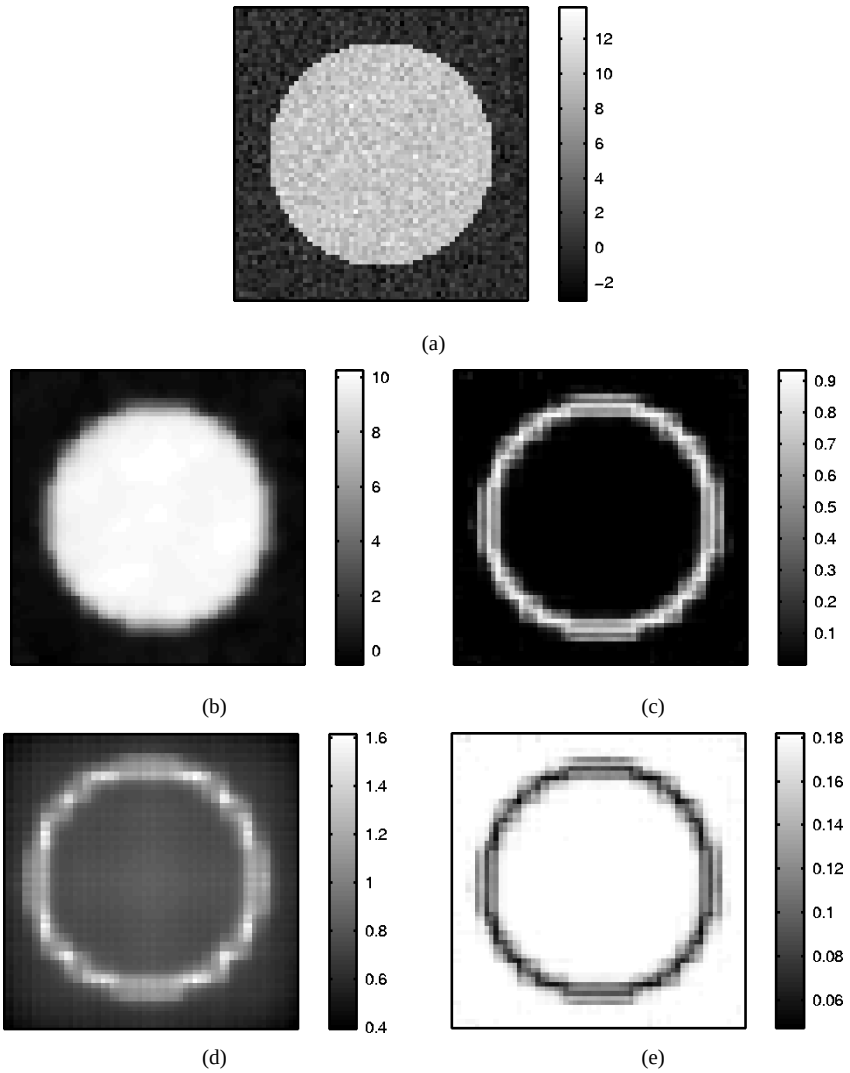


Fig. 7. Segmentation of a synthetic circle image computed using the multiscale method. (a) Data g, (b) reconstruction $\hat{f}$, (c) estimate of the edge-strength function $\hat{s}$; (d) reconstruction error standard deviations $\sqrt{P_f}$; and (e) estimate of edge-strength function error standard deviations $\sqrt{P_s}$.

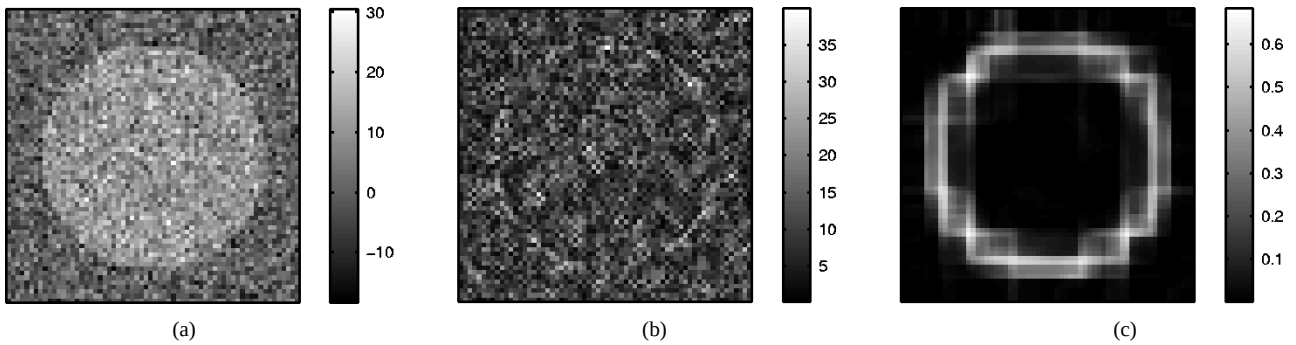


Fig. 8. This figure illustrates the difference between a gradient map and the estimate of the edge-strength function for a noisy circle image. (a) Data g, (b) magnitude of the gradient, and (c) estimate of the edge-strength function $\hat{s}$.

Fig. 8 illustrates the difference between a gradient map and the estimate of the edge-strength function. As for the example in Fig. 7, the unprocessed image is a circle embedded in white noise. The standard deviation of the noise is six. The magnitude of the gradient is very noisy, but the edge-strength function is only moderately noisy and clearly indicates the location of the underlying circle.

In addition to the reconstruction and estimate of the edge-strength function, the multiscale algorithm computes the standard deviations of the error in the reconstruction and estimate of the edge-strength function. Notice that the error standard deviations for the reconstruction increase near edges and, for the estimate of the edge-strength function, decrease near edges, as in the 1-D case. Thus, one expects that the error standard

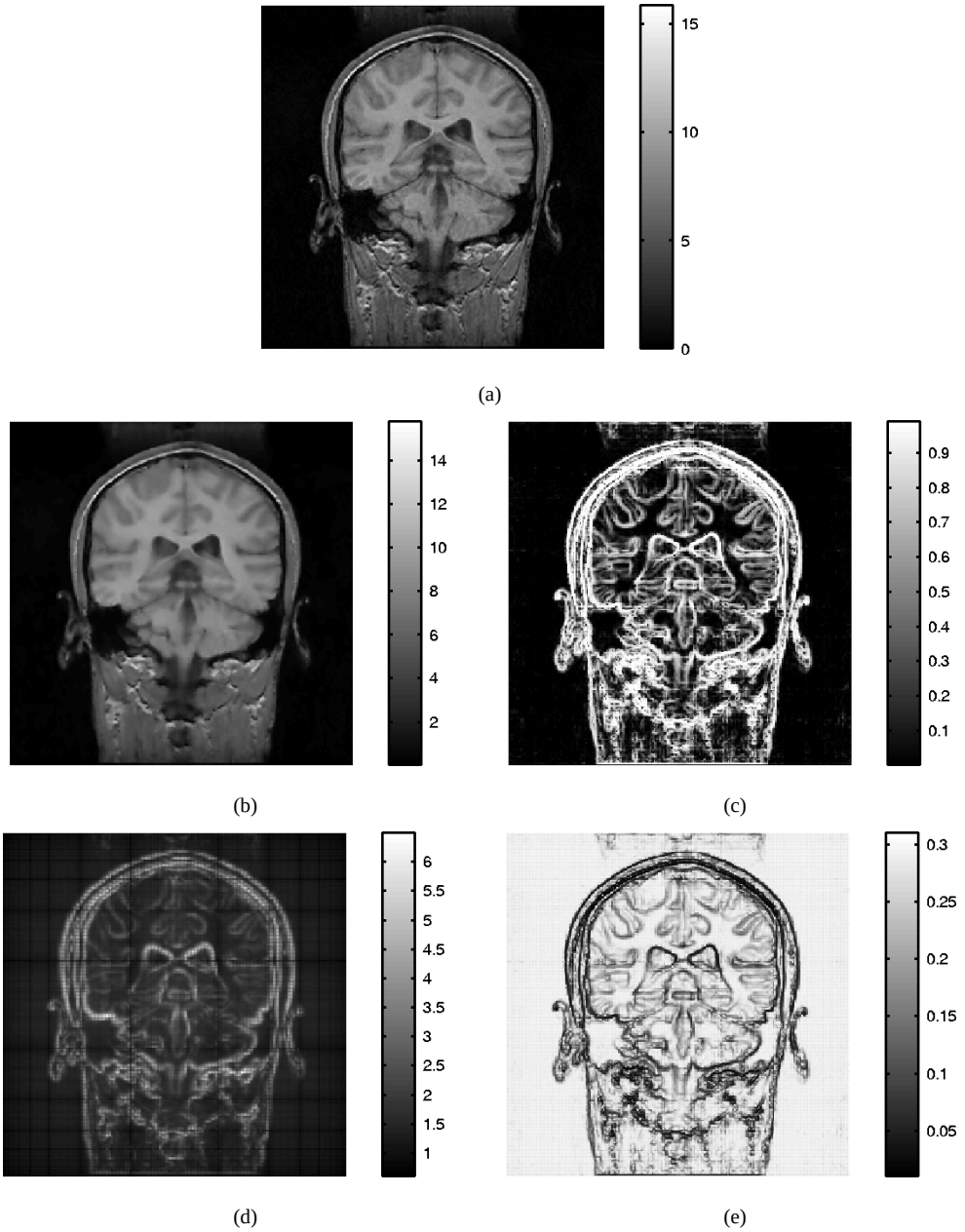


Fig. 9. MRI segmentation computed using the multiscale method. (a) Data g, (b) reconstruction $\hat{f}$, (c) estimate of the edge-strength function $\hat{s}$; (d) reconstruction error standard deviations $\sqrt{P_f}$; and (e) estimate of edge-strength function error standard deviations $\sqrt{P_s}$.

deviations in Figs. 7 and 9 are of similar significance to those generated in 1-D. Another consequence of the above-mentioned properties of the error standard deviations is that they can be used not only to estimate one's confidence in segmenting the image but also to improve one's estimate of the boundary locations. This is a consequence of the error standard deviations marking the edges in the image as well or better than the estimate of the edge-strength function.

Fig. 9 displays a multiscale segmentation of a 256 × 256 MRI brain scan[3]. There are many different potential uses of MRI brain segmentation. Associated with each of these different uses are different goals of the segmentation. One of these goals is to demarcate the boundaries of the ventricles, the two hollow regions in the middle of the brain. Another goal is to determine the boundary between gray and white matter in the brain. The estimate of the edge-strength function displayed in Fig. 9 does a good job at indicating likely boundaries both of the ventricles and of the gray and white matter.

[3]An MRI image with pixel values ranging over more than 800 integers was shifted and scaled to produce the data for this figure. The dynamic range of the scaled MRI data is close to that of the circle images.

VII. Conclusion

In this paper, we have described a new approach to the reconstruction and segmentation of noise corrupted signals and images. The points of departure for our work are the variational formulation of Mumford and Shah [2] and, more explicitly, the relaxed variational formulation of Ambrosio and Tortorelli [5].

The latter formulation leads to the simultaneous computation of both a reconstructed signal or image and an edge-strength function which provides a measure of the likely locations of edges or abrupt changes in the image or signal values.

The contributions of this paper are several. First, we provide precise statistical interpretations of the steps involved in estimating both the reconstructed signal or image and the edge-strength function. This interpretation not only establishes an intellectual bridge to statistical estimation, but it also leads directly to the extension of the algorithm to produce error variances for the reconstructed image and edge-strength function.

Moreover, this statistical interpretation leads us to another significant extension and new algorithm, in the case of 2-D image processing. In particular, replacing the random field image models implied by the variational formulations in [1], [7] with a multiscale prior model with similar characteristics yields an extremely efficient algorithm for the simultaneous estimation of the reconstructed image and edge-strength function. Furthermore, the new algorithm can efficiently compute the variances of the errors in these estimates. The algorithm for computing both estimates and error variances given a multiscale model has constant computational complexity per pixel. In contrast, the best known algorithm for computing estimates and error variance given a random field model has a complexity per pixel that increases as the square root of the total number of pixels.

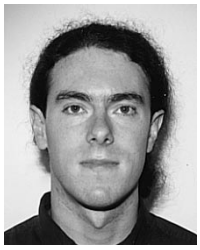
In addition to these substantial computational advantages, we have also demonstrated the efficacy of these algorithms through several experimental studies. In 1-D, we demonstrated the accuracy of the error variance calculations that our algorithm automatically produces for the reconstructed image. In 2-D, we also demonstrated the ability of our algorithm to produce reliable estimates even in the presence of very high noise. In particular, we demonstrated that the estimates of edge-strength functions produced by our approach are much more robust to noise than image gradient maps. Thus, the edge-strength function estimates can be subsequently processed to produce robust and accurate edge estimates, e.g., by thresholding [1] or more sophisticated methods such as curve evolution [8].

ACKNOWLEDGMENT

The authors would like to thank several of the reviewers for their constructive criticisms and suggestions.

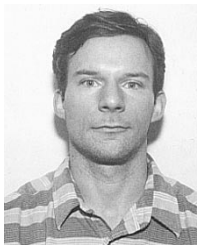
REFERENCES

[1] J. Shah, "Segmentation by nonlinear diffusion, II," in *Proc. IEEE Computer Vision Pattern Recognition Conf.*, 1992, pp. 644–664.
[2] D. Mumford and J. Shah, "Boundary detection by minimizing functionals, I," in *Proc. IEEE Conf. Computer Vision Pattern Recognition*, 1985, pp. 22–26.
[3] ——, "Optimal approximations by piecewise smooth functions and associated variational problems," *Commun. Pure Appl. Math.*, vol. 42, pp. 577–684, 1989.
[4] J. Morel and S. Solimini, *Variational Methods in Image Segmentation*. Boston, MA: Birkhäuser, 1995.
[5] L. Ambrosio and V. M. Tortorelli, "Approximation of functionals depending on jumps by elliptic functionals via Γ-convergence," *Commun. Pure Appl. Math.*.
[6] ——, "On the approximation of free discontinuity problems," *Bolletino Della Unione Matematica Italiana*, vol. 6-B, pp. 105–123, 1992.
[7] H. Pien and J. Gauch, "Variational segmentation of multi-channel MRI images," in *Proc. IEEE Int. Conf. Image Processing*, Nov. 1994, pp. 508–512.
[8] H. H. Pien, M. Desai, and J. Shah, "Segmentation of MR images using curve evolution and prior information," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 116, no. 8, pp. 1233–1245, 1997.
[9] M. Luettgen, W. C. Karl, and A. S. Willsky, "Efficient multiscale regularization with applications to the computation of optical flow," *IEEE Trans. Image Processing*, vol. 3, pp. 41–64, Jan. 1994.
[10] P. Fieguth, W. Karl, A. Willsky, and C. Wunsch, "Multiresolution optimal interpolation and statistical analysis of TOPEX/POSEIDON satellite altimetry," *IEEE Trans. Geosci. Remote Sensing*, vol. 33, pp. 280–292, Feb. 1995.
[11] J. Kaufhold, M. Schneider, A. W. Willsky, and W. C. Karl, "A statistical method for efficient segmentation of MR imagery," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 11, no. 8, pp. 1213–1231, 1997.
[12] P. Fieguth, "Application of multiscale estimation to large scale multidimensional imaging and remote sensing problems," Ph.D. dissertation, Mass. Inst. Technol., Cambridge, June 1995.
[13] M. Luettgen, "Image processing with multiscale stochastic models," Ph.D. dissertation, Mass. Inst. Technol., Cambridge, May 1993.
[14] K. Chou and A. Beneveniste, "Multiscale recursive estimation, data fusion, and regularization," *IEEE Trans. Automat. Contr.*, vol. 39, pp. 464–478, Mar. 1994.
[15] P. W. Fieguth, A. S. Willsky, and W. C. Karl, "Efficient multiresolution counterparts to variational methods for surface reconstruction," *Comput. Vis. Image Understand.*, vol. 70, pp. 157–176, May 1998.
[16] R. Szeliski, *Bayesian Modeling of Uncertainty in Low-Level Vision*. Norwell, MA: Kluwer, 1989.
[17] K. Chou, "A stochastic modeling approach to multiscale signal processing," Ph.D. dissertation, Mass. Inst. Technol., Cambridge, May 1991.
[18] M. R. Luettgen, W. C. Karl, A. S. Willsky, and R. R. Tenney, "Multiscale representations of Markov random fields," *IEEE Trans. Signal Processing*, vol. 41, pp. 3377–3396, Dec 1993.
[19] M. K. Schneider, "Multiscale methods for the segmentation of images," M.S. thesis, Mass. Inst. Technol., Cambridge, May 1996.
[20] W. W. Irving, P. W. Fieguth, and A. S. Willsky, "An overlapping tree approach to multiscale stochastic modeling and estimation," *IEEE Trans. Image Processing*, vol. 6, pp. 1517–1529, Nov. 1997.



Michael K. Schneider (S'93) received the B.S.E. degree in electrical engineering from Princeton University, Princeton, NJ, in 1994, and the M.S. degree in electrical engineering from the Massachusetts Institute of Technology (MIT), Cambridge, in 1996. He is currently pursuing the Ph.D. degree at MIT.

His research interests include statistical signal and image processing and numerical linear algebra.



Paul W. Fieguth (S'87–M'96) received the B.A.Sc. degree from the University of Waterloo, Waterloo, Ont., Canada, in 1991, and the Ph.D. degree from the Massachusetts Institute of Technology (MIT), Cambridge, in 1995, both in electrical engineering.

He joined the University of Waterloo in 1996, where he is currently Assistant Professor in systems design engineering. Prior to this, he held postdoctoral positions with the Department of Computer Science, University of Toronto, Toronto, Ont., and in the Laboratory for Information and Decision Systems, MIT. His research interests include statistical signal and image processing, hierarchical algorithms, and interdisciplinary applications of such methods, particularly to remote sensing.

William C. Karl (M'91) received the S.M., E.E., S.B., and Ph.D. degrees in electrical engineering and computer science from the Massachusetts Institute of Technology, Cambridge.

He was Staff Research Scientist with Brown–Harvard-MIT Center for Intelligent Control Systems and the MIT Laboratory for Information and Decision Systems, Cambridge, from 1992 to 1994. He joined the faculty of Boston University, Boston, MA, in 1995, where he is currently Assistant Professor of electrical, computer, and systems engineering. Since January 1996, he has held a joint appointment in the Department of Biomedical Engineering. His research interests are in the areas of multidimensional and multiscale signal and image processing, geometric estimation, detection, and medical signal and image processing.

Dr. Karl is associate editor of the TRANSACTIONS ON IMAGE PROCESSING. He is a member of Sigma Xi.

Alan S. Willsky (S'70–M'73–SM'82–F'86) received the S.B. and Ph.D. degrees from the Massachusetts Institute of Technology (MIT), Cambridge, 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 Laboratory for Information and Decision Systems, MIT. He is also a founder and member of the board of directors of Alphatech, Inc., and is currently a member of the U.S. Air Force Scientific Advisory Board. He has held visiting positions at Imperial College, London, U.K., L'Université de Paris-Sud, and the Institut de Recherche en Informatique et Systémes Aléatoires, Rennes, France. He is the author of the research monograph *Digital Signal Processing and Control and Estimation* and is the co-author of the undergraduate text *Signals and Systems*. His research interests are in development and application of advanced methods of estimation and statistical signal and image processing. Methods he has developed have been successfully applied in a wide variety of applications including failure detection in high-performance aircraft, advanced surveillance and tracking systems, electrocardiogram analysis, computerized tomography, and remote sensing.

Dr. Willsky received the Donald P. Eckman Award from the American Automatic Control Council in 1975. He was Program Chairman for the 17th IEEE Conference on Decision and Control, has been an associate editor of several journals and special guest editor for several special issues, and has served as a member of the Board of Governors and Vice President for Technical Affairs of the IEEE Control Systems Society. In 1988, has was made a Distinguished Member of the IEEE Control Systems Society. He has given plenary and keynote lectures at a number of major scientific meetings including the 20th IEEE Conference on Decision and Control, the 1991 IEEE International Conference on Systems Engineering, the 1991 SIAM Conference on Applied Linear Algebra, the 1992 Inaugural Workshop for the National Centre for Robust and Adaptive Systems, Canberra, Australia, the 1992 INRIA 25th Anniversary Symposium in Paris, the 1993 IEEE Symposium on Image and Multidimensional Signal Processing, Cannes, France, and the 1997 Wavelet Applications in Signal and Image Processing Conference. He was awarded the 1979 Alfred Noble Prize by the ASCE and the 1980 Browder J. Thompson Memorial Prize Award by the IEEE for a paper excerpted from his monograph.