

Nonparametric Fixed-Interval Smoothing of Nonlinear Vector-Valued Measurements

Jeffrey A. Fessler

Abstract—This paper addresses the problem of estimating a smooth vector-valued function given noisy nonlinear vector-valued measurements of that function. We present a nonparametric optimality criterion for this estimation problem, and develop a computationally efficient iterative algorithm for its solution. The new criterion is the natural generalization of our earlier work on vector splines with linear measurement models. The new algorithm provides an alternative to the extended Kalman filter, as it does not require a parametric state-space model. We also present an automatic procedure that uses the measurements to determine how much to smooth. The algorithm demonstrates subpixel estimation accuracy on two examples: the estimation of a curved edge in a noisy image, and a biomedical image-processing application.

I. INTRODUCTION

THIS paper considers the problem of estimating a smooth vector-valued function from noisy measurements observed through a nonlinear mapping. We assume the following nonlinear measurement model:

$$y_n = h_n(x_n) + \varepsilon_n, \quad n = 1, \dots, N \quad (1)$$

where

$$\varepsilon_n, y_n \in \mathbb{R}^L, \quad x_n \in \mathbb{R}^M, \quad \text{and } h_n: \mathbb{R}^M \rightarrow \mathbb{R}^L.$$

We assume the additive measurement errors are independent between samples and are normally distributed with mean zero. Without loss of generality, we assume the covariance matrix of ε_n is $\sigma^2 \mathbf{I}$, where σ^2 may be unknown.¹ The states $\{x_n\}$ are (possibly unequally spaced) samples of a smooth vector-valued function g :

$$x_n = [g_1(t_n), \dots, g_M(t_n)]' \triangleq g(t_n), \quad t_n < t_{n+1} \quad \forall n \quad (2)$$

where “'” denotes matrix transposition. The goal is to estimate g from the measurements $\{y_n\}_{n=1}^N$.

The prevalent approach to this estimation problem is the extended Kalman filter (EKF) [1]. The EKF hinges on an assumption that the states adhere to a parametric Gauss-Markov state-space model. However, in applications such as the edge-estimation example given in Section VI, the parameters required by the EKF formulation (state evolution matrices and

covariances) are unknown and are difficult to determine. Furthermore, the state-space formulae imply the *a priori* variance of the function varies with t . Although it is natural for tracking applications, where one is given a starting state that evolves with increasing uncertainty over time, this variation is counterintuitive for off-line applications such as image processing, where t often represents *space* rather than *time*. For example, when detecting and estimating an edge in an image, the *a priori* variance of the position of the edge (the uncertainty before actually seeing the image) is the same throughout the image. Despite these objectives to parametric methods, we must use our *a priori* knowledge of the smoothness of the underlying functions if we are to obtain accurate estimates. This necessity has motivated nonparametric approaches to smoothing [2], [3], and is the basis for the new algorithm presented in this paper.

In [4], we presented a computationally efficient algorithm for nonparametric smoothing for the special case when h_n is linear, and we presented the rationale behind “penalized likelihood” estimation. Here, just as in the linear case, we must compromise between the agreement with the data and the smoothness of the estimated functions. Thus we propose the following optimality criterion:

$$\hat{g} \triangleq \arg \min_g \Phi_*(g)$$

$$\Phi_*(g) \triangleq \sum_{n=1}^N \|y_n - h_n(g(t_n))\|^2 + \sum_{m=1}^M \alpha_m \int (g_m^{(k)}(t))^2 dt. \quad (3)$$

This criterion is the natural generalization of [4, eq. (11)]. O’Sullivan [5] considered this criterion for the case of scalar measurements. For simplicity, we assume $k = 2$. The parameter $\alpha = (\alpha_1, \dots, \alpha_M)$ controls the influence of the penalty term, and in Section IV we describe how to automatically estimate α from the measurements. Until then, we assume α is known.

By the “minimal property of splines” proven in [6, theorem 2], any function \hat{g} that achieves the minimum of Φ_* is a vector spline with component functions that are cubic splines (for $k = 2$). (We restrict our attention here to natural cubic splines by imposing the end conditions that $g_m(t)$ is linear for $t < t_1$ and $t > t_N$.) However, unlike the linear case, in general, there may be multiple minima.² Physical constraints will usually rule out the irrelevant solutions. The EKF suffers the same ambiguity, a fact usually ignored since the filter update equations are initialized at some (presumably meaningful) starting state. The iterative algorithms we present below also require an initial estimate.

²Consider $h_n(x) = x^2$, then $\Phi_*(g(t)) = \Phi_*(-g(t))$.

Manuscript received September 25, 1989; revised May 10, 1990. This work was supported in part by the National Institute of Health under Contract NO1-HV-38045 and Grants R01-HL-39045, HL-39297, HL-34962, and HL-39478, by the National Science Foundation under Contract ECS-8213959, and by GE Medical Systems Group under Contract 22-84.

The author was with the Information Systems Laboratory, Department of Electrical Engineering, Stanford University, Stanford, CA 94305. He is now with the Division of Nuclear Medicine, University of Michigan Medical Center, Ann Arbor, MI 48109.

IEEE Log Number 9042311.

¹If the measurement error has the (positive definite) covariance matrix $\sigma^2 \Sigma_n$, then we can premultiply y_n and h_n by $\Sigma_n^{-1/2}$. Singular covariances may be the result of linearly dependent measurements, indicating that other constraints should be incorporated.

Since the component functions of \hat{g} are natural cubic splines, we need only estimate the coefficients of their piecewise-polynomial expansions (or, better numerical stability, their B-spline expansion [7]). In fact, if we compute $\hat{g}(t)$ at t_1, \dots, t_N , then we can compute all the coefficients from [4, eq. (6)]. From (2), this is equivalent to estimating the states $\{x_n\}_{n=1}^N$. From Appendix A of [4]

$$\sum_{m=1}^M \alpha_m \int (g_m^{(2)}(t))^2 dt = x' S_\alpha x$$

where S_α is defined in [4, eq. (23)], and

$$x \triangleq [x'_1, \dots, x'_N]'$$

Therefore, the variational problem (3) is equivalent to the following penalized nonlinear least squares problem:

$$\hat{x}_\alpha = \arg \min_x \Phi_\alpha(x)$$

$$\Phi_\alpha(x) = \|y - h(x)\|^2 + x' S_\alpha x$$

where

$$y \triangleq [y'_1, \dots, y'_N]', \quad h(x) \triangleq [h_1(x_1)', \dots, h_N(x_N)']'$$

S_α , which also depends on k in general, is the spline penalty matrix that prohibits excessive local variation in \hat{g} .

In Sections II and III, we develop an iterative method for computing \hat{x}_α . This method is summarized as a computationally efficient algorithm in Section V, after we discuss selection of α in Section IV. We demonstrate the algorithm on a curved-edge estimation problem and a biomedical image-processing application in Section VI, and conclude with open problems in Section VII.

II. LINEARIZATION

We first consider estimating x by a linearization method similar to the EKF approach. Assume $x_o \triangleq [x'_{o,1}, \dots, x'_{o,N}]'$ is an initial estimate³ of x . By the first-order Taylor's expansion of h_n about $x_{o,n}$

$$h_n(x_n) \approx h_n(x_{o,n}) + H_n(x_n - x_{o,n}) \quad (4)$$

where H_n is the $L_n \times M$ Jacobian of h_n evaluated at $x_{o,n}$. Substituting (4) into (1), we get

$$y_n \approx h_n(x_{o,n}) + H_n(x_n - x_{o,n}) + \varepsilon_n.$$

Multiplying both sides by $(H_n' H_n)^{-1} H_n'$ and rearranging yields

$$z_n \approx x_n + v_n \quad (5)$$

where

$$z_n \triangleq x_{o,n} + (H_n' H_n)^{-1} H_n' (y_n - h_n(x_{o,n})).$$

The transformed measurement error v_n is normally distributed with mean zero and covariance matrix $\Pi_n = (H_n' H_n)^{-1}$. This procedure requires⁴ that $(H_n' H_n)$ be invertible, or, equivalently,

that the Jacobians all have rank M . A necessary condition is therefore that $L_n \geq M \forall n$. (One special case is worth noting: if the initial estimate is the (unpenalized) maximum-likelihood estimate, i.e., $x_{o,n} = \arg \min_{x_n} \|y_n - h_n(x_n)\|^2$, then $z_n = x_{o,n}$, and each covariance matrix Π_n is a corresponding Fisher information matrix.)

We have transformed the nonlinear measurements (1) into a set of linearized measurements (5) that are now in a form suitable for the linear vector-spline smoothing algorithm of [4]. The resulting estimate, denoted \hat{x}_{Lin} satisfies

$$\hat{x}_{\text{Lin}} = \arg \min_x \Phi_o(x)$$

$$\Phi_o(x) = (z - x)' \Pi^{-1} (z - x) + x' S_\alpha x$$

where

$$z \triangleq [z'_1, \dots, z'_N]', \quad \Pi = \text{diag}(\Pi_n).$$

Φ_o is a quadratic form, and its minimizer (cf. [4, eq. (22)]) is

$$\hat{x}_{\text{Lin}} = (\Pi^{-1} + S_\alpha)^{-1} \Pi^{-1} z. \quad (6)$$

In the implementation of this algorithm, we compute \hat{x}_{Lin} with the $O(M^3 N)$ algorithm of [4], rather than by direct evaluation of (6).

A significant difference between this nonparametric approach and the EKF approach is the timing of the linearization. Here, we first independently linearize all the measurements, and then smooth. For the EKF, the measurements are linearized about the most recent estimate from the recursive update formulae [1]. Though our approach therefore requires more effort "up front" in obtaining initial estimates, it does avoid some of the potential problems of EKF mistracking [8], [9].

Since the accuracy of \hat{x}_{Lin} depends on the accuracy of (4), we would usually iterate by using \hat{x}_{Lin} as a new "initial" estimate and repeating the above procedure. Unfortunately, there is no guarantee such iterations will accomplish our original goal of minimizing Φ_* or will even converge. The most we can claim is that the optimal estimate \hat{x}_α is a fixed point of the iterations, i.e., if $x_o = \hat{x}_\alpha$ then $\hat{x}_{\text{Lin}} = \hat{x}_\alpha$. The standard solution to this dilemma is to introduce a relaxation parameter. It is not clear how to do this from the above derivation, despite its intuitive appeal. With an eye towards applying the Levenberg-Marquardt relaxation method [10], in the next section we derive the Hessian estimate of x .

III. HESSIAN APPROACH

The Hessian approach [11] is to locally approximate the functional Φ_α by a quadratic

$$\Phi_\alpha(x) \approx \Phi_\alpha(x_o) - 2d'(x - x_o) + (x - x_o)' D (x - x_o).$$

The estimate is then given as

$$\hat{x}_{\text{Hess}} = x_o + \delta$$

where δ is the solution to

$$D \delta = d. \quad (7)$$

Neglecting second derivatives [11], one can easily compute d and D :

$$d = -\frac{1}{2} \frac{\partial \Phi_\alpha}{\partial x} \Big|_{x=x_o} = H'(y - h(x_o)) - S_\alpha x_o$$

³Obtaining an initial estimate is clearly application dependent. The transform approach of Bresler and Macovski [8] is well suited for nonlinearities that separate into "shift" and "shape" parameters.

⁴It is not strictly necessary that all the Jacobians exist or have rank M . Spline smoothing can be applied to nonuniformly spaced measurements, so one could simply discard any measurements violating the existence or rank conditions.

and

$$D = \frac{1}{2} \left\{ \frac{\partial \Phi_\alpha}{\partial x_i} \frac{\partial \Phi_\alpha}{\partial x_j} \right\} = (\Pi^{-1} + S_\alpha)$$

where

$$H \triangleq \text{diag}(H_n).$$

Therefore, the Hessian estimate is:

$$\hat{x}_{\text{Hess}} = x_o + (\Pi^{-1} + S_\alpha)^{-1} [H'(y - h(x_o)) - S_\alpha x_o]. \quad (8)$$

The Levenberg-Marquardt (LM) approach [10] to relaxation of the Hessian nonlinear least squares method is equally applicable to our penalized nonlinear least squares problem, since the penalty is a quadratic. Instead of (7), the LM approach (see discussion in [11, sec. 14.4]) is to compute the update as follows:

$$(D + \lambda \Lambda) \delta_\lambda = d$$

yielding the estimate

$$\hat{x}_{\text{Hess}, \lambda} = x_o + \delta_\lambda.$$

Λ is a diagonal matrix whose elements are a measure of scale; we take Λ to be the diagonal elements of Π^{-1} . The LM algorithm provides a procedure for choosing the relaxation parameter λ to ensure that the new estimate is better than the previous estimate, i.e., $\Phi_\alpha(\hat{x}_{\text{Hess}, \lambda}) < \Phi_\alpha(x_o)$. This procedure guarantees convergence to a local minimum when one iterates the Hessian method.

By applying (5), (6), and (8)

$$\begin{aligned} \hat{x}_{\text{Lin}} &= (\Pi^{-1} + S_\alpha)^{-1} \Pi^{-1} z \\ &= (\Pi^{-1} + S_\alpha)^{-1} \Pi^{-1} [\Pi H'(y - h(x_o)) + x_o] \\ &= (\Pi^{-1} + S_\alpha)^{-1} [H'(y - h(x_o)) \\ &\quad + \Pi^{-1} x_o + S_\alpha x_o - S_\alpha x_o] \\ &= x_o + (\Pi^{-1} + S_\alpha)^{-1} [H'(y - h(x_o)) - S_\alpha x_o] \\ &= \hat{x}_{\text{Hess}} \end{aligned}$$

we see that the Hessian approach and the linearization approach of Section II are equivalent, i.e., $\hat{x}_{\text{Lin}} = \hat{x}_{\text{Hess}}$. Using this equivalence, we can translate the relaxation parameter idea back into the spline-smoothing formulation. By the same arguments as above, if we define

$$\hat{x}_{\text{Lin}, \lambda} \triangleq (\Pi_\lambda^{-1} + S_\alpha)^{-1} \Pi_\lambda^{-1} z_\lambda$$

where

$$z_\lambda \triangleq \Pi_\lambda H'(y - h(x_o)) + x_o$$

and

$$\Pi_\lambda^{-1} \triangleq \Pi^{-1} + \lambda \Lambda$$

then $\hat{x}_{\text{Lin}, \lambda} = \hat{x}_{\text{Hess}, \lambda}$. In words, rather than smoothing the pseudomeasurements z_n (with covariances Π_n), we smooth $z_{n, \lambda}$ (with covariances $\Pi_{n, \lambda}$). This estimation procedure is translated into an algorithm in Section V.

IV. CHOOSING THE SMOOTHING PARAMETER

As in the linear case, we want to choose the smoothing parameter α to provide good estimates of g . One method with intuitive appeal and high (statistical) efficiency (as shown in

[4]) in the linear case is to choose the smoothing parameter that minimizes the cross validation (CV) score, defined by

$$\text{CV}(\alpha) \triangleq \frac{1}{N} \sum_{n=1}^N \|y_n - h_n(\hat{g}_{\alpha, -n}(t_n))\|^2 \quad (9)$$

where

$$\begin{aligned} \hat{g}_{\alpha, -i} &\triangleq \arg \min_g \sum_{n=1, n \neq i}^N \|y_n - h_n(g(t_n))\|^2 \\ &\quad + \sum_{m=1}^M \alpha_m \int (\hat{g}_m(t))^2 dt. \end{aligned}$$

$\hat{g}_{\alpha, -i}$ is the solution to the smoothing problem posed without data point y_i . Exact evaluation of (9) is impractical since it would require N iterative smoothing problems for each value of α . Motivated by the corresponding formula for the linear case [4, eq. (17)], we propose the following approach: for a given value of α , compute \hat{x}_α , and use \hat{x}_α to compute the linearized measurement z , the Jacobian H , and the covariance $\Pi = (H'H)^{-1}$. Then an approximation for $\text{CV}(\alpha)$ is

$$\text{CV}_0(\alpha) \triangleq \frac{1}{N} \sum_{n=1}^N \left\| \Pi_n^{-(1/2)} (I_M - A_{(nn)}(\alpha))^{-1} (z_n - \hat{x}_{\alpha, n}) \right\|^2 \quad (10)$$

where (cf. (6))

$$A(\alpha) \triangleq (\Pi^{-1} + S_\alpha)^{-1} \Pi^{-1}$$

and $A_{(nn)}(\alpha)$ is the n th $M \times M$ block diagonal submatrix of $A(\alpha)$. This approximation is based on the expectation that \hat{x}_α will be close enough to g that the Taylor expansion (4) will be accurate. Once \hat{x}_α is computed, (10) is evaluated in $O(M^3N)$ operations as discussed in [4]. The accuracy of the approximation used in deriving CV_0 is less important than whether or not the minimum of CV_0 occurs at a value of α for which \hat{x}_α is a good estimate. In Section VI we show an empirical result that indicates the utility of CV_0 .

V. ALGORITHM

The algorithm depicted in Table I iteratively computes \hat{x}_α for a particular value of α . The computational complexity is only $O(M^3N)$. We have borrowed ideas from [11, sec. 14.4], substituting in our optimality criteria. All operations containing terms with the subscript n are repeated for $n = 1, \dots, N$. Source code for this algorithm is available as VSPLINE from NETLIB [12]. The dominant computational requirements are the vector-spline smoothing and the computation of CV_0 . Since these computations are required even in the linear measurement case, the principal "penalty" incurred when considering nonlinear problems is the necessity of iteration.

The algorithm of Table I is implemented as a procedure that returns $\text{CV}_0(\alpha)$. This procedure is typically called with several different values of α to minimize $\text{CV}_0(\alpha)$. We use the subroutine given in [11] for Powell's method for this minimization. We can make considerable computational savings by using the smoothed estimates for one value of α as the initial state when smoothing for a nearby value of α . Using this procedure, we have found empirically that although the smoothing algorithm may require six to ten iterations for the first value of α , on subsequent calls the smoothing procedure typically converges to within 0.1% of $\min_x \Phi_\alpha(x)$ in just one or two steps.

TABLE I
ITERATIVE NONLINEAR ESTIMATION ALGORITHM AND COMPUTATIONAL
REQUIREMENTS

Computation	Flops
Obtain initial estimate \hat{x}_0	?
$\hat{x} := \hat{x}_0$	
$\lambda := 0.001$	
$\hat{y}_n := h_n(\hat{x}_n)$?
$H_n :=$ Jacobian of h_n at \hat{x}_n	?
$e_n := y_n - \hat{y}_n$	$N(L)$
$w_n := H_n^T e_n$	$N(2ML)$
$\Pi_n^{-1} := (H_n^T H_n)^{-1}$	$N(2M^2L)$
$f_{\text{best}} := \Phi_a(\hat{x})$	$N(14M)$
$\Lambda_n :=$ diagonal elements of Π_n^{-1}	
repeat {	
$\Pi_{n,\lambda}^{-1} := \Pi_n^{-1} + \lambda \Lambda_n$	$N(2M)$
Invert $\Pi_{n,\lambda}^{-1}$	$N(M^3)$
$z_{n,\lambda} := \hat{x}_n + \Pi_{n,\lambda}^{-1} w_n$	$N(2M^2)$
$\hat{x}_{\text{lin},\lambda} :=$ vector-spline smooth [4] $\{z_{n,\lambda}\}$, with covariances $\{\Pi_{n,\lambda}\}$	$N(\frac{27}{2}M^3)$
$\hat{y}_n = h_n(\hat{x}_{\text{lin},\lambda})$?
$\hat{e}_n = y_n - \hat{y}_n$	$N(L)$
$f_{\text{new}} := \Phi_a(\hat{x}_{\text{lin},\lambda})$	$N(7M)$
if ($f_{\text{new}} < f_{\text{best}}$)	
$\hat{x} := \hat{x}_{\text{lin},\lambda}$	
$f_{\text{best}} := f_{\text{new}}$	
$\hat{y}_n := \hat{y}_n$	
$H_n :=$ Jacobian of h_n at \hat{x}_n	?
$w_n := H_n^T \hat{e}_n$	$N(2ML)$
$\Pi_n^{-1} := (H_n^T H_n)^{-1}$	$N(2M^2L)$
$\Lambda_n :=$ diagonal elements of Π_n^{-1}	
$\lambda := 0.1\lambda$	1
else	
$\lambda := 10\lambda$	1
} until $\Phi_a(\hat{x})$ decreases insignificantly.	
compute CV_0 score [4] for \hat{x}_a	$N(\frac{37}{2}M^3)$

In the examples of Section VI, the "else" section of this procedure was rarely executed, hence the iterations converged nearly quadratically to the estimate \hat{x}_a .

VI. EXAMPLE APPLICATIONS

A. Edge Estimation

One simple application of the nonlinear smoothing algorithm described above is to the problem of estimating the position of edges in digital images. Consider Figs. 1 and 2; each of the ($N = 64$) rows of these images contains ($L = 64$) samples of a step function of unknown shift ($M = 1$). If the edge is known to be straight, then high accuracy techniques exist for estimating the edge [13]. However, if the edge is smoothly varying curve, the nonlinear estimation approach of this paper is applicable.

An approximate model for the measurement function for this problem is

$$h_i(\tau) \triangleq \int_{i-1}^i \mathbf{1}_{\{s \leq \tau\}} ds \quad (11)$$

where

$$\mathbf{1}_{\{s \leq \tau\}} = \begin{cases} 1, & s \leq \tau \\ 0, & s > \tau \end{cases}$$

with corresponding Jacobian:

$$\frac{\partial h_i(\tau)}{\partial \tau} = \mathbf{1}_{\{|i-\tau-1/2| \leq 1/2\}}$$

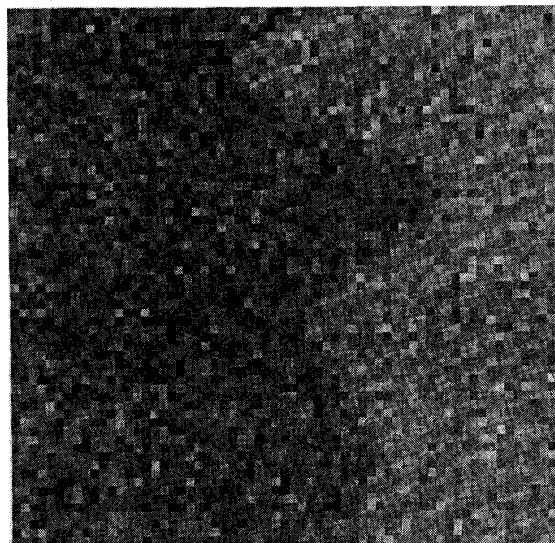


Fig. 1. Noisy image data for curved edge estimation example.

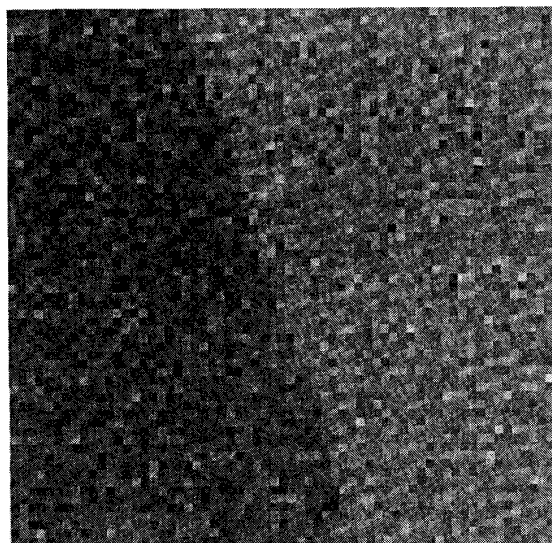


Fig. 2. Noisy image data for straight edge estimation example.

We generated the data displayed in Figs. 1 and 2 by using (11) and adding Gaussian noise with variance $\sigma^2 = 0.25$. The resulting SNR ($=1/\sigma$) is 2.

Assuming that the underlying edge is smoothly varying (which Figs. 1 and 2 do seem to suggest), the only remaining requirement for the nonlinear smoothing algorithm is to provide an initial estimate. We used the following simple heuristic: a temporary copy of each row of the image was convolved with an approximate matched filter kernel $[1, 1, 1, 1, 1, 1, 0, -1, -1, -1, -1, -1]$, and the index of the pixel with maximum value was stored. This set of $N = 64$ numbers was then median filtered, and the result was the initial estimate of the edge position.

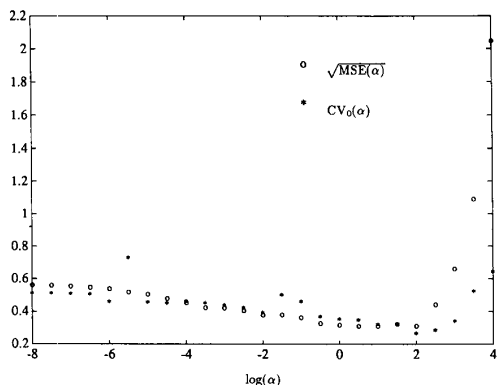


Fig. 3. Comparison of MSE and CV_0 for curved edge example.

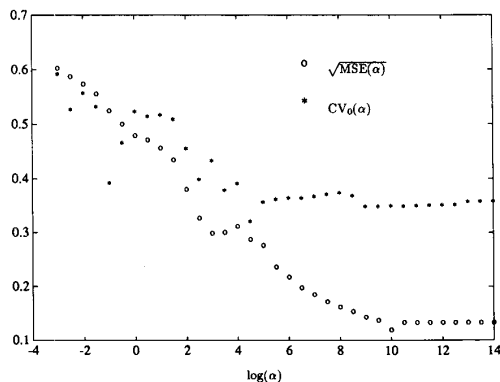


Fig. 4. Comparison of MSE and CV_0 for straight edge example.

We do not have any reason to stipulate a particular smoothing parameter, so we use cross validation. To verify the CV_0 approximation, we show in Fig. 3 a plot of the mean-squared error and the CV_0 score as a function of α for the data set shown in Fig. 1, where

$$MSE(\alpha) \triangleq \frac{1}{N} \sum_{n=1}^N \|\hat{g}_\alpha(t_n) - g(t_n)\|^2.$$

The minimum of the CV_0 curve is very close to the minimum of the MSE curve, thus our approximation for the CV score is useful for achieving accurate estimates. The underlying curve in Fig. 2 is truly a straight line. Hence, as shown in Fig. 4, the MSE is monotonically decreasing with increasing α . Because of the low signal-to-noise ratio, the CV_0 score decreases to a certain point and then increases again. Nevertheless, the minimum of CV_0 does occur where the MSE is reasonably small.

Figs. 5 and 6 show a comparison of the true and the estimated position functions for the optimal α 's. The algorithm adapted itself to both the curved edge and the straight edge—choosing a much larger value for the smoothing parameter in the latter case. This example highlights the versatility of this nonparametric paradigm. Fig. 7 shows plots of the estimation errors for the above examples. The subpixel errors demonstrate the estimation accuracy of this approach.

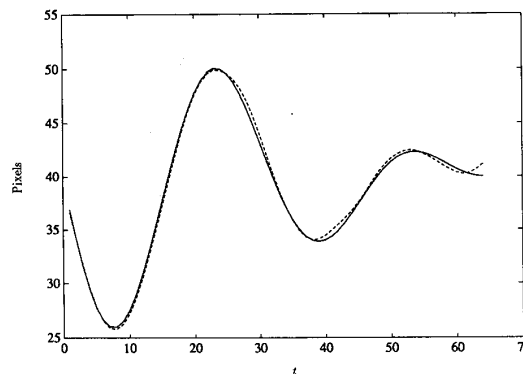


Fig. 5. True (solid) and estimated (dashed) edge position from Fig. 1.

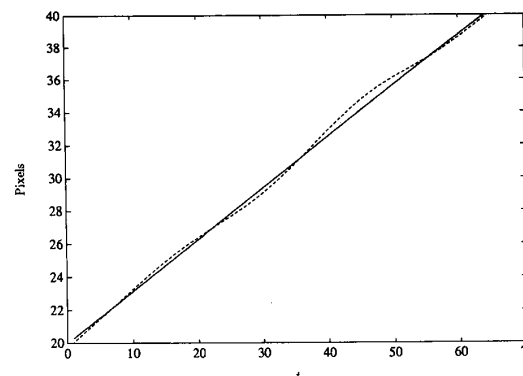


Fig. 6. True (solid) and estimated (dashed) edge position from Fig. 2.

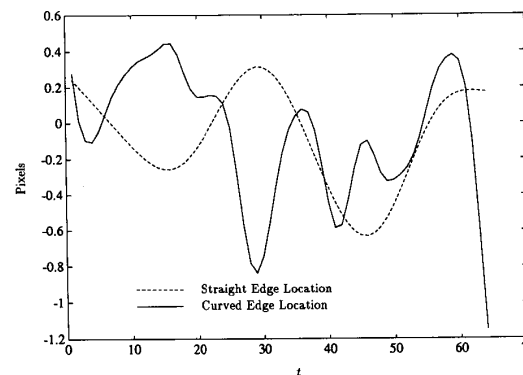


Fig. 7. Estimation errors for edge estimation examples.

B. Quantitative Angiography

In this section, we consider a medical image application: estimating the position and radius of a cylindrical blood vessel from a noisy angiogram (line-integral projection measurement). We simplify the problem considerably here, see [9], [14] for thorough discussions. Fig. 8 is a simulated angiogram. Each of the $N = 64$ rows of this image contains $L = 64$ samples of a shifted semiellipse “bump” whose radius and position func-

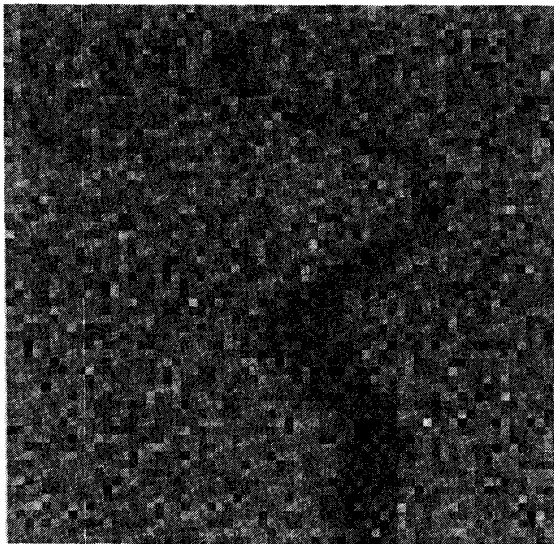


Fig. 8. Simulated angiogram measurements.

tions are to be estimated ($M = 2$). The following definition of the measurement function is an accurate model:

$$y_{n,i} = h_i(g_1(n), g_2(n)) + \epsilon_{n,i},$$

$$i = 1, \dots, L, \quad n = 1, \dots, N$$

where

$$h_i(\tau, r) \triangleq 2 \int_{i-1}^i \sqrt{r^2 - (s - \tau)^2} \mathbf{1}_{\{|s - \tau| \leq r\}} ds$$

$$= [r^2(\mu_+ \sqrt{1 - \mu_+^2} + \arcsin \mu_+) - r^2(\mu_- \sqrt{1 - \mu_-^2} + \arcsin \mu_-)] \cdot \mathbf{1}_{\{|i - \tau - 1/2| \leq r + 1/2\}} \quad (12)$$

and

$$\mu_+ \triangleq \min \left\{ 1, \left| \frac{i - \tau}{r} \right| \right\}, \quad \mu_- \triangleq \min \left\{ -1, \left| \frac{i - 1 - \tau}{r} \right| \right\}.$$

The additive noise is independent normal with variance σ^2 . The Jacobians are

$$\frac{\partial h_i(\tau, r)}{\partial \tau} = -2r(\sqrt{1 - \mu_+^2} - \sqrt{1 - \mu_-^2}) \mathbf{1}_{\{|i - \tau - 1/2| \leq r + 1/2\}}$$

and

$$\frac{\partial h_i(\tau, r)}{\partial r} = 2(\arcsin(\mu_+) - \arcsin(\mu_-)) \mathbf{1}_{\{|i - \tau - 1/2| \leq r + 1/2\}}.$$

Equipped with *a priori* knowledge of the smoothness of blood vessels, and having specified the measurement model, all that we need to apply the smoothing algorithm to the measurements of Fig. 8 is an initial estimate. We again used the matched filter/median filter heuristic to compute the initial position, with a "boxcar" kernel: [11111] for the matched filter. We arbitrarily initialized the radius to be 5 pixels; in a typical clinical setting the initial radius would be set to the normal size of the particular artery being studied. The measurements shown in Fig. 8 were

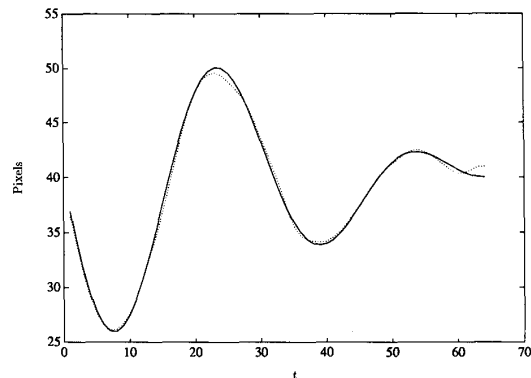


Fig. 9. True (solid) and estimated (dotted) position from Fig. 8.

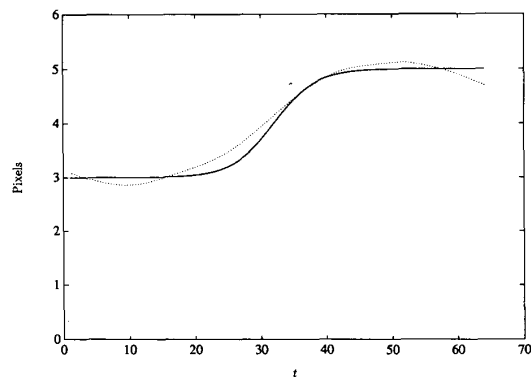


Fig. 10. True (solid) and estimated (dotted) radius from Fig. 8.

generated using (12) and adding pseudorandom white Gaussian noise with variance $\sigma^2 = 16$. The resulting SNR ($=2r/\sigma$) ranges from 1.5 to 2.5.

Figs. 9 and 10 are plots of the true and the estimated position and radius functions, where the smoothing parameter was chosen by minimizing the CV_0 score over both α_1 and α_2 . The rms estimation errors for position and radius were 0.1 and 0.04 pixels, respectively. Such subpixel estimation accuracy justifies the computational effort of this global approach.

VII. CONCLUSIONS

We have presented an iterative algorithm for nonlinear estimation of a smooth vector-valued function, based on a nonparametric optimality criterion. This algorithm provides an alternative to the EKF that is useful for off-line processing. We have suggested one approximate method for automatically choosing the smoothing parameter. There are a plethora of methods in use for the linear case, including robust choices [15]; a detailed comparison of these methods in the nonlinear case is an open problem.

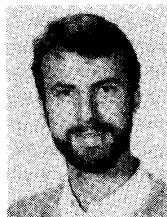
That our algorithm requires an initial estimate for every state is a mixed blessing. Recursive formulae have also been developed for linear spline smoothing [16], [17]. Perhaps an extension of this work would yield a recursive nonlinear smoother that would only require a single initial state.

In this paper, we have demonstrated the potential of this algorithm on two very simple applications. In other work, we

have used this algorithm extensively for the problem of reconstructing three-dimensional arterial tree descriptions from limited projections [18]. Another useful application may be to the field of biomechanics [19]. We look forward to hearing of other problems for which the nonparametric approach may be more suitable than parametric methods.

REFERENCES

- [1] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*. Englewood Cliffs, NJ: Prentice-Hall, 1979.
- [2] B. Silverman, "Some aspects of the spline smoothing approach to nonparametric regression curve fitting," *J. Roy. Stat. Soc., ser. B*, vol. 47, no. 1, pp. 1-52, 1985.
- [3] R. L. Eubank, *Spline Smoothing and Nonparametric Regression*. New York: Marcel Dekker, 1988.
- [4] J. A. Fessler, "Nonparametric fixed-interval smoothing with vector splines," *IEEE Trans. Signal Processing*, this issue, pp. 852-859.
- [5] F. O'Sullivan, "A statistical perspective on ill-posed inverse problems," *Stat. Sci.*, vol. 1, no. 4, pp. 502-527, 1986.
- [6] I. J. Schoenberg, "On interpolation by spline functions and its minimal properties," in *On Approximation Theory, Proc. Conf. in Oberwolfach, Black Forest*, Aug. 1963, pp. 109-129.
- [7] H. J. Woltring, "A Fortran package for generalized cross-validatory spline smoothing and differentiation," *Adv. Eng. Software*, vol. 8, no. 2, pp. 104-112, 1986.
- [8] Y. Bresler and A. Macovski, "Nonlinear estimation of dynamic shift and shape parameters from set function measurements," in *Automat. Contr. Conf. Proc.* (San Diego, CA), June 1984.
- [9] Y. Bresler and A. Macovski, "Three-dimensional reconstruction from projections with incomplete and noisy data by object estimation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 35, pp. 1139-1152, Aug. 1987.
- [10] D. W. Marquardt, "An algorithm for least squares estimation of nonlinear parameters," *J. Soc. Ind. Appl. Math.*, vol. 11, pp. 431-441, June 1963.
- [11] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in C*. Cambridge University Press, 1988.
- [12] J. J. Dongarra and E. Grosse, "Distribution of mathematical software via electronic mail," *Commun. Ass. Comput. Mach.*, vol. 30, pp. 403-407, Oct. 1987.
- [13] D. Petkovic, W. Niblack, and M. Flickner, "Projection-based high accuracy measurement of straight line edges," *Mach. Vision Appl.*, vol. 1, pp. 183-199, 1988.
- [14] T. Pappas and J. Lim, "A new method for estimation of coronary artery dimensions in angiograms," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 1501-1513, Sept. 1988.
- [15] T. Robinson and R. Moyeed, "Making robust the cross-validatory choice of smoothing parameter in spline smoothing regression," *Commun. Statist.—Theory Meth.*, vol. 18, no. 2, pp. 523-539, 1989.
- [16] G. S. Sidhu and H. L. Weinert, "Vector-valued L_g -splines II. Smoothing splines," *J. Math. Anal. Appl.*, vol. 101, pp. 380-396, July 1984.
- [17] W. Wecker and C. Ansley, "The signal extraction approach to nonlinear regression and spline smoothing," *J. Amer. Stat. Ass.*, vol. 78, pp. 81-89, Mar. 1983.
- [18] J. Fessler and A. Macovski, "Object-based 3-D reconstruction of arterial trees from a few projections," *IEEE Trans. Med. Imaging*, Mar. 1991.
- [19] H. J. Woltring, "On optimal smoothing and derivative estimation from noisy displacement data in biomechanics," *Human Movement Sci.*, vol. 4, no. 3, pp. 229-230, 1985.



Jeffrey A. Fessler received the B.S.E.E. degree from Purdue University in 1985, the M.S.E.E. degree in 1986, the M.S. degree in statistics in 1989, and the Ph.D. degree in electrical engineering in 1990, all from Stanford University.

From 1985 to 1988 he was a National Science Foundation Graduate Fellow at Stanford. He is currently a postdoctoral fellow in the Department of Nuclear Medicine at the University of Michigan. His research interests include estimation and pattern recognition and their applications in medical imaging and bioengineering.