Monte-Carlo Sure: A Black-Box Optimization of Regularization Parameters for General Denoising Algorithms

Sathish Ramani, *Student Member, IEEE*, Thierry Blu, *Senior Member, IEEE*, and Michael Unser, *Fellow, IEEE*

***Abstract*—We consider the problem of optimizing the parameters of a given denoising algorithm for restoration of a signal corrupted by white Gaussian noise. To achieve this, we propose to minimize *Stein's unbiased risk estimate* (SURE) which provides a means of assessing the true mean-squared error (MSE) purely from the measured data without need for any knowledge about the noise-free signal. Specifically, we present a novel Monte-Carlo technique which enables the user to calculate SURE for an arbitrary denoising algorithm characterized by some specific parameter setting. Our method is a black-box approach which solely uses the response of the denoising operator to additional input noise and does not ask for any information about its functional form. This, therefore, permits the use of SURE for optimization of a wide variety of denoising algorithms. We justify our claims by presenting experimental results for SURE-based optimization of a series of popular image-denoising algorithms such as total-variation denoising, wavelet soft-thresholding, and Wiener filtering/smoothing splines. In the process, we also compare the performance of these methods. We demonstrate numerically that SURE computed using the new approach accurately predicts the true MSE for all the considered algorithms. We also show that SURE uncovers the optimal values of the parameters in all cases.**

***Index Terms*—Monte-Carlo methods, regularization parameter, smoothing splines, Stein's unbiased risk estimate (SURE), total-variation denoising, wavelet denoising.**

I. Introduction

IMAGES are often corrupted by noise during the acquisition process. Denoising aims at eliminating this measurement noise while trying to preserve important signal features such as texture and edges. Over the past few decades, a large variety of algorithms has been developed for that purpose. They can be roughly categorized into linear denoising methods such as Wiener filtering and smoothing splines, variational and partial-differential-equation-based (PDE) methods that use nonquadratic regularization functionals such as total-variation, and multiresolution methods such as wavelet denoising. Formally, any denoising algorithm can be thought of as an operator $\mathbf{f}_{\boldsymbol{\lambda}}$ (which depends on the set of parameters $\boldsymbol{\lambda}$) that maps the noisy data $\mathbf{y}$ onto the signal estimate $\tilde{\mathbf{x}} = \mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})$. When applying a particular algorithm, the user is faced with the difficult task of adjusting $\boldsymbol{\lambda}$ to obtain best performance. To accomplish this, researchers usually resort to empirical methods or pose the problem in a Bayesian framework. Empirical methods have proliferated, especially in the variational context where one of the key problems is the selection of the "best" regularization parameter. The most common techniques include the use of the discrepancy principle [1], generalized cross validation (GCV) [1]–[7], and the L-curve methods [8]–[11]. Alternatively, the problem can also be formulated in a Bayesian framework by imposing model-based constraints as prior knowledge on the noise-free signal [12]–[15].

In a denoising scenario, the mean-squared error (MSE) of the signal estimate is the preferred measure of quality to optimize $\boldsymbol{\lambda}$. Unfortunately, the MSE depends on the noise-free signal which is generally unavailable or unknown *a priori*. A practical approach, therefore, is to replace the true MSE of $\tilde{\mathbf{x}}$ by some estimate in the scheme of things. A theoretical result due to Stein [16] makes this possible in the Gaussian scenario. *Stein's unbiased risk estimate*—SURE, as it is called—provides a means for unbiased estimation of the true MSE. Without ever requiring knowledge of the noise-free signal, this unbiased estimate solely depends on the given data and on some description of the first-order dependence of the denoising operator with respect to the data. The unbiasedness of SURE can be mathematically established, which makes it nonempirical. Moreover, the closeness of SURE to the true MSE is aided by the law of large numbers for large data size (especially images).

The divergence of the denoising operator $\mathbf{f}_{\boldsymbol{\lambda}}$ with respect to $\mathbf{y}$ is the key ingredient of SURE [16]. It can be computed analytically only in some special cases such as when the denoising operator performs a coordinate-wise nonlinear mapping, when the signal estimate is obtained by a linear transformation of the noisy data (linear filtering [7]), or when both are combined in a specific way (e.g., wavelet thresholding [17]–[20]). For linear algorithms, the desired divergence reduces to the trace of the corresponding matrix transformation. However, in a general setting, the explicit evaluation of the divergence is often out of reach. Especially challenging are cases where the functional form of the denoising operator is not known, for example when the denoised output is the result of an iterative optimization procedure. Since most of the variational and Bayesian methods fall into this category, there are many key algorithms for which the evaluation of the required divergence term is neither tractable mathematically nor even feasible numerically.

Manuscript received November 20, 2007; revised April 25, 2008. Published August 13, 2008 (projected). The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Eric Kolaczyk.

S. Ramani and M. Unser are with the Biomedical Imaging Group (BIG), Ecole polytechnique fédérale de Lausanne (EPFL), CH-1015 Lausanne, Switzerland (sathish.ramani@epfl.ch; michael.unser@epfl.ch).

T. Blu is with the Department of Electronic Engineering, The Chinese University of Hong Kong, Shatin, N.T., Hong Kong (e-mail: thierry.blu@m4x.org).

Digital Object Identifier 10.1109/TIP.2008.2001404

1057-7149/$25.00 © 2008 IEEE

In this paper, we address this limitation by proposing a novel scheme that is applicable for a general denoising scenario. Our method is based on Monte-Carlo simulation: the denoising algorithm is probed with additive noise and the response signal is manipulated to estimate the desired divergence. This leads to a black-box interpretation of the proposed technique—it completely relies on the output of the denoising operator and does not need any information about its functional form. We validate the proposed scheme by presenting numerical results for a variety of popular denoising methods—total-variation (TV) denoising, redundant-wavelet soft-thresholding, and some classical ones such as orthonormal-wavelet soft-thresholding and smoothing splines.

The paper is structured as follows. After setting up the problem in Section II, we provide a brief overview of the SURE theory in Section III. In Section IV, we present Monte-Carlo strategies for estimating the MSE of a particular denoising algorithm. First, we propose a simple scheme for the special case of linear algorithms and then proceed to describe a new method for arbitrary nonlinear operators. In Section V, we present experimental results and demonstrate numerically that SURE, computed using the new Monte-Carlo strategy, faithfully imitates the true MSE curve. Moreover, it is always capable of uncovering the optimal value of the parameter (regularization parameter for the variational methods and soft-threshold value for the wavelet-based methods). Additionally, we illustrate that the proposed scheme is applicable for denoising methods characterized by multiple parameters. In the process, we also compare the performance of these denoising algorithms in terms of visual quality and signal-to-noise ratio (SNR). We finally draw our conclusions in Section VI.

II. Notation and Problem Formulation

We adopt the standard vector formulation of a denoising problem: We observe the noisy data $\mathbf{y} \in \mathbb{R}^N$ given by

$$\mathbf{y} = \mathbf{x} + \mathbf{b} \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^N$ represents the vector containing the samples of the unknown deterministic noise-free signal and $\mathbf{b} \in \mathbb{R}^N$ denotes the vector containing zero-mean white Gaussian noise of variance σ^2, respectively. We are given a denoising algorithm which is represented by the operator $\mathbf{f}_{\boldsymbol{\lambda}} : \mathbb{R}^N \rightarrow \mathbb{R}^N$ that maps the input data $\mathbf{y}$ onto the signal estimate

$$\tilde{\mathbf{x}} = \mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y}) \tag{2}$$

where $\boldsymbol{\lambda}$ represents the set of parameters characterizing $\mathbf{f}_{\boldsymbol{\lambda}}$.

Our primal aim in this work is to optimize $\boldsymbol{\lambda}$ knowing only $\mathbf{y}$ and $\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})$ as illustrated by the "MSE estimation" box in Fig. 1. To achieve this, we propose the use of SURE as a reliable estimate of the true MSE. SURE computation is greatly simplified if the denoising is performed by coordinate-wise filtering in an orthonormal transform domain (e.g., Fourier transform, orthonormal wavelet transform, which preserve the MSE during the transformation). However, complications appear as soon as the transform becomes nonorthogonal or redundant. Then, one is forced to compute SURE in the signal domain, which may or may not be mathematically tractable depending on the type of filtering that is applied.

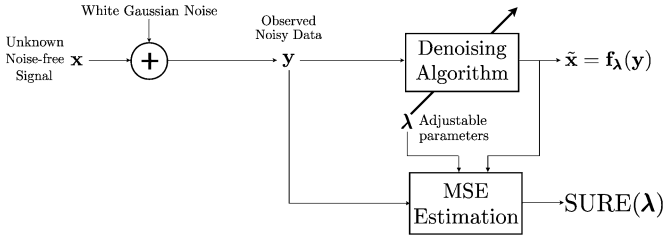

Fig. 1. Signal estimate $\tilde{\mathbf{x}}$ is obtained by applying the $\boldsymbol{\lambda}$-dependent denoising algorithm on the observed data $\mathbf{y}$. The MSE box then computes the estimate SURE ($\boldsymbol{\lambda}$) of the MSE between the noise-free $\mathbf{x}$ and the denoised $\tilde{\mathbf{x}}$ as a function of $\boldsymbol{\lambda}$, knowing only $\mathbf{y}$ and $\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})$. The best estimate of the signal is obtained by finding that $\boldsymbol{\lambda}$ which minimizes the surrogate mean-squared error.

In the variational framework, the denoised output is obtained in general by minimizing the problem-specific cost functional

$$\mathbf{f}_{\lambda}(\mathbf{y}) = \arg\min_{\mathbf{u}} \mathcal{J}(\mathbf{y}, \mathbf{u}) \tag{3}$$

$$\mathcal{J}(\mathbf{y}, \mathbf{u}) = \mathcal{D}(\mathbf{y}, \mathbf{u}) + \lambda \mathcal{R}(\mathbf{u}) \tag{4}$$

where $\mathcal{D}(\cdot, \cdot)$ is the data fidelity term that measures the consistency of $\mathbf{u}$ to the given data, while $\mathcal{R}(\,\cdot\,)$ is a suitable regularization functional that often penalizes a lack of smoothness in $\mathbf{u}$. When $\mathcal{J}$ is quadratic in $\mathbf{u}$, $\mathbf{f}_{\lambda}$ becomes linear. However, for most other $\mathcal{J}$, $\mathbf{f}_{\lambda}$ is nonlinear, in which case it is usually not possible to write a closed-form expression for $\mathbf{f}_{\lambda}$. The corresponding estimation is typically implemented iteratively by running a suitable optimization procedure that may involve large-scale image-domain filtering.

In the above variational formulation, $\boldsymbol{\lambda} = \lambda$ is a positive scalar that controls the amount of regularization imposed on the solution. When $\lambda \rightarrow 0$, the solution tends to fit the data $\mathbf{y}$ more closely (implying a less significant noise reduction), while a large value of λ yields a solution that is heavily constrained (typically resulting in a loss of features and over-smoothing). Thus, the choice of the appropriate λ is crucial. Much effort has been dedicated to this problem [1], [21]. The primary techniques to optimize λ can be broadly classified as follows:

1) use of the discrepancy principle [1], [6], [7];
2) L-curve based methods [8]–[11];
3) Bayesian methods [12]–[15];
4) the C_L criterion [22];
5) MSE-based methods [6], [7], [23];
6) generalized cross validation (GCV) [1]–[7].

The discrepancy principle selects λ by matching data fidelity term to noise variance; this generally yields over-penalized solutions [7]. The L-curve methods are entirely deterministic and choose λ by "balancing" the effect of data-fidelity and regularization terms, while Bayesian methods have a statistical interpretation in terms of Baye's rule and assume some prior knowledge on the noise-free signal. The C_L criterion requires the knowledge of σ^2 and was originally proposed for linear methods [22]. Moreover, it has been noted in [24] that, for linear algorithms, C_L is an unbiased estimate of MSE (up to a constant). Some researchers in signal processing have also made explicit attempts to minimize an estimate of the MSE but these methods

are either restricted to the case of a linear estimator [6], [7] or they are largely empirical [23].

The most popular method for linear algorithms is probably GCV which does not require the knowledge of the noise variance. GCV is based on the "leave-one-out" principle [2]–[5] and is known to yield λ which asymptotically minimizes (under certain hypotheses) the true MSE [25]. In [24], Girard proposed Monte-Carlo versions of GCV and C_L (namely, RGCV and RC_L) for linear algorithms when the associated quantities are not explicitly computable. Following this, an extension of RGCV for "mildly" nonquadratic (nonlinear) problems was suggested by Wahba in [26] and [27] and by Girard in [28]. In this paper, we propose an approach that is similar in spirit to these Monte-Carlo methods but which brings in the following improvements:

1) the proposed method is applicable for algorithms with "arbitrary" nonlinearities;
2) the adjustment of parameters is based on SURE which is optimal even in the nonasymptotic case unlike GCV.

III. Stein's Unbiased Risk Estimate—SURE

In his hallmark paper [16], Stein established the framework for unbiased estimation of the risk (or MSE) of an arbitrary estimator in the presence of Gaussian noise. While SURE is a well-established technique in the statistical literature, it is not so widely known in signal processing. There is a notable exception in the context of (orthonormal) wavelet denoising [17], [18] where the SURE strategy has proven to be quite powerful and has been incorporated in some state-of-the-art algorithms [19], [20], [29]; specifically, SURE-based denoising using nonorthonormal transforms is described in [20]. In what follows, we briefly review the theory of SURE in the context of general nonlinear algorithms. We then illustrate the concept in the simpler case of a linear algorithm, which also yields a closed-form solution.

A. Theoretical Background

In the sequel, we assume that $\mathbf{f}_{\boldsymbol{\lambda}}$ is a continuous and bounded operator (i.e., the input-output mapping is continuous and a small perturbation of the input necessarily results in a small perturbation of the output). We also require that the divergence of $\mathbf{f}_{\boldsymbol{\lambda}}$ with respect to the data $\mathbf{y}$ given by

$$\mathrm{div}_{\mathbf{y}}\{\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\} = \sum_{k=1}^{N} \frac{\partial \mathrm{f}_{\boldsymbol{\lambda}k}(\mathbf{y})}{\partial y_k} \tag{5}$$

where $\mathrm{f}_{\boldsymbol{\lambda}k}(\mathbf{y})$ and y_k represent the kth component of the vectors $\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})$ and $\mathbf{y}$, respectively, is well defined in the weak sense.

Definition 1: Given $\mathbf{y}$ as in (1), SURE corresponding to $\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})$ is a random variable $\eta : \mathbb{R}^N \rightarrow \mathbb{R}$, specified as

$$\eta(\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})) = \frac{1}{N}\|\mathbf{y} - \mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\|^2 - \sigma^2 + \frac{2\sigma^2}{N}\,\mathrm{div}_{\mathbf{y}}\{\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\} \tag{6}$$

where $\|\cdot\|^2$ represents the Euclidean norm. ■

The following theorem, due to Stein [16], then states that η is indeed unbiased.

Theorem 1: The random variable $\eta(\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y}))$ is an unbiased estimator of

$$\mathrm{MSE}(\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})) = \frac{1}{N}\|\mathbf{x} - \mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\|^2 \tag{7}$$

that is

$$E_{\mathbf{b}}\left\{\frac{1}{N}\|\mathbf{x} - \mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\|^2\right\} = E_{\mathbf{b}}\{\eta(\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y}))\} \tag{8}$$

where $E_{\mathbf{b}}\{\cdot\}$ represents the expectation with respect to $\mathbf{b}$. ■

For a proof that is accessible to signal processing audience, see [20]. (it requires the assumption that $\|\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\|$ is bounded by a rapidly increasing function such as $C\exp((\|\mathbf{y}\|^2)/(2(\sigma^2+\epsilon^2)))$; $C, \epsilon > 0$).

In the SURE formulation, the MSE is estimated purely based on the input data $\mathbf{y}$, the divergence of $\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})$, and the noise statistics; it requires no knowledge whatsoever of the noise free signal $\mathbf{x}$. The basis for the approach is that there are many more data points than unknown parameters $\boldsymbol{\lambda}$. Therefore, thanks to the law of large numbers, both $(1)/(N)\|\mathbf{x} - \mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\|^2$ and $\mathrm{div}_{\mathbf{y}}\{\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\}$ are quite stable estimates of $E_{\mathbf{b}}\{(1)/(N)\|\mathbf{x} - \mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\|^2\}$ and $E_{\mathbf{b}}\{\mathrm{div}_{\mathbf{y}}\{\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\}\}$, respectively, meaning that SURE provides a fairly accurate proxy for the true MSE. Hence, it can be applied for data-driven optimization of a wide range of denoising problems. However, the catch with (6) is that the evaluation of $\mathrm{div}_{\mathbf{y}}\{\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\}$ turns out to be difficult or even infeasible when there is no explicit form for the estimator (as is usually the case for iterative algorithms). We close this section by presenting a few cases where the desired divergence takes an explicit form.

B. Special Case: Linear Algorithms

Classical signal-reconstruction algorithms are linear in nature. These are usually associated with quadratic cost functions; the better-known examples are Tikhonov filters [7], [10] and smoothing splines [30]–[33] in the variational setting, MAP estimators under the Gaussian prior [11], [14] and Wiener filter [7], [34] in the stochastic setting. Such estimators can be described by the following matrix transformation:

$$\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y}) = \mathbf{F}_{\boldsymbol{\lambda}}\mathbf{y} \tag{9}$$

where $\mathbf{F}_{\boldsymbol{\lambda}}$ is a $N \times N$ matrix that depends on $\boldsymbol{\lambda}$. Thus, the desired divergence term is explicitly evaluated as

$$\mathrm{div}_{\mathbf{y}}\{\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\} = \mathrm{div}_{\mathbf{y}}\{\mathbf{F}_{\boldsymbol{\lambda}}\,\mathbf{y}\} = \mathrm{trace}\{\mathbf{F}_{\boldsymbol{\lambda}}\} \tag{10}$$

which yields an explicit expression for SURE. In this context, circulant matrices deserve a special mention because their structure can be exploited for efficient computation of the trace as we shall see in Section V-A4.

C. Special Case: Coordinate-Wise Nonlinearity

Let each component of $\mathbf{f}_{\boldsymbol{\lambda}}$ be a nonlinear function of a single argument, that is, the kth component of the output $\tilde{\mathbf{x}}$ is obtained as

$$\tilde{x}_k = \mathrm{f}_{\boldsymbol{\lambda}k}(y_k). \tag{11}$$

In this case, too, the divergence can be analytically evaluated since it amounts to computing the sum of the first derivatives $\mathrm{f}'_{\boldsymbol{\lambda}k}$ of the individual components of $\mathbf{f}_{\boldsymbol{\lambda}}$

$$\mathrm{div}_{\mathbf{y}}\{\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\} = \sum_{k=1}^{N} \frac{\partial \mathrm{f}_{\boldsymbol{\lambda}k}(y_k)}{\partial y_k} = \sum_{k=1}^{N} \mathrm{f}'_{\boldsymbol{\lambda}k}. \tag{12}$$

Even though the coordinate-wise processing described by (11) is not very interesting as such, it becomes quite powerful when applied in a transform domain; in particular, in a wavelet or similar multiresolution transform wherein $\mathrm{f}_{\boldsymbol{\lambda}k}$ is a function of the kth noisy transform coefficient [17]–[20]. The present result is directly transposable to the case of an orthonormal transform which permits exact mapping of the MSE and the divergence between the signal and transform domain using expressions similar to (11) and (12). We are going to illustrate such a case in Section V-A1.

IV. Monte-Carlo Estimation of $\mathrm{div}_{\mathbf{y}}\{\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\}$

The crucial step for evaluating the SURE formula in (6) is the computation of $\mathrm{div}_{\mathbf{y}}\{\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\}$. As we just saw, this can be done explicitly in the cases of linear and coordinate-wise nonlinear estimators [17]–[20]; but it is more difficult otherwise. In this section, we investigate Monte-Carlo techniques to achieve this goal. We start by revisiting a method that is valid in the linear case only [35], [36], but which can be very useful when the matrix $\mathbf{F}_{\boldsymbol{\lambda}}$ is not available explicitly. Following that, we introduce a more general technique that is applicable for arbitrary (nonlinear) algorithms.

A. Linear Algorithm With Unstructured $\mathbf{F}_{\boldsymbol{\lambda}}$

In many practical situations, especially with large data-sets, the matrix $\mathbf{F}_{\boldsymbol{\lambda}}$ is not available explicitly; instead, (9) is implemented iteratively by using some suitable numerical solver (e.g., conjugate gradient, multigrid technique). It follows that the trace is not directly accessible. There are matrix methods (such as the power method) that can produce an estimate of $\mathrm{trace}\{\mathbf{F}_{\boldsymbol{\lambda}}\}$ in an iterative fashion starting from (9), but they tend to be memory- and computation-intensive. To tackle this difficulty, we propose the use of the following Monte-Carlo algorithm which estimates the required trace stochastically with $O(N)$ computational cost [up to the complexity of realizing (9)]. It is implemented by applying the estimator to noise only, as described next.

Algorithm 1: Monte-Carlo algorithm for estimating $(1)/(N)\mathrm{trace}\{\mathbf{F}_{\boldsymbol{\lambda}}\}$

- Generate a zero-mean i.i.d. random vector $\mathbf{b}'$ of unit variance.
- For a given $\boldsymbol{\lambda} = \boldsymbol{\lambda}_0$ do the following:
  1. Evaluate $\tilde{\mathbf{b}} = \mathbf{F}_{\boldsymbol{\lambda}}\mathbf{b}'$ for $\boldsymbol{\lambda} = \boldsymbol{\lambda}_0$
  2. Compute the estimate of $(1)/(N)\mathrm{trace}\{\mathbf{F}_{\boldsymbol{\lambda}}\}$ as $(1)/(N)\mathbf{b}'^{\mathrm{T}}\tilde{\mathbf{b}}$

Algorithm 1 is a standard procedure in the literature [35], [36] and has a twofold advantage over the iterative matrix methods mentioned before: first, it is memory-efficient because, at any given point, it only stores $\mathbf{F}_{\boldsymbol{\lambda}}\mathbf{b}'$ and not $\mathbf{F}_{\boldsymbol{\lambda}}$ itself. Second, from a computation point of view, the method is as good as the initial algorithm itself since we can simply apply it to noise. The validity of the algorithm is guaranteed by the fact that the random variable $\mathbf{b}'^{\mathrm{T}}\mathbf{F}_{\boldsymbol{\lambda}}\mathbf{b}'$ is an unbiased estimator of $\mathrm{trace}\{\mathbf{F}_{\boldsymbol{\lambda}}\}$, which is a well-established result in the literature [35]–[38].

Proposition 1: Let $\mathbf{b}'$ be a zero-mean i.i.d. random vector with unit variance and $\hat{t} = (1)/(N)\mathbf{b}'^{\mathrm{T}}\mathbf{F}_{\boldsymbol{\lambda}}\mathbf{b}'$, where the factor $(1)/(N)$ accounts for the averaging of the MSE (7) over all samples. Then

$$E_{\mathbf{b}'}\{\hat{t}\} = \frac{1}{N}\mathrm{trace}\{\mathbf{F}_{\boldsymbol{\lambda}}\}.\blacksquare \tag{13}$$

For image-processing applications, it is reasonable to believe that a single realization of $\mathbf{b}'$ will yield a sufficiently low variance estimate [35], [24]. This is because, in practice, most denoising algorithms operate only "locally" (i.e., $\mathbf{F}_{\boldsymbol{\lambda}}$ is more or less diagonal with rapidly decaying off-diagonal elements). Qualitatively speaking, the components $\{\tilde{b}_i\}_{i=1}^{N}$ of $\tilde{\mathbf{b}}$ are, therefore, "nearly" independent. Since N is large for images (typically $N \geq 256^2$), by law of large numbers $\hat{t} - E_{\mathbf{b}'}\{\hat{t}\}$ does not fluctuate more than $(1)/(\sqrt{N})$; this eliminates any necessity for additional algorithm evaluations. A more quantitative argument can be made by computing the variance of $\hat{t}$ which is given by $\mathrm{Var}_{\mathbf{b}'}\{\hat{t}\} = (1)/(N^2)(\mathrm{trace}\{\mathbf{F}_{\boldsymbol{\lambda}}^{\mathrm{T}}\mathbf{F}_{\boldsymbol{\lambda}}\} + \mathrm{trace}\{\mathbf{F}_{\boldsymbol{\lambda}}^2\} + (E_{b'}\{b'^4\} - 3)\sum_{k=1}^{N} F_{kk}^2)$, where F_{kk} is the kth diagonal element of $\mathbf{F}_{\boldsymbol{\lambda}}$ and $E_{b'}\{b'^4\}$ is the fourth-order moment of the random variable b'. Again, since $\mathbf{F}_{\boldsymbol{\lambda}}$ is "approximately" diagonal, the quantities $\mathrm{trace}\{\mathbf{F}_{\boldsymbol{\lambda}}^{\mathrm{T}}\mathbf{F}_{\boldsymbol{\lambda}}\}$ and $\mathrm{trace}\{\mathbf{F}_{\boldsymbol{\lambda}}^2\}$ are of the order of N. The variance is then bounded as $\mathrm{Var}_{\mathbf{b}'}\{\hat{t}\} \leq \mathrm{constant}/N$. Thus, in principle, $\hat{t}$ asymptotically converges to $(1)/(N)\mathrm{trace}\{\mathbf{F}_{\boldsymbol{\lambda}}\}$ in the mean-squared error sense. A further option is to reduce $\mathrm{Var}_{\mathbf{b}'}\{\hat{t}\}$ by selecting a $\mathbf{b}'$ that has small a fourth-order moment. For instance, it has been suggested to choose $\mathbf{b}'$ such that its components are either $+1$ or -1 with probability 0.5 [36]–[38]; for such a $\mathbf{b}'$, the variance is lower than that obtained using a Gaussian $\mathbf{b}'$ [36], [38].

B. General Algorithm for Nonlinear Problems

Similar to the technique described above, our strategy for a nonlinear $\mathbf{f}_{\boldsymbol{\lambda}}$ is essentially based on probing the system with noise, but is slightly more involved because of the nonlinearity of $\mathbf{f}_{\boldsymbol{\lambda}}$. Specifically, we propose to investigate $\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y} + \varepsilon\mathbf{b}')$ which may be thought of as a random perturbation around the operating point of the algorithm. The output is then compared with $\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})$ which yields a differential response of $\mathbf{f}_{\boldsymbol{\lambda}}$ evaluated at $\mathbf{y}$. The following theorem states that this differential response yields the desired divergence as ε decreases.

Theorem 2: Let $\mathbf{b}'$ be a zero-mean i.i.d. random vector (that is independent of $\mathbf{y}$) with unit variance and bounded higher order moments. Then

$$\mathrm{div}_{\mathbf{y}}\{\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\} = \lim_{\varepsilon \to 0} E_{\mathbf{b}'}\left\{\mathbf{b}'^{\mathrm{T}}\left(\frac{\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y} + \varepsilon\mathbf{b}') - \mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})}{\varepsilon}\right)\right\} \tag{14}$$

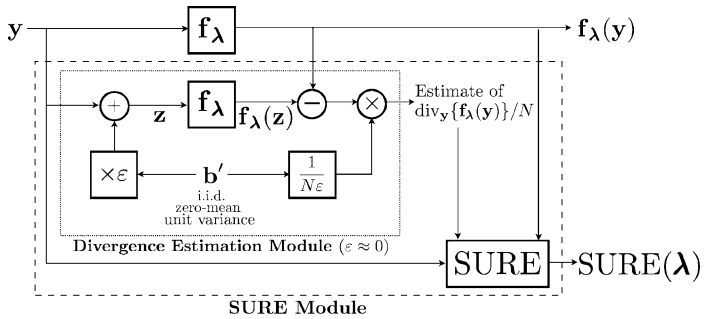

Fig. 2. Dotted box depicts the module that estimates $(1)/(N)\mathrm{div}_{\mathbf{y}}\{\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\}$ according to (17). The dashed box represents the SURE module (depicted as the MSE estimation box in Fig. 1) which computes the SURE according to (6).

provided that $\mathbf{f}_{\boldsymbol{\lambda}}$ admits a well-defined second-order Taylor expansion. Otherwise, the expression is still valid in the weak sense (sufficient to apply Theorem 1) provided that

$$\|\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\| \leq C_0(1+\|\mathbf{y}\|^{n_0}) \tag{15}$$

for some $n_0 > 1$ and $C_0 > 0$ (that is, $\mathbf{f}_{\boldsymbol{\lambda}}$ is tempered).

Proof: We write the second-order Taylor expansion of $\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y}+\varepsilon\mathbf{b}')$ as

$$\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y}+\varepsilon\mathbf{b}') = \mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y}) + \varepsilon\,\mathbf{J}_{\mathbf{f}_{\boldsymbol{\lambda}}}(\mathbf{y})\,\mathbf{b}' + \varepsilon^2\mathbf{r}_{\mathbf{f}_{\boldsymbol{\lambda}}} \tag{16}$$

where $\mathbf{J}_{\mathbf{f}_{\boldsymbol{\lambda}}}(\mathbf{y})$ is the Jacobian matrix of $\mathbf{f}_{\boldsymbol{\lambda}}$ evaluated at $\mathbf{y}$ and $\mathbf{r}_{\mathbf{f}_{\boldsymbol{\lambda}}}$ represents the vector containing the (Lagrange) remainder terms corresponding to each component of $\mathbf{f}_{\boldsymbol{\lambda}}$. In this case, the components $\mathrm{r}_{\mathbf{f}_{\boldsymbol{\lambda}}k}$ of $\mathbf{r}_{\mathbf{f}_{\boldsymbol{\lambda}}}$ are bounded in the expectation sense; that is, $E_{\mathbf{b}'}\{|\mathrm{r}_{\mathbf{f}_{\boldsymbol{\lambda}}k}|\} < +\infty, k = 1, 2, \ldots, N$.

Then, subtracting $\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})$ from (16) and multiplying by $\mathbf{b}'^{\mathrm{T}}$ from the left yields

$$\begin{aligned} &E_{\mathbf{b}'}\{\mathbf{b}'^{\mathrm{T}}(\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y}+\varepsilon\mathbf{b}') - \mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y}))\} \\ &\quad = \varepsilon\,E_{\mathbf{b}'}\left\{\mathbf{b}'^{\mathrm{T}}\,\mathbf{J}_{\mathbf{f}_{\boldsymbol{\lambda}}}(\mathbf{y})\,\mathbf{b}'\right\} + \varepsilon^2 E_{\mathbf{b}'}\left\{\mathbf{b}'^{\mathrm{T}}\mathbf{r}_{\mathbf{f}_{\boldsymbol{\lambda}}}\right\} \\ &\quad = \varepsilon\,\mathrm{trace}\left\{\mathbf{J}_{\mathbf{f}_{\boldsymbol{\lambda}}}(\mathbf{y})\right\} + C_2\varepsilon^2 \end{aligned}$$

where $E_{\mathbf{b}'}\{\mathbf{b}'^{\mathrm{T}}\mathbf{r}_{\mathbf{f}_{\boldsymbol{\lambda}}}\} = C_2$ and $|C_2| < +\infty$ because $\{E_{\mathbf{b}'}\{|\mathrm{r}_{\mathbf{f}_{\boldsymbol{\lambda}}k}|\} < +\infty\}_{k=1}^{N}$ and $\mathbf{b}'$ has bounded higher-order moments. When $\varepsilon \to 0$, we immediately see that

$$\begin{aligned} &\lim_{\varepsilon\to 0}\frac{1}{\varepsilon}E_{\mathbf{b}'}\{\mathbf{b}'^{\mathrm{T}}(\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y}+\varepsilon\mathbf{b}') - \mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y}))\} \\ &\quad = \mathrm{trace}\{\mathbf{J}_{\mathbf{f}_{\boldsymbol{\lambda}}}(\mathbf{y})\} \\ &\quad = \mathrm{div}_{\mathbf{y}}\{\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\} \end{aligned}$$

which yields the desired result.

We could also obtain the proof of the weak form of the result (when the second derivatives are not necessarily well-defined), but is more technical. It involves standard but tedious usage of mathematical tools of measure theory such as the Fubini theorem and the Lebesgue's dominated convergence theorem and is not included in this paper.[1] ■

Theorem 2 is a powerful result since nowhere did we have to express the functional form of $\mathbf{f}_{\boldsymbol{\lambda}}$ explicitly, thus making (14) suitable for a wide variety of algorithms. The important point is that $\mathbf{f}_{\boldsymbol{\lambda}}$ is treated as a black box, meaning that we only need access to the output of the operator, irrespective of how it is implemented. From a calculus point of view, it can be regarded as the stochastic definition of the divergence of a vector field in multiple dimensions where $\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y}+\varepsilon\mathbf{b}') - \mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})$ may be understood as the first-order (random) difference of $\mathbf{f}_{\boldsymbol{\lambda}}$. It may also be thought of as a generalization of a result due to Wahba [26], [27] and Girard [28] developed in the context of RGCV which is only applicable for "mildly" nonlinear problems, in the sense that $\mathbf{J}_{\mathbf{f}_{\boldsymbol{\lambda}}}(\mathbf{y}) \approx \mathbf{J}_{\mathbf{f}_{\boldsymbol{\lambda}}}(\mathbf{x})$. We discuss this further in Section V-C1.

Equation (14) (including the limit) forms the basis of our Monte-Carlo approach for computing SURE for a nonlinear $\mathbf{f}_{\boldsymbol{\lambda}}$. Since, in practice, the limit in (14) cannot be implemented due to finite machine precision, we propose the following approximation:

$$\frac{1}{N}\mathrm{div}_{\mathbf{y}}\{\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\} \approx \frac{1}{N\varepsilon}\mathbf{b}'^{\mathrm{T}}(\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y}+\varepsilon\mathbf{b}') - \mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})) \tag{17}$$

where the factor $(1)/(N)$ accounts for the averaging (of SURE) over all the pixels. The R.H.S. of (17) amounts to adding a small amount of noise (of variance ε^2) to $\mathbf{y}$ and evaluate $\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y}+\varepsilon\mathbf{b}')$. The difference $\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y}+\varepsilon\mathbf{b}') - \mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})$ is then used to obtain an estimate of the divergence. The schematics of implementing (17) is illustrated in Fig. 2. The validity of the approximation in (17) depends on how small ε can be made. In practice, we must select a ε small enough to mimic the limit, but still large enough so as to avoid round-off errors in $\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y}+\varepsilon\mathbf{b}')$. As demonstrated in Section V-B, the admissible range of ε covers several decades, so that the choice of this parameter is not critical.

We now give an algorithm for Monte-Carlo divergence estimation (and SURE) which is quite straightforward and easy to implement. It assumes that a "suitably" small ε has been selected and a zero-mean unit variance i.i.d. random vector $\mathbf{b}'$ has been generated.

Algorithm 2: Algorithm for estimating $(1)/(N)\mathrm{div}_{\mathbf{y}}\{\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\}$ and SURE ($\boldsymbol{\lambda}$) for a given $\boldsymbol{\lambda} = \boldsymbol{\lambda}_0$

1. For $\boldsymbol{\lambda} = \boldsymbol{\lambda}_0$, evaluate $\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})$.
2. Build $\mathbf{z} = \mathbf{y} + \varepsilon\mathbf{b}'$. Evaluate $\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{z})$ for $\boldsymbol{\lambda} = \boldsymbol{\lambda}_0$.
3. Compute $\mathrm{div} = (1)/(N\varepsilon)\,\mathbf{b}'^{\mathrm{T}}(\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{z}) - \mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y}))$ and $\mathrm{SURE}(\boldsymbol{\lambda}_0)$ using (6).

[1] A formal proof of this result is available at http://bigwww.epfl.ch/publications/ramani0803doc01.pdf

Algorithm 2 also uses only one realization of $\mathbf{b}'$ for the same reason given in Section IV-A: the law of large numbers is applicable to $(1)/(N)\mathbf{b}'^{\mathrm{T}}\mathbf{f}_{\boldsymbol{\lambda}k}(\mathbf{y}+\varepsilon\mathbf{b}')$ whenever $\mathrm{f}_{\boldsymbol{\lambda}k}(\mathbf{y}+\varepsilon\mathbf{b}')$ is "approximately" independent for different k. This assumption is quite valid in practice because $\mathbf{f}_{\boldsymbol{\lambda}}$ mostly performs "local" operations (for instance, finite-length wavelet filters and coordinate-wise thresholding are used in wavelet-based methods and finite-difference filters are used in TV denoising). We present experimental results in Sections V-C2–V-D that support this claim.

Another significant observation is that whenever $\mathbf{f}_{\boldsymbol{\lambda}}$ is linear, the two Monte-Carlo algorithms discussed in this work turn out to be rigorously equivalent. This is formally stated in the following proposition which is easily proven.

Proposition 2: Let $\mathbf{f}_{\boldsymbol{\lambda}}$ be linear as in (9) and $\mathbf{b}'$ be a zero-mean i.i.d. random vector with unit variance. Then, without the limit, the R.H.S. of (14) reduces to that of (13), independent of ε. ■

V. Validation and Comparison of Denoising Techniques

Now that we have a practical means of estimating $\mathrm{div}_{\mathbf{y}}\{\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\}$ for an arbitrary $\mathbf{f}_{\boldsymbol{\lambda}}$, we demonstrate the applicability of Monte-Carlo SURE for some popular denoising algorithms such as total-variation denoising (TVD) and redundant scale-dependent wavelet soft-thresholding (RSWST). Also included in the evaluation are orthonormal scale-dependent wavelet soft-thresholding and smoothing splines for which SURE takes an explicit form. For the variational methods (TVD and smoothing splines), the parameter $\boldsymbol{\lambda} = \lambda$ represents the regularization tradeoff, while for the wavelet-based methods, $\boldsymbol{\lambda}$ controls the scale-dependent thresholds. In the forthcoming sections, we first describe each algorithm along with its associated characteristics. We then discuss numerical issues related to choice of ε to be used in Algorithm 2. Finally, we present experimental results that validate our arguments.

A. Description of Denoising Methods

1) Orthonormal Scale-Dependent Wavelet Soft-Thresholding (OSWST): If $\mathbf{W}$ is the matrix corresponding to an orthonormal wavelet transform, the OSWST denoised signal is given by $\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y}) = \mathbf{W}^{\mathrm{T}}\tilde{\mathbf{c}}$, where

$$\tilde{\mathbf{c}} = \arg\min_{\mathbf{c}} \underbrace{\left\{ \|\mathbf{y} - \mathbf{W}^{\mathrm{T}}\mathbf{c}\|^2 + \sum_{i,k} \lambda_{i,s,q} \, |c_k^i|^q \right\}}_{\mathcal{J}_{\mathrm{W}}\{\mathbf{y},\mathbf{c}\}}. \tag{18}$$

The second term in the R.H.S. of the above equation is equivalent to the Besov norm of the corresponding continuously defined signal estimate [39]. The quantity c_k^i is the kth wavelet coefficient in the ith sub-vector of $\mathbf{c}$ (corresponding to the ith sub-band) and $\lambda_{i,s,q} = 2^{-iq(s+(d)/(2)-(d)/(q))}\lambda$ is the scale-dependent regularization parameter for s, $\lambda \in \mathbb{R}^+$; the dimension of the data is d, while q corresponds to the ℓ_q norm of the coefficient vector. For our experiments, we set $d = 2$ and $q = 1$ (for image denoising with ℓ_1 constraint on the wavelet coefficients), which yields the scale-dependent regularization parameter

$$\lambda_{i,s} = 2^{-i(s-1)}\lambda. \tag{19}$$

The advantage of selecting an orthogonal transform is that it decouples $\mathcal{J}_{\mathrm{W}}$ so that (18) is equivalent to independently minimizing scalar cost functions on a coefficient-by-coefficient basis. The minimization of scalar cost corresponding to $\tilde{c}_k^i$ is then simply achieved by a soft-thresholding operation [39] with the threshold $(\lambda_{i,s})/(2)$ so that

$$\begin{aligned} \tilde{c}_k^i &= \mathcal{T}_{\lambda_{i,s}}\left(c_k^i\right) \\ &= \begin{cases} c_k^i - \frac{\lambda_{i,s}}{2}\,\mathrm{sign}\left(c_k^i\right), & \text{if } |c_k^i| > \frac{\lambda_{i,s}}{2} \\ 0, & \text{if } |c_k^i| \le \frac{\lambda_{i,s}}{2} \end{cases} \end{aligned} \tag{20}$$

where $\mathbf{c}_k^i$ is the kth wavelet coefficient in the ith sub-band of the wavelet transform $\mathbf{c} = \mathbf{W}\mathbf{y}$. Due to the orthonormality of $\mathbf{W}$ the MSE (and, hence, SURE) is invariant under the transform (Parseval equivalence). Therefore, $\mathbf{c}$ replaces $\mathbf{y}$, while $\mathcal{T}_{\lambda_{i,s}}$ replaces $\mathbf{f}_{\boldsymbol{\lambda}}$ in (6). The required divergence is then simply computed to be $\sigma^2 \sum_{i,k} \mathbb{I}_{\mathcal{A}}(c_k^i)$, where $\mathcal{A} = \{c_k^i : |c_k^i| > (\lambda_{i,s})/(2)\ \forall\, i,k\}$ and $\mathbb{I}\{\cdot\}$ is the indicator function.

The OSWST is akin to the *SureShrink* algorithm of Donoho *et al.* [17] in that they both apply soft-thresholding in an orthonormal (wavelet) transform domain. However, the two methods significantly differ from each other in the way they select the threshold levels: while *SureShrink* assigns a threshold value to each sub-band by independent sub-band minimization of SURE, OSWST optimizes the threshold parameters (λ, s) (that characterize the sub-band dependent threshold value in (19)) by minimization of SURE computed over all the sub-bands (entire wavelet decomposition).

2) Redundant Scale-Dependent Wavelet Soft-Thresholding (RSWST): Redundant discrete wavelet transforms are over-complete representations that are advantageous for denoising, mainly due to their better shift-invariant properties [40]–[42]. We consider the undecimated wavelet transform (UWT) with an orthonormal filter pair in the redundant paradigm (tight-frame). Our denoising function is again the scale dependent soft-thresholding operator $\mathcal{T}_{\lambda_{i,s}}$ but now applied on the UWT coefficients. For $s = 1$ in (19), $\lambda_{i,s} = \lambda$ yields the same threshold level for all sub-bands i in which case both OSWST and RSWST perform universal soft-thresholding of the corresponding wavelet coefficients. However, unlike OSWST, there is no cost function associated with RSWST. Moreover, as shown in [20], Parseval's equivalence is no longer valid in the redundant wavelet domain which forces us to evaluate SURE in the signal domain.

Writing $\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y}) = \mathbf{W}^{\mathrm{T}}\boldsymbol{T}(\mathbf{W}\mathbf{y})$, where $\mathbf{W}$ is a UWT matrix and $\boldsymbol{T}$ the vector containing the soft-thresholding operators $\mathcal{T}_{\lambda_{i,s}}$ [see (20)], it is immediately clear that evaluating $\mathrm{div}_{\mathbf{y}}\{\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\}$ is arduous because the output of $\boldsymbol{T}$ depends on $\mathbf{W}\mathbf{y}$ thus demanding explicit access to each element of $\mathbf{W}$. However, since the soft-thresholding operator is continuous and weakly-differentiable [18], RSWST (and OSWST included) satisfies the weaker hypotheses of Theorem 2 and, therefore, qualifies for Monte-Carlo estimation procedure described in Section IV-B. In fact, RSWST constitutes a good demonstration example for illustrating the signal-domain computation of SURE using Algorithm 2 to perform a combined optimization of the two threshold parameters $\boldsymbol{\lambda} = (\lambda, s)$.

3) Total-Variation Denoising (TVD): While wavelet-based denoising forms an active research area in its own right, other denoising procedures that have flourished in the literature, include variational and PDE based methods of which the most popular is TV denoising [43]. The idea behind TVD is to minimize the total-variation of an image that is constrained to be "close" to the given noisy data. The problem has been formulated in both continuous and discrete domains [43], [44]. The solution is either found by evolving a PDE derived from the Euler–Lagrange equation or by performing some kind of iterative optimization [e.g., bounded optimization using majorization-minimization (MM) [45] or half-quadratic [46] optimization].

Here, we consider the discrete domain formulation of Figueiredo *et al.* [44] where the TV denoised image is obtained by minimizing the cost functional

$$\mathcal{J}_{\mathrm{TV}}(\mathbf{y},\mathbf{u}) = \|\mathbf{y}-\mathbf{u}\|^2 + \lambda\,\mathrm{TV}(\mathbf{u}) \tag{21}$$

where $\mathrm{TV}(\mathbf{u}) = \sum_k \sqrt{(\mathbf{D}_h\mathbf{u})[k]^2 + (\mathbf{D}_v\mathbf{u})[k]^2}$ is the discrete 2-D total-variation norm and $\mathbf{D}_h$ and $\mathbf{D}_v$ are matrices corresponding to the first order finite difference in the horizontal and vertical directions, respectively. $\mathcal{J}_{\mathrm{TV}}$ is convex and can be minimized using an iterative MM algorithm [44]. Then, starting from the update equation, it can be established in a straightforward (but tedious) manner that $\mathbf{f}_{\boldsymbol{\lambda}}$ for TVD admits at least a second-order Taylor expansion.[2] TVD is a typical example where SURE cannot be evaluated analytically and while our Monte-Carlo method circumvents the difficulty.

4) Smoothing Splines: The smoothing splines problem corresponds to reconstructing a continuously-defined function from an infinitely long sequence ($N \to \infty$) of noisy data on a uniform grid. It is generally formulated in the shift-invariant framework [30]–[33] where the B-spline coefficients are obtained by linear (digital) filtering of the noisy data.

We will slightly digress from the vector notation to accurately formulate what we said in the paragraph above. Let $\{y[\mathbf{k}]\}_{\mathbf{k}\in\mathbb{Z}^d}$ represent the infinite sequence of noise-corrupted input in d dimensions. The smoothing spline algorithm is usually described by a generator $\varphi : \mathbb{R}^d \to \mathbb{R}$ which specifies the approximation space (e.g., polyharmonic spline) and a digital correction filter $h_{\boldsymbol{\lambda}}$. In the denoising scenario, the denoised output is obtained by re-sampling the smoothing spline on the grid which yields an estimate of the form

$$\begin{aligned}\mathbf{f}_{\boldsymbol{\lambda}}(y)[\mathbf{k}] &= \sum_{\mathbf{m}\in\mathbb{Z}^d} (y * h_{\boldsymbol{\lambda}})[\mathbf{m}]\,\varphi(\mathbf{k}-\mathbf{m}) \\ &= (y * h_\lambda * b)[\mathbf{k}]\end{aligned} \tag{22}$$

where $\mathbf{f}_{\boldsymbol{\lambda}}(y)[\mathbf{k}]$ is the $\mathbf{k}$th component of the infinite-dimensional vector $\mathbf{f}_{\boldsymbol{\lambda}}(y)$ and $b[\mathbf{k}] = \varphi(\mathbf{x})|_{\mathbf{x}=\mathbf{k}\in\mathbb{Z}^d}$. The required divergence is $\mathrm{div}_{\mathbf{y}}\{\mathbf{f}_{\boldsymbol{\lambda}}(y)\}$ whose $\mathbf{k}$th component is given by

$$\frac{\partial \mathbf{f}_{\boldsymbol{\lambda}}(y)[\mathbf{k}]}{\partial y[\mathbf{k}]} = (h_{\boldsymbol{\lambda}} * b)[\mathbf{0}]. \tag{23}$$

[2]The derivation of this result can be found at http://bigwww.epfl.ch/publications/ramani0803doc01.pdf

It is independent of $\mathbf{k}$ and can be computed in the Fourier domain as

$$(h_{\boldsymbol{\lambda}} * b)[\mathbf{0}] = \frac{1}{(2\pi)^d}\int_{\boldsymbol{\omega}\in[0,\,2\pi)^d} \underbrace{H_{\boldsymbol{\lambda}}(e^{j\boldsymbol{\omega}})\left(\sum_{\mathbf{k}\in\mathbb{Z}^d}\hat{\varphi}(\boldsymbol{\omega}+2\pi\mathbf{k})\right)}_{F_{\boldsymbol{\lambda}}(e^{j\boldsymbol{\omega}})}\,d\boldsymbol{\omega}$$

where $H_{\boldsymbol{\lambda}}(e^{j\boldsymbol{\omega}})$ is the frequency response of $h_{\boldsymbol{\lambda}}$ and $\hat{\varphi}$ is the Fourier transform of φ.

In the finite-dimensional case, the smoothing spline denoised output can be obtained using (9) where $\mathbf{F}_{\boldsymbol{\lambda}}$ is the block-circulant matrix formed from the filter taps $(h_{\boldsymbol{\lambda}} * b)[\mathbf{k}]$ and is diagonalized by the fast Fourier transform (FFT) matrix. Its eigenvalues are nothing but the samples of the frequency response $F_{\boldsymbol{\lambda}}(e^{j\boldsymbol{\omega}})$ whose sum yields the desired trace.

B. Range of Validity of the Proposed Monte-Carlo SURE

The two main conditions for Algorithm 2 to work are that $\mathbf{f}_{\boldsymbol{\lambda}}$ satisfies the hypotheses of Theorem 2 and ε is "small". Ideally, we would like to let ε tend towards zero in (17) as dictated by (14), but this cannot be realized exactly in practice due to finite machine precision. When ε is too small, numerical round-off errors become more prominent because $\mathbf{f}_{\boldsymbol{\lambda}}$ becomes insensitive to changes in ε. In effect, this phenomenon fixes a lower bound for ε which may vary depending on the sensitivity of $\mathbf{f}_{\boldsymbol{\lambda}}$. To elucidate this, we selected the following nonlinear algorithms: TVD and RSWST with threshold value $(\lambda)/(2)$ (which satisfy at least one of the hypotheses of Theorem 2) and found, based on numerical experiments with JAVA that $\varepsilon \geq 10^{-12}$ was admissible for these algorithms. We then applied Algorithm 2 with Gaussian $\mathbf{b}'$ for each of these methods for different values of ε and a wide range of λ for the Boats test image with input SNR 4 dB.

We observed that when ε was decreased from $\varepsilon = 1$ down to 10^{-12}, Algorithm 2 yielded SURE values which not only captured the trend of the true MSE over a wide range of λ but also yielded very good estimates of the optimal λ for the TVD and RSWST methods, in agreement with Theorem 2. We illustrate this in Fig. 3 for the cases of $\varepsilon = 0.1$ and $\varepsilon = 0.01$ for TVD and RSWST where the corresponding curves nearly overlap and are also close to the true MSE curve over the entire range of λ. At the other end, as soon as $\varepsilon \gtrsim 2$, we started to observe significant bias (cf. uppermost curves in Fig. 3 corresponding to $\varepsilon = 10$) which indicates that large ε is not desirable for nonlinear problems. We, therefore, conclude that whenever the assumptions of Theorem 2 are valid, the proposed estimation procedure is quite robust with respect to ε (when $\varepsilon \to 0$) and it yields meaningful results when ε is made "small."

Next, to investigate the relevance of the underlying differentiability hypotheses in Theorem 2, we applied Algorithm 2 to RSWHT which performs hard-thresholding with the threshold value $(\lambda)/(2)$. Since the hard-thresholding operator is neither continuous nor weakly-differentiable [47], RSWHT violates the hypotheses of Theorem 2. Numerically, this is reflected in the increasing instability of the SURE curves as ε decreases in Fig. 4. In this case, violating the hypotheses of Theorem 2 leads to a

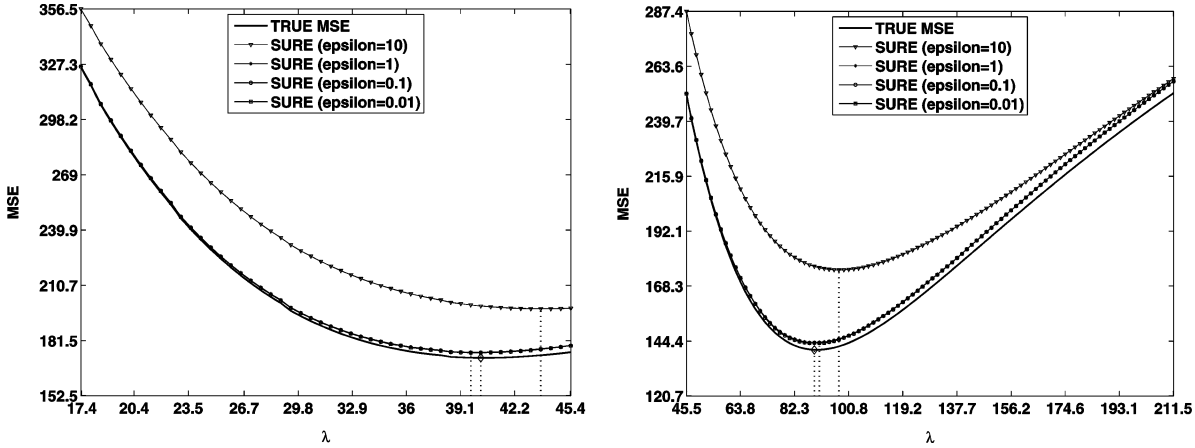

Fig. 3. Plots of MSE (λ) and (Monte-Carlo) SURE (λ) for different ε: TVD (left); Haar-RSWST with threshold value $(\lambda)/(2)$ (right); Noisy Boats image with SNR $= 4$ dB; $\sigma = 29.45$.

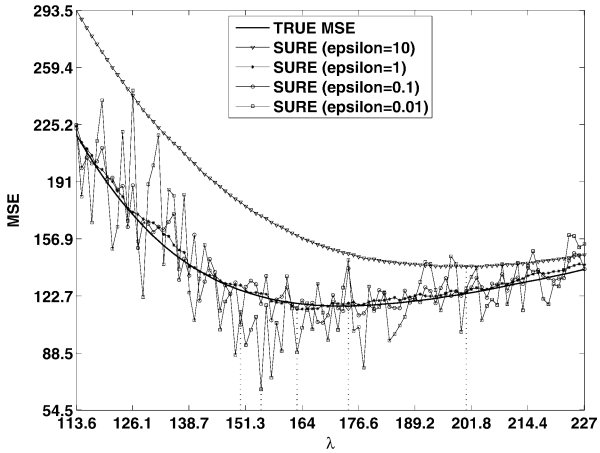

Fig. 4. Plots of MSE (λ) and (Monte-Carlo) SURE (λ) for different ε: Haar-RSWHT with threshold value $(\lambda)/(2)$; Noisy Boats image with SNR $= 4$ dB; $\sigma = 29.45$.

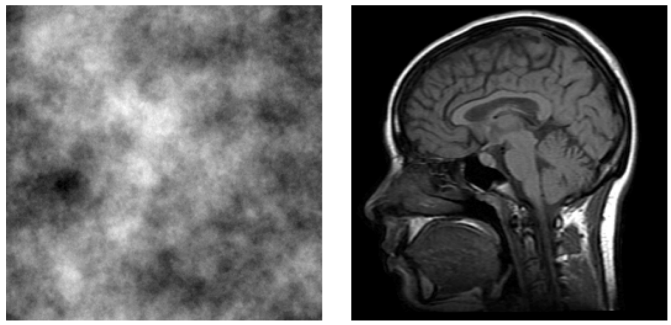

Fig. 5. Specific noise-free images considered in this paper apart from other standard test images.

variance of Monte-Carlo SURE that increases without bound with decreasing ε.

It must be noted that the hard-thresholding function is quite an extreme case and has been considered here purely to illustrate the sharpness of the hypotheses of Theorem 2 to certify whether or not a denoising algorithm is suitable for the proposed Monte-Carlo SURE. Fortunately for us, most common algorithms encountered in practice satisfy the required differentiability hypothesis and can be optimized with Algorithm 2 as demonstrated next.

C. Results With One-Parameter Optimization

We now present numerical results for SURE-based optimization of a single parameter (only λ) for the methods discussed in Section V-A. In doing this, we exemplify the use of SURE, but do not contend with state-of-the-art denoising methods. For our experiments, we consider different categories of test images including a medical image (MRI 256×256), a stochastic image [a realization of fractional Brownian motion (fBm) with Hurst exponent 0.5 on a uniform grid of size 256×256, see Fig. 5], a tomography phantom (Shepp–Logan phantom of size 256×256) together with three standard natural images: Barbara (512×512), Boats (512×512), and Peppers (256×256). To test the effectiveness of smoothing splines for denoising of stochastic signals, we implement the polyharmonic smoothing spline (PSS) of degree equal to 1 which is known to be the optimal estimator for the considered fBm image [33]. We choose the Haar wavelet transform for the wavelet based methods to match the wavelet filter with the first-order finite difference filter employed in TVD. We used $J = 4$ levels of decomposition in all cases and did not perform any thresholding on the coarse-scale projection of the signal.

The performance of the methods is quantified by the SNR of the output $\mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})$, which is computed as

$$\mathrm{SNR} = 10 \log_{10} \left(\frac{\|\mathbf{x}\|^2}{\|\mathbf{x} - \mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\|^2} \right). \tag{24}$$

All SNR values reported in this paper were obtained by averaging over three independent simulations. We consider images corrupted by white Gaussian noise whose standard deviation σ is known (it can be estimated reliably in practice using the median estimator of Donoho *et al.* [17]). In all the experiments, the value of σ is set to achieve the desired input SNR computed by replacing $\|\mathbf{x} - \mathbf{f}_{\boldsymbol{\lambda}}(\mathbf{y})\|^2$ with $N\sigma^2$ in (24). Besides, in the implementation of all the methods, periodic boundary conditions were

TABLE I
COMPARISON OF GCV AND SURE IN TERMS OF SNR IMPROVEMENT

	Input SNR (dB)	4	8	12	16	20	4	8	12	16	20
Method	Measure	Boats					MRI				
PSS (Degree 1)	Oracle	11.83	13.69	15.81	18.27	21.20	12.40	14.63	17.02	19.59	22.35
	SURE	11.83	13.69	15.81	18.27	21.20	12.40	14.63	17.02	19.59	22.35
	GCV	11.76	13.36	14.80	16.04	20.02	11.98	13.57	14.77	16.07	20.04
RSWST ($s = 1$)	Oracle	11.87	14.07	16.49	19.07	21.91	12.20	14.64	17.26	20.08	23.08
	SURE	11.87	14.06	16.49	19.07	21.90	12.19	14.64	17.26	20.07	23.08
	$\mathrm{RGCV}_{\mathrm{NL}}$	9.42	11.45	13.05	16.56	20.03	9.63	12.04	13.90	17.82	21.18
	$\mathrm{RGCV}^{\star}_{\mathrm{NL}}$	11.65	13.97	15.19	18.83	20.60	12.11	14.44	16.97	19.99	22.87
Method	Measure	Peppers					Shepp-Logan				
PSS (Degree 1)	Oracle	10.74	12.47	14.70	17.44	20.68	9.91	11.79	14.13	17.06	20.45
	SURE	10.74	12.47	14.70	17.44	20.68	9.91	11.79	14.13	17.06	20.45
	GCV	10.74	12.42	12.10	16.04	20.01	9.88	11.78	14.12	17.00	20.28
RSWST ($s = 1$)	Oracle	12.05	14.57	17.28	20.04	22.88	13.98	17.59	21.28	25.02	28.82
	SURE	12.04	14.56	17.28	20.04	22.88	13.98	17.58	21.26	25.00	28.81
	$\mathrm{RGCV}_{\mathrm{NL}}$	9.34	11.96	13.93	17.86	20.86	10.84	14.51	17.19	22.04	25.93
	$\mathrm{RGCV}^{\star}_{\mathrm{NL}}$	11.94	14.27	16.18	19.98	22.82	13.66	16.90	19.89	24.27	28.32

used when required. For PSS and OSWST, SURE was computed analytically, while for TVD and RSWST, the proposed Monte-Carlo method (Algorithm 2) was used with zero-mean i.i.d. Gaussian random vectors of standard deviation $\varepsilon = 0.1$.

1) Comparison With Other Performance Measures: Here, we compare the performances of SURE and generalized cross validation for a linear (PSS method) and a nonlinear (RSWST ($s = 1$)) algorithm in terms of SNR improvement. The GCV is computed explicitly for the PSS method, while, for RSWST, we consider the Monte-Carlo version (for nonlinear algorithms) proposed by Girard [28] which we denote $\mathrm{RGCV}_{\mathrm{NL}}$. Thus, we write GCV [24] and $\mathrm{RGCV}_{\mathrm{NL}}$ [28] as

$$\mathrm{GCV}(\lambda) = \frac{N^{-1}\|\mathbf{y} - \mathbf{F}_\lambda \mathbf{y}\|^2}{(1 - N^{-1}\mathrm{trace}\{\mathbf{F}_\lambda\})^2} \tag{25}$$

$$\mathrm{RGCV}_{\mathrm{NL}}(\lambda) = \frac{N^{-1}\|\mathbf{y} - \mathbf{f}_\lambda(\mathbf{y})\|^2}{\left(1 - N^{-1}\varepsilon^{-1}\mathbf{b}'^{\mathrm{T}}[\mathbf{f}_\lambda(\mathbf{y} + \varepsilon\mathbf{b}') - \mathbf{f}_\lambda(\mathbf{y})]\right)^2} \tag{26}$$

where $\varepsilon = 0.9\sigma$ is used in (26) as recommended in [28]. The output SNR obtained by adjusting λ based on SURE and generalized cross validation (GCV and $\mathrm{RGCV}_{\mathrm{NL}}$) is tabulated for various input noise levels and test images in Table I.

As seen from the table, for the PSS method, the performance of GCV becomes steadily poorer with decreasing noise level. This may be due to the fact that GCV does only perform optimally under special conditions (cf. Proposition 3.1 in [25]) which are probably not fulfilled in the present experiments. As for $\mathrm{RGCV}_{\mathrm{NL}}$, it was observed that the selected λ was far from the optimum value in all cases: this can be attributed to the bias originating from the recommended value of ε and the fact that RSWST does not probably satisfy the "mild" nonlinearity assumption. As a result, the performance of $\mathrm{RGCV}_{\mathrm{NL}}$ is poor at all noise levels.

Following the philosophy underlying (14) and the argumentation in Section V-B, we, therefore, decided to inspect another version of $\mathrm{RGCV}_{\mathrm{NL}}$, denoted by $\mathrm{RGCV}^{\star}_{\mathrm{NL}}$, which utilized a small value: $\varepsilon = 0.1$. It is observed that $\varepsilon = 0.1$ dramatically improves the performance as reflected in the output SNR values corresponding to $\mathrm{RGCV}^{\star}_{\mathrm{NL}}$: this demonstrates the validity of the proposed Monte-Carlo procedure for estimating the divergence for algorithms with "arbitrary" nonlinearities. However, it should be noted that the performance of $\mathrm{RGCV}^{\star}_{\mathrm{NL}}$ is still not on par with SURE, which consistently imitates the oracle for both the methods and for all noise levels and considered test images. This indicates that GCV-like measures, though having the advantage of not requiring σ^2, may not always yield optimal performance for all denoising algorithms.

2) MSE-SURE Comparison: A series of relevant graphs ($\mathrm{SURE}(\lambda), \mathrm{MSE}(\lambda)$ versus λ) for four denoising methods are shown in Figs. 6 and 7. It is observed that SURE follows the true MSE curve remarkably well in all the cases thereby leading to accurate estimates of the optimal λ. We observed the same trend for all test images and input SNRs which confirms the consistency of our method. The agreement is somewhat better in the case of larger images (Boats, Barbara) as compared to the Peppers image which is probably due to the fact that we have four times more pixels to estimate the MSE (law of large numbers).

These results demonstrate the validity of the approximation in (17). The RSWST method is a borderline case for which the formula (14) is only true in the weak sense because the second derivative of the soft-thresholding operator is not well-defined for the two critical values $\pm(\lambda)/(2)$. Yet, Algorithm 2 still performs well in accordance with the second part of Theorem 2.

It should also be noted that this type of extensive estimation over a wide range of λ (as shown in Figs. 6 and 7) has been done purely for the purpose of illustration. In practice, we can rely on bracketing methods (golden-mean search) which do not use any derivative information in order to find the minimum of SURE in a much smaller number of steps (typically 10 steps).

3) Visual Comparison: To highlight the different characteristics of the denoising methods it is best to compare the results visually. Fig. 8 shows the denoised outputs of four algorithms with optimized parameters. The smoothing spline estimator, as its name suggests, attempts to smooth the noisy fluctuations during the denoising process. But in doing so, it also smoothes the underlying image leading to smudged edges [as seen in Fig. 8(c)], which is the main disadvantage of this approach.

The Haar-OSWST ($s = 1$) preserves some edge information but produces a blocky output because small detail coefficients are set to zero by the univariate soft-thresholding op-

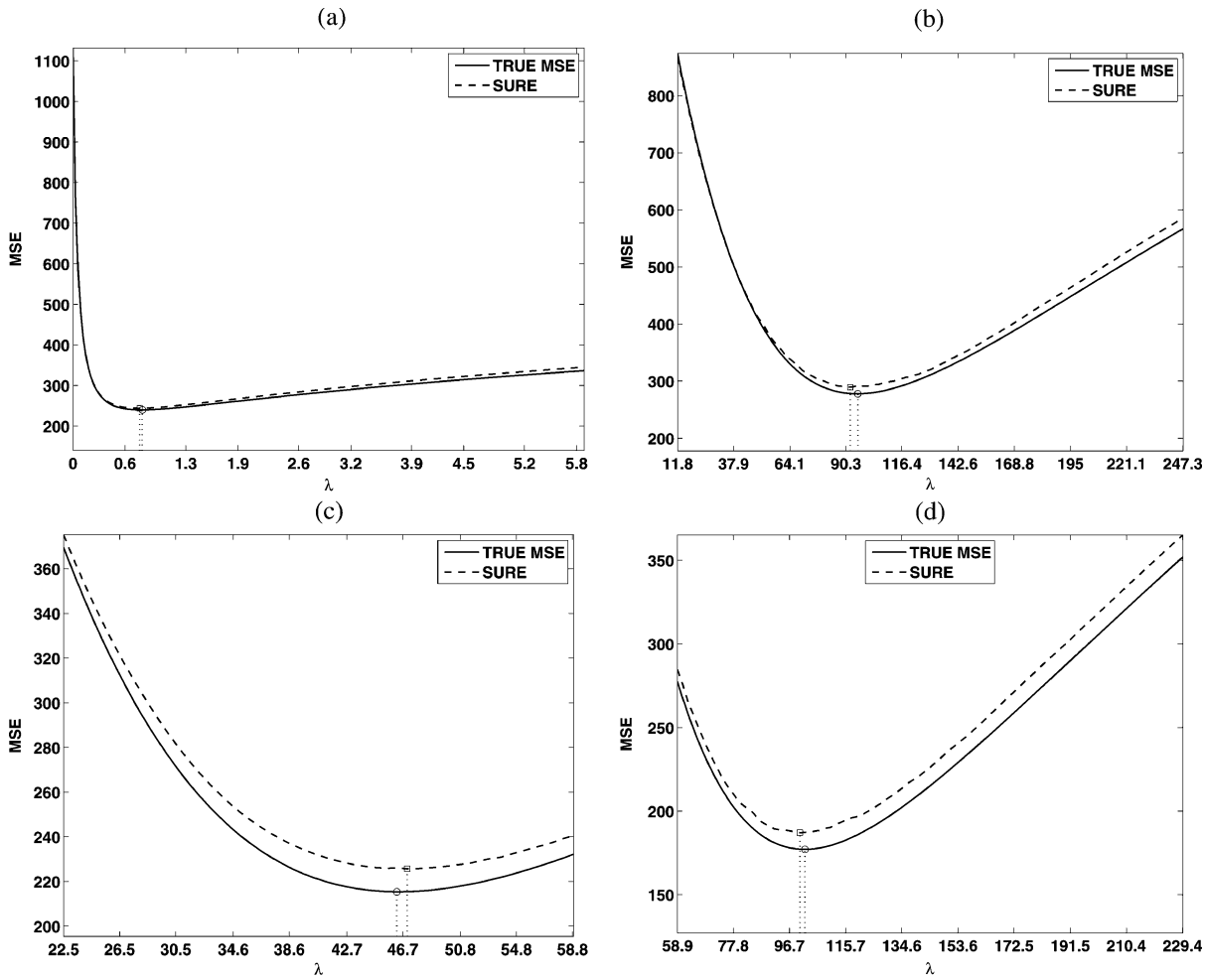

Fig. 6. MSE (λ) and SURE(λ) for all considered methods (Noisy Peppers image with SNR $= 4$ dB, $\sigma = 33.54$), (a) PSS (Degree $= 1$), (b) Haar-OSWST ($s = 1$), (c) TVD ($\epsilon = 0.1$), (d) Haar-RSWST ($s = 1, \epsilon = 0.1$).

erator. There is a loss of image details and the reconstructed output exhibits artifacts corresponding to the footprints of the basis function (Haar wavelet). The Haar wavelet is at the low end of what can be achieved with an orthonormal wavelet transform; the use of a wavelet with better regularization properties (symlets, higher order spline wavelets, etc.) yields better results—typically +0.5 dB additional gain (results not shown).

The TV denoised image appears significantly better than the earlier two. Yet, although the edges are preserved as per the TV constraint, the output exhibits some artificial blockiness due to the fact that the algorithm tends to favour piecewise constant solutions.

The Haar-RSWST ($s = 1$) yields the best visual output, which correlates with the higher SNR value (11.90 dB). This can be attributed to the redundant nature of the underlying transform. Interestingly enough, the result is not penalized by the lower order of the Haar transform (piecewise-constant approximation), in fact, it is quite the contrary (as was also noticed in [20]). This is in contrast with the nonredundant case where higher order wavelets yield better results, but nothing that comes close to the result in Fig. 8(f).

4) Computational Cost: Two main aspects of any denoising algorithm are the associated computational cost and the yielded SNR improvement. In general, these two aspects are conflicting in nature and the user must strike a good balance between them. In terms of computational efficiency, the four methods can be ranked as follows.

i) The Haar-OSWST method ($J = 4$ levels), which requires of the order of $2 \times 4N$ operations, while it uses the same amount of storage as the image itself.

ii) Polyharmonic smoothing splines; these are implemented efficiently using the FFT and, therefore, require $O(N \log_2 N) + N$ operations while storage-wise, it is equivalent to the Haar-OSWST method.

iii) The Haar-RSWST method; it is implemented using the *algorithm à trous* [41] which, for $J = 4$, requires a total of $13 \times 2 \times 4N$ computations. It should be noted that the performance improvement yielded by the redundancy of the transform is at the cost of requiring $13N$ storage locations which is probably one potential downside of this method.

iv) TVD; the MM algorithm of [44] required an average of 13 main iterations. At any given iteration, the method uses few-N locations (typically $<4N$) for storing intermediate iteration variables. Additionally, for each main iteration, we performed 20 conjugate-gradient iterations

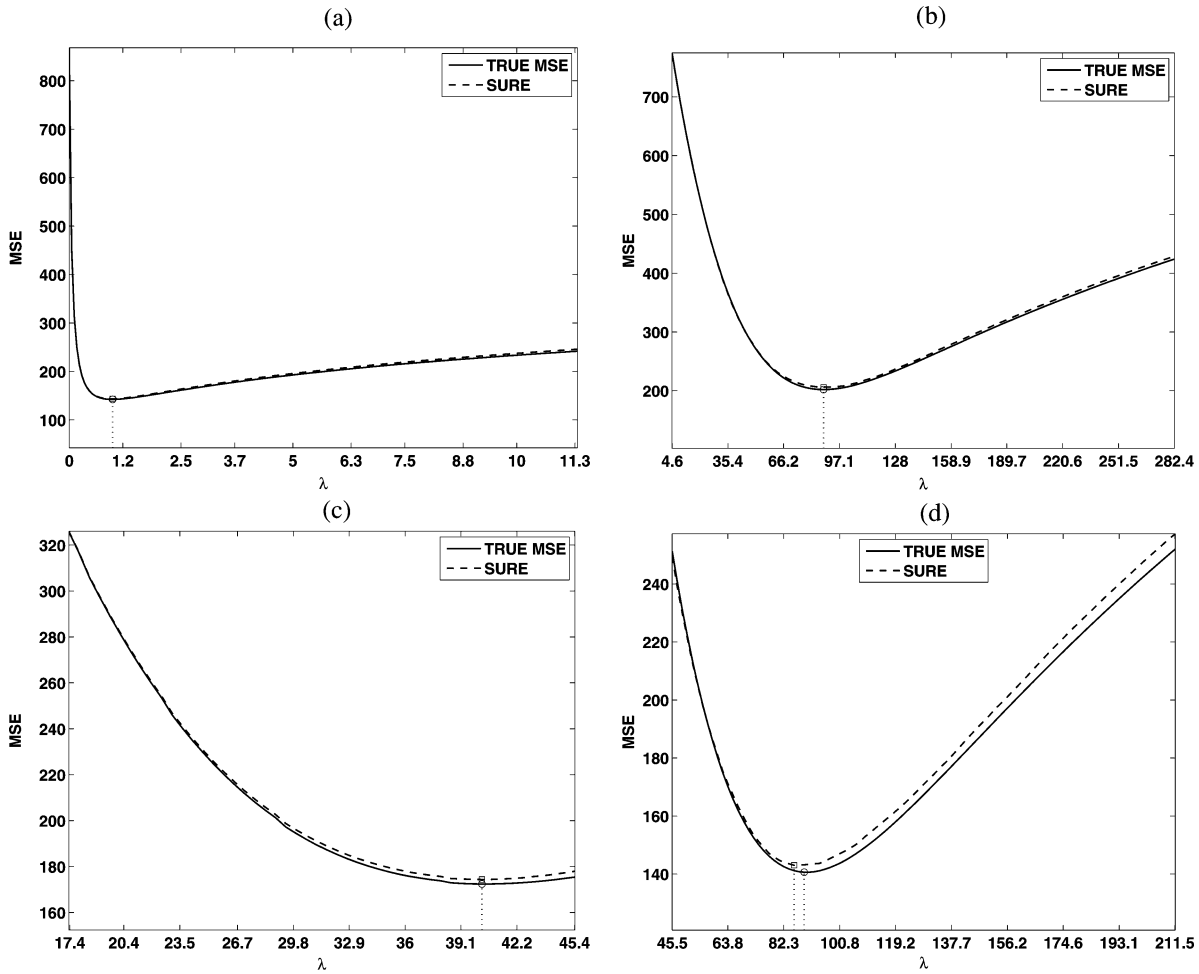

Fig. 7. MSE(λ) and SURE (λ) for all considered methods (Noisy Boats image with input SNR $= 4$ dB, $\sigma = 29.45$)). (a) PSS (Degree $= 1$), (b) Haar-OSWST ($s = 1$), (c) TVD ($\epsilon = 0.1$), (d) Haar-RSWST ($s = 1, \epsilon = 0.1$).

to solve an associated linear system. This leads to a total of 260 N operations to obtain a single denoised signal estimate implying that TVD is the costliest of all the considered methods.

5) SNR Improvement: We now make a quantitative comparison of the methods in terms of SNR improvement. For the sake of comparison, the SNR is computed for outputs obtained by setting λ based on both the true MSE and SURE. This is tabulated in Table II where the first value in each cell gives the SNR obtained by choosing λ based on the true MSE (oracle SNR), while the second corresponds to that obtained by Monte-Carlo SURE optimization. The maximum of the SNRs with respect to all the methods is indicated in bold-face font for each image and noise variance. Several observations are in order:

- The first and the most important one for this paper is that the SNR obtained based on the true MSE and SURE are either equal or different only in the second decimal place for all tested cases. This indicates the reliability and robustness of our Monte-Carlo SURE optimization procedure.
- Haar-OSWST ($s = 1$) performs poorly, especially at high noise levels. This is due to the inflexible nature of the soft-thresholding operator and blocky-reconstruction of the Haar wavelet. However, as noted earlier, one may be able to boost the performance slightly by using a higher order wavelet (typically $+0.5$ dB additional gain).
- The linear smoothing spline technique is among the least effective method for natural image denoising. It is seemingly better than Haar-OSWST ($s = 1$) at high noise levels for almost all images due to the fact that it smoothes the noisy image thereby strongly reducing the harsh effect of noisy fluctuations. But, it also smoothes the underlying signal making it the least-preferred method for images with rich texture (for instance, the Barbara image).
  However, the polyharmonic smoothing spline of degree 1 outperforms all the other methods for the fBm image, which is in agreement with the theory. This also strengthens the fact that smoothing splines are ideal whenever the underlying image fits the statistical model. A similar behaviour is observed for the MRI image which may be due to the fact that MRI images are mostly fractal-like [48] and their power spectrum can be well approximated by the $1/\|\boldsymbol{\omega}\|^{\alpha}$ spectral law [49].
- As expected, the use of redundant transform improves the denoising quality compared to Haar-OSWST ($s = 1$). The Haar-RSWST (with $s = 1$) method provides a gain of more than 2 dB compared to Haar-OSWST ($s = 1$) at large levels of noise. Notably, it is also the best method for all the

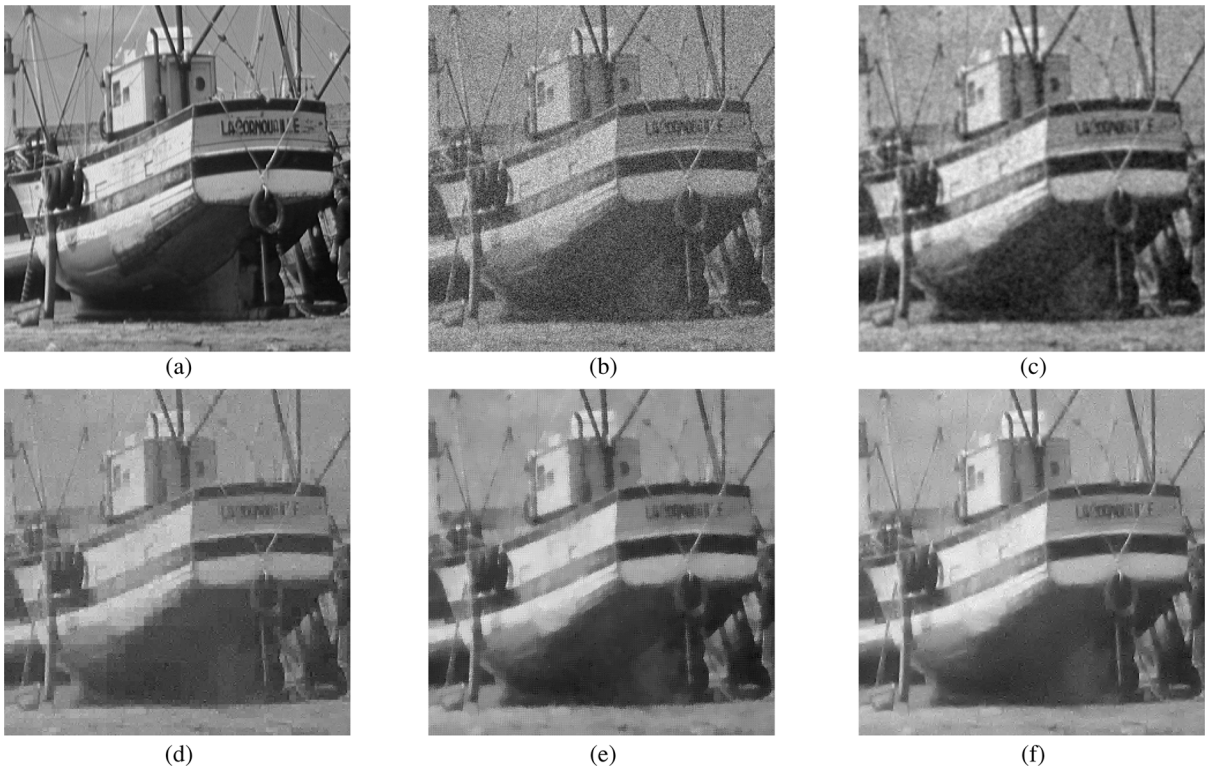

Fig. 8. Visual comparison of SURE-optimized denoising results for the Boats image (zoomed): (a) Noise-free image; (b) noisy observations ($\sigma = 29.45$; SNR $=$ 4 dB), (c) polyharmonic smoothing spline (Degree 1) result (SNR $= 11.84$ dB) (d) Haar-OSWST ($s = 1$) result (SNR $= 10.33$ dB); (e) TVD result (SNR $=$ 11.02 dB); (f) Haar-RSWST result ($s = 1$, SNR $= 11.90$ dB).

images with the exception of fBm and the Shepp–Logan phantom.

- TVD performs better than PSS and Haar-OSWST ($s = 1$) (and even (λ, s)-optimized Haar-OSWST, see the following subsection for details), whenever the images are smooth without strong textures (for instance the Peppers image and the Shepp–Logan phantom). This shows that TVD is competitive or even better than classical wavelet denoising methods [44] for images that fall well within the piecewise-constant category. The Shepp–Logan phantom is noteworthy in this context as it is a good example of a piecewise constant image. Unsurprisingly, TVD performs better than all the considered methods for this particular image, as indicated in Table II.

  In the presence of rich texture (the Barbara image), however, TVD performs worse than all wavelet based methods, which is quite expected because the TV prior is not well-suited for such images. In fact, any texture is considered part of the noise and is annihilated by TVD.

To conclude, we infer that of the considered methods, some are better suited than others for certain type of images: while overall Haar-RSWST yields the best results for natural images, smoothing splines are well adapted to fractal-like processes and TVD does best for piecewise-constant images.

D. Results With Multiparameter Optimization

So far we have only provided results for SURE-based one-parameter optimization. However, there is no major difficulty in applying our method for multiparameter optimization as well. The brute force approach would be to perform an exhaustive search in multiple dimensions to find the best parameter values that minimize SURE. A better way is to perform the search by applying derivative-free optimization. The Powell-Brent algorithm, which uses bracketing and parabolic interpolation for line-search and takes about $n(n+1)/2$ iterations to converge for n set of parameters, is well-suited for our problem as long as the number of parameters stays reasonably small (typically $n < 10$).

Here, we test the concept with the optimization of $\boldsymbol{\lambda} = (\lambda,\ s)$ for the PSS, Haar-OSWST and Haar-RSWST methods. For the PSS method, s matches the order of the spline to the Hurst exponent of the underlying noise-free signal. This fact has been applied in [33] where the optimal (λ, s) is obtained by fitting a fractal-like model to the power spectrum of the noisy image. However, in our approach, this is not required as λ and s are optimized together using SURE. For the wavelet methods, adjusting s changes the threshold value in each sub-band according to (19) and our understanding is that this yields better denoising performance than universal soft-thresholding. In all our experiments, we observed that the 2-D Powell optimization of the respective methods took no more than four iterations at various noise levels for all the test images. The results are tabulated in Table III.

With PSS, the combined optimization does not yield any significant improvement for the fBm since a degree 1 spline is theoretically the best in the MSE sense (Wiener solution). As expected PSS still performs the best of all the methods for the fBm image. The improvement for Boats, Barbara, Peppers and the

TABLE II
PERFORMANCE OF CONSIDERED METHODS IN TERMS OF SNR†

Image	Input SNR (dB)	4	8	12	16	20
Boats	PSS (Degree 1)	(11.83, 11.83)	(13.69, 13.69)	(15.81, 15.81)	(18.27, 18.27)	(21.20, 21.20)
	Haar-OSWST ($s = 1$)	(10.33, 10.32)	(12.63, 12.63)	(15.22, 15.22)	(18.09, 18.08)	(21.23, 21.23)
	Haar-RSWST ($s = 1$)	(**11.87**, **11.87**)	(**14.07**, **14.06**)	(**16.49**, **16.49**)	(**19.07**, **19.07**)	(**21.91**, **21.90**)
	TVD	(10.98, 10.98)	(13.13, 13.13)	(15.61, 15.61)	(18.36, 18.36)	(21.42, 21.42)
Barbara	PSS (Degree 1)	(9.76, 9.76)	(11.63, 11.63)	(14.08, 14.08)	(17.08, 17.08)	(20.51, 20.51)
	Haar-OSWST ($s = 1$)	(9.29, 9.29)	(11.71, 11.71)	(14.59, 14.58)	(17.78, 17.78)	(21.24, 21.24)
	Haar-RSWST ($s = 1$)	(**10.56**, **10.55**)	(**12.87**, **12.86**)	(**15.58**, **15.58**)	(**18.61**, **18.61**)	(**21.89**, **21.89**)
	TVD	(9.45, 9.45)	(11.66, 11.66)	(14.48, 14.48)	(17.70, 17.70)	(21.15, 21.15)
fBm	PSS (Degree 1)	(**15.29**, **15.29**)	(**16.87**, **16.87**)	(**18.64**, **18.64**)	(**20.59**, **20.59**)	(**22.85**, **22.85**)
	Haar-OSWST ($s = 1$)	(11.15, 11.14)	(13.12, 13.11)	(15.38, 15.37)	(18.01, 18.00)	(21.05, 21.04)
	Haar-RSWST ($s = 1$)	(13.24, 13.23)	(14.95, 14.95)	(16.90, 16.90)	(19.14, 19.14)	(21.78, 21.78)
	TVD	(12.40, 12.40)	(14.03, 14.03)	(15.93, 15.93)	(18.26, 18.26)	(21.14, 21.14)
MRI	PSS (Degree 1)	(**12.40**, **12.40**)	(14.63, 14.63)	(17.02, 17.02)	(19.59, 19.59)	(22.35, 22.35)
	Haar-OSWST ($s = 1$)	(10.29, 10.29)	(12.86, 12.84)	(15.66, 15.66)	(18.69, 18.68)	(21.90, 21.89)
	Haar-RSWST ($s = 1$)	(12.20, 12.19)	(**14.64**, **14.64**)	(**17.26**, **17.26**)	(**20.08**, **20.07**)	(**23.08**, **23.08**)
	TVD	(11.40, 11.39)	(13.70, 13.70)	(16.24, 16.24)	(19.09, 19.08)	(22.19, 22.19)
Peppers	PSS (Degree 1)	(10.74, 10.74)	(12.47, 12.47)	(14.70, 14.70)	(17.44, 17.44)	(20.68, 20.68)
	Haar-OSWST ($s = 1$)	(10.07, 10.07)	(12.66, 12.64)	(15.53, 15.52)	(18.53, 18.52)	(21.68, 21.68)
	Haar-RSWST ($s = 1$)	(**12.05**, **12.04**)	(**14.57**, **14.56**)	(**17.28**, **17.28**)	(**20.04**, **20.04**)	(**22.88**, **22.88**)
	TVD	(11.22, 11.22)	(13.67, 13.67)	(16.35, 16.35)	(19.17, 19.17)	(22.18, 22.18)
Shepp-Logan	PSS (Degree 1)	(9.91, 9.91)	(11.79, 11.79)	(14.13, 14.13)	(17.06, 17.06)	(20.45, 20.45)
	Haar-OSWST ($s = 1$)	(11.94, 11.93)	(15.47, 15.46)	(19.10, 19.09)	(22.82, 22.81)	(26.62, 26.61)
	Haar-RSWST ($s = 1$)	(13.98, 13.98)	(17.58, 17.58)	(21.28, 21.26)	(25.01, 25.00)	(28.82, 28.81)
	TVD	(**15.33**, **15.32**)	(**18.92**, **18.91**)	(**22.66**, **22.66**)	(**26.38**, **26.37**)	(**30.13**, **30.13**)

†Each cell is formatted as (Oracle value, Estimated value)

TABLE III
COMPARISON OF (λ, s)-OPTIMIZED METHODS†

Image	Input SNR (dB)	4	8	12	16	20
Boats	PSS	(11.85, 11.85)	(13.76, 13.76)	(15.92, 15.91)	(18.38, 18.38)	(21.29, 21.29)
	Haar-OSWST	(11.07, 11.06)	(13.08, 13.06)	(15.48, 15.48)	(18.25, 18.23)	(21.31, 21.31)
	Haar-RSWST	(**12.87**, **12.87**)	(**14.92**, **14.92**)	(**17.16**, **17.16**)	(**19.53**, **19.52**)	(**22.15**, **22.15**)
Barbara	PSS	(9.85, 9.85)	(11.63, 11.63)	(14.23, 14.23)	(17.29, 17.29)	(20.66, 20.66)
	Haar-OSWST	(9.63, 9.62)	(11.92, 11.91)	(14.74, 14.71)	(17.89, 17.88)	(21.33, 21.32)
	Haar-RSWST	(**10.86**, **10.79**)	(**13.03**, **13.01**)	(**15.70**, **15.70**)	(**18.84**, **18.80**)	(**22.12**, **22.12**)
fBm	PSS	(**15.30**, **15.29**)	(**16.89**, **16.89**)	(**18.67**, **18.67**)	(**20.66**, **20.66**)	(**22.95**, **22.95**)
	Haar-OSWST	(12.85, 12.84)	(14.43, 14.42)	(16.24, 16.21)	(18.42, 18.40)	(21.21, 21.21)
	Haar-RSWST	(15.04, 15.00)	(16.68, 16.67)	(18.49, 18.49)	(20.37, 20.36)	(22.16, 22.12)
MRI	PSS	(12.70, 12.70)	(15.19, 15.18)	(17.85, 17.85)	(20.65, 20.65)	(23.51, 23.51)
	Haar-OSWST	(11.09, 11.08)	(13.39, 13.36)	(16.06, 16.01)	(18.95, 18.92)	(22.07, 22.06)
	Haar-RSWST	(**13.73**, **13.72**)	(**16.05**, **16.04**)	(**18.56**, **18.56**)	(**21.21**, **21.17**)	(**23.98**, **23.95**)
Peppers	PSS	(10.74, 10.74)	(12.47, 12.47)	(14.71, 14.71)	(17.51, 17.51)	(20.77, 20.77)
	Haar-OSWST	(10.85, 10.84)	(13.24, 13.23)	(15.97, 15.94)	(18.84, 18.82)	(21.89, 21.88)
	Haar-RSWST	(**12.95**, **12.94**)	(**15.42**, **15.41**)	(**18.07**, **18.06**)	(**20.71**, **20.71**)	(**23.40**, **23.40**)
Shepp-Logan	PSS	(9.92, 9.92)	(11.80, 11.80)	(14.13, 14.13)	(17.06, 17.06)	(20.45, 20.45)
	Haar-OSWST	(12.40, 12.31)	(15.91, 15.85)	(19.51, 19.48)	(23.26, 23.21)	(27.04, 26.99)
	Haar-RSWST	(**14.36**, **14.24**)	(**17.90**, **17.84**)	(**21.57**, **21.54**)	(**25.26**, **25.26**)	(**29.05**, **29.02**)

†Each cell is formatted as (Oracle value, Estimated value)

Shepp–Logan phantom is also less significant because these images are not very fractal-like. In contrast, there is a significant improvement ($\gtrapprox$1 dB at high input SNR) for the MRI image which provides further support for the claim that MRI images are fractal-like and the order s must be matched to the fractal dimension to obtain best results.

As noted in Table III, this combined optimization is shown to produce a consistent SNR increase for both Haar-OSWST and Haar-RSWST methods. In fact, in the redundant case it leads to an increase of about +1 dB for smooth images like Peppers, Boats and fBm at high noise levels. Thus, the optimized Haar-RSWST performs the best of all the considered methods for all natural images which exemplifies the fact that redundant transforms make a powerful denoising tool.

However, it must be emphasized that the results provided in this section are purely for the purpose of illustrating multiparameter optimization of SURE computed by the proposed Monte-Carlo scheme. In our experiments, we considered a set of popular denoising algorithms with adjustable parameters without making any specific claim concerning their overall optimality. In fact, we have intentionally chosen some test images which favour one or the other algorithm to illustrate that the issue of finding a "best" algorithm is not so clear-cut.

The reader who is interested in state-of-the-art methods is referred to the relevant literature; in particular the BiShrink (dual tree complex wavelet decomposition) [50], BLS-GSM (full steerable pyramidal decomposition) [51], ProbShrink (undecimated Daubechies symlets) [52], and SURE-LET (with redundant Haar transform) [20]. Depending on the type of image these more-advanced techniques can yield a further SNR improvement of the order of 1 dB. In some cases such as SURE-LET, they already take full advantage of the possibility of automatic SURE-based parameter adjustment, with the important difference that the underlying solution is explicit as opposed to our black-box approach where it is obtained numerically. The benefit with the latter scheme is that it requires no hypothesis concerning the analytical form of the solution and, therefore, has a wider range of applicability.

VI. Summary and Conclusion

Computation and application of SURE for denoising problems demands the evaluation of the divergence of the denoising operator with respect to the given noisy data. The calculation of this divergence for a general denoising problem may turn out to be nontrivial, especially if the operator does not have explicit analytical form as is the case with iterative algorithms (variational, PDE-based and Bayesian methods). In this paper, we introduced a Monte-Carlo technique that circumvents this difficulty and makes SURE viable for an arbitrary denoising scenario, especially when the computation of the associated divergence is mathematically intractable or numerically infeasible. By adding a perturbation to the signal, our method essentially implements a random first-order difference estimator of the divergence of the denoising operator. From a calculus point of view, this can be related to a stochastic definition of the divergence of a vector field. The final outcome is a black-box scheme which yields SURE numerically using only the output of the denoising algorithm without the need for any knowledge of its internal working.

We demonstrated the applicability of our method by performing Monte-Carlo SURE optimization of some popular denoising algorithms in the wavelet (both orthonormal and redundant) and variational (linear and nonlinear) settings. We found that SURE computed using the proposed technique perfectly predicts the true MSE in all considered cases, thereby yielding correct values for the optimal threshold and the regularization parameter for the respective problems. We also substantiated this argument in the multivariate case by performing SURE-based optimization of the thresholds for denoising by scale-dependent wavelet soft-thresholding. We showed that the SNR obtained by SURE-based optimization is in almost perfect agreement with the oracle solution (minimum MSE) for all considered cases. This suggests that Monte-Carlo SURE can be reliably employed for data-driven adjustment of parameters in a large variety of denoising problems involving Gaussian noise.

Acknowledgment

The authors would like to thank the anonymous reviewer for indicating relevant work [24] in the literature, and Dr. P. Thévénaz for proofreading this paper.

References

[1] W. C. Karl, "Regularization in image restoration and reconstruction," in *Handbook of Image and Video Processing*, A. Bovik, Ed., 2nd ed. New York: Elsevier, 2005, pp. 183–202.
[2] P. Craven and G. Wahba, "Smoothing noisy data with spline functions," *Numer. Math.*, vol. 31, pp. 377–403, 1979.
[3] G. Wahba, *Spline Models for Observational Data*. Philadelphia, PA: SIAM, 1990.
[4] G. H. Golub, M. Heath, and G. Wahba, "Generalized cross-validation as a method for choosing a good ridge parameter," *Technometrics*, vol. 21, no. 2, pp. 215–223, May 1979.
[5] D. Nychka, "Confidence intervals for smoothing splines," *J. Amer. Statist. Assoc.*, vol. 83, pp. 1134–1143, 1988.
[6] A. M. Thompson, J. C. Brown, J. W. Kay, and D. M. Titterington, "A study of methods of choosing the smoothing parameter in image restoration by regularization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 4, pp. 326–339, Apr. 1991.
[7] N. P. Galatsanos and A. K. Katsaggelos, "Methods for choosing the regularization parameter and estimating the noise variance in image restoration and their relation," *IEEE Trans. Image Process.*, vol. 1, no. 3, pp. 322–336, May 1992.
[8] P. C. Hansen, "Analysis of discrete ill-posed problems by means of the L-curve," *SIAM Rev.*, vol. 34, no. 4, pp. 561–580, 1992.
[9] P. C. Hansen and D. P. O'Leary, "The use of the L-curve in the regularization of discrete ill-posed problems," *SIAM J. Sci. Comput.*, vol. 14, no. 6, pp. 1487–1503, 1993.
[10] T. Regińska, "A regularization parameter in discrete ill-posed problems," *SIAM J. Sci. Comput.*, vol. 17, no. 3, pp. 740–749, 1996.
[11] S. Oraintara, W. C. Karl, D. A. Castanon, and T. Q. Nguyen, "A method for choosing the regularization parameter in generalized Tikhonov regularized linear inverse problems," in *Proc. IEEE Int. Conf. Image Processing*, Sep. 2000, vol. 1, pp. 93–96.
[12] R. Molina, A. K. Katsaggelos, and J. Mateos, "Bayesian and regularization methods for hyperparameter estimation in image restoration," *IEEE Trans. Image Process.*, vol. 8, no. 2, pp. 231–246, Feb. 1999.
[13] N. P. Galatsanos, V. Mesarovic, R. Molina, J. Mateos, and A. K. Katsaggelos, "Hyper-parameter estimation using gamma hyper-priors in image restoration from partially-known blurs," *Opt. Eng.*, vol. 41, pp. 1845–1854, 2002.
[14] G. Archer and D. M. Titterington, "On some Bayesian/regularization methods for image restoration," *IEEE Trans. Image Process.*, vol. 4, no. 7, pp. 989–995, Jul. 1995.
[15] A. Mohammad-Djafari, "A full bayesian approach for inverse problems," in *Maximum Entropy and Bayesian Methods*, K. Hanson and R. Silver, Eds. Norwell, MA: Kluwer, 1996.
[16] C. Stein, "Estimation of the mean of a multivariate normal distribution," *Ann. Statist.*, vol. 9, pp. 1135–1151, 1981.
[17] D. L. Donoho and I. M. Johnstone, "Adapting to unknown smoothness via wavelet shrinkage," *J. Amer. Statist. Assoc.*, vol. 90, no. 432, pp. 1200–1224, 1995.
[18] X.-P. Zhang and M. D. Desai, "Adaptive denoising based on SURE risk," *IEEE Signal Process. Lett.*, vol. 5, no. 10, pp. 265–267, Oct. 1998.
[19] F. Luisier, T. Blu, and M. Unser, "A new SURE approach to image denoising: Interscale orthonormal wavelet thresholding," *IEEE Trans. Image Process.*, vol. 16, no. 3, pp. 593–606, May 2007.
[20] T. Blu and F. Luisier, "The SURE-LET approach to image denoising," *IEEE Trans. Image Process.*, vol. 16, no. 11, pp. 2778–2786, Nov. 2007.
[21] C. R. Vogel, *Computational Methods for Inverse Problems*. Philadelphia, PA: SIAM, 2002.
[22] C. L. Mallows, "Some commnets on C_P," *Technometrics*, vol. 15, pp. 661–675, 1973.
[23] G. Gilboa, N. Sochen, and Y. Y. Zeevi, "Estimation of optimal PDE-based denoising in the SNR sense," *IEEE Trans. Image Process.*, vol. 15, no. 8, pp. 2269–2280, Aug. 2006.
[24] D. A. Girard, "The fast Monte-Carlo Cross-Validation and C_L procedures: Comments, new results and application to image recovery problems," *Comput. Statist.*, vol. 10, pp. 205–231, 1995.
[25] K. C. Li, "Asymptotic optimality of c_L and generalized cross-validation in ridge regression with application to spline smoothing," *Ann. Statist.*, vol. 14, no. 3, pp. 1101–1112, 1986.
[26] G. Wahba, D. Johnson, F. Gao, and J. Gong, "Adaptive Tuning of Numerical Weather Prediction Models: Part I: Randomized GCV and Related Methods in Three and Four Dimensionsal Data Assimilation," Tech. Rep. 920, Dept. Statistics, University of Wisconsin, Madison, 1994.

[27] G. Wahba, "The fast Monte-Carlo Cross-Validation and C_L procedures: Comments, new results and application to image recovery problems—Comments," *Comput. Statist.*, vol. 10, pp. 249–250, 1995.
[28] D. A. Girard, "The fast Monte-Carlo Cross-Validation and C_L procedures: Comments, new results and application to image recovery problems—Rejoinder," *Comput. Statist.*, vol. 10, pp. 251–258, 1995.
[29] A. Benazza-Benyahia and J.-C. Pesquet, "Building robust wavelet estimators for multicomponent images using Stein's principle," *IEEE Trans. Image Process.*, vol. 14, no. 11, pp. 1814–1830, Nov. 2005.
[30] W. R. Madych and S. A. Nelson, "Polyharmonic cardinal splines: A minimization property," *J. Approx. Theory*, vol. 60, pp. 141–156, 1990.
[31] C. Rabut, "Elementary m-harmonic cardinal B-splines," *Numer. Alg.*, vol. 2, pp. 39–62, 1992.
[32] M. Unser and T. Blu, "Generalized smoothing splines and the optimal discretization of the Wiener filter," *IEEE Trans. Signal Process.*, vol. 53, no. 6, pp. 2146–2159, Jun. 2005.
[33] S. Tirosh, D. Van De Ville, and M. Unser, "Polyharmonic smoothing splines and the multidimensional Wiener filtering of fractal-like signals," *IEEE Trans. Image Process.*, vol. 15, no. 9, pp. 2616–2630, Sep. 2006.
[34] C. L. Fales, F. O. Huck, J. A. McCormick, and S. K. Park, "Wiener restoration of sampled image data: End-to-end analysis," *J. Opt. Soc. Amer. A*, vol. 5, no. 3, pp. 300–314, 1988.
[35] D. A. Girard, "A fast 'Monte-Carlo Cross-Validation' procedure for large least squares problems with noisy data," *Numer. Math.*, vol. 56, pp. 1–23, 1989.
[36] M. F. Hutchinson, "A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines," *Commun. Statist. Simul. Comput.*, vol. 18, no. 3, pp. 1059–1076, 1989.
[37] Z. Bai, M. Fahey, and G. Golub, "Some large-scale matrix computation problems," *J. Comput. Appl. Math.*, vol. 74, pp. 71–89, 1996.
[38] S. Dong and K. Liu, "Stochastic estimation with z_2 noise," *Phys. Lett. B*, vol. 328, pp. 130–136, 1994.
[39] A. Chambolle, R. A. DeVore, N.-Y. Lee, and B. J. Lucier, "Nonlinear wavelet image processing: Variational problems, compression, and noise removal through wavelet shrinkage," *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 319–335, Mar. 1998.
[40] M. Lang, H. Guo, J. E. Odegard, C. S. Burrus, and R. O. Wells Jr, "Noise reduction using an undecimated discrete wavelet transform," *IEEE Signal Process. Lett.*, vol. 3, no. 1, pp. 10–12, Jan. 1996.
[41] P. Dutilleux, "An implementation of the algorithm à trous to compute the wavelet transform," in *Wavelets: Time-Frequency Methods and Phase Space*, J.-M. Combes, A. Grossman, and P. Tchamitchian, Eds. Berlin, Germany: Spinger-Verlag, 1989, pp. 298–304.
[42] M. Holschneider, R. Kronland-Martinet, J. Morlet, and P. Tchamitchian, "A real-time algorithm for signal analysis with the help of the wavelet transform," in *Wavelets: Time-Frequency Methods and Phase Space*, J.-M. Combes, A. Grossman, and P. Tchamitchian, Eds. Berlin, Germany: Spinger-Verlag, 1989, pp. 286–297.
[43] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D*, vol. 60, no. 1–4, pp. 259–268, Nov. 1992.
[44] M. A. T. Figueiredo, J. B. Dias, J. P. Oliveira, and R. D. Nowak, "On total variation denoising: A new Majorization-Minimization algorithm and an experimental comparison with wavalet denoising," in *Proc. IEEE Int. Conf. Image Processing*, Atlanta, GA, Oct. 2006, pp. 2633–2636.
[45] D. R. Hunter and K. Lange, "A tutorial on MM algorithms," *Amer. Statist.*, vol. 59, no. 1, pp. 30–37, Feb. 2004.
[46] D. Geman and C. Yang, "Nonlinear image recovery with half-quadratic regularization," *IEEE Trans. Image Process.*, vol. 4, no. 7, pp. 932–946, Jul. 1995.
[47] C. M. Hurvich and C. L. Tsai, "A crossvalidatory AIC for hard wavelet thresholding in spatially adaptive function estimation," *Biometrika*, vol. 85, no. 3, pp. 701–710, 1998.
[48] E. Zarahn, G. Aguirre, and M. D'Esposito, "Empirical analyses of BOLD fMRI statistics. I. Spatially unsmoothed data collected under null-hypothesis conditions," *NeuroImage*, vol. 5, pp. 179–197, 1997.
[49] B. Pesquet Popescu and J. L. Véhel, "Stochastic fractal models for image processing," *IEEE Signal Process. Mag.*, vol. 19, no. 5, pp. 48–62, May 2002.
[50] L. Sendur and I. W. Selesnick, "Bivariate shrinkage with local variance estimation," *IEEE Signal Process. Lett.*, vol. 9, no. 12, pp. 438–441, Dec. 2002.
[51] J. Portilla, V. Strela, M. J. Wainwright, and E. P. Simoncelli, "Image denoising using scale mixtures of Gaussians in the wavelet domain," *IEEE Trans. Image Process.*, vol. 12, no. 11, pp. 1338–1351, Nov. 2003.
[52] A. Pižurica and W. Philips, "Estimating the probability of the presence of a signal of interest in multiresolution single and multiband image denoising," *IEEE Trans. Image Process.*, vol. 15, no. 3, pp. 654–665, Mar. 2006.

Sathish Ramani (S'08) received the M.S. degree in physics from Sri Sathya Sai Institute of Higher Learning (SSSIHL), Puttaparthy, Andhra Pradesh, India, in 2000, and the M.S. degree in electrical communication Engineering from the Indian Institute of Science, Bangalore, India, in 2004. He is currently pursuing the Ph.D. degree in the Biomedical Imaging Group, Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland.

His research interests are in splines, interpolation, and variational methods in image processing.

Mr. Ramani received a Best Student Paper Award at the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP) in 2006.

Thierry Blu (M'96–SM'06) was born in Orléans, France, in 1964. He received the "Diplôme d'ingénieur" from the École Polytechnique, France, in 1986, and from Télécom Paris (ENST), France, in 1988, and the Ph.D. degree in electrical engineering from ENST for a study on iterated rational filterbanks, with application to wideband audio coding, in 1996.

Between 1998 and 2007, he was with the Biomedical Imaging Group, Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland. He is currently a Professor in the Department of Electronic Engineering, The Chinese University of Hong Kong. His research interests include (multi)wavelets, multiresolution analysis, multirate filterbanks, interpolation, approximation and sampling theory, image denoising, psychoacoustics, optics, wave propagation, etc.

Dr. Blu was the recipient of two best paper awards from the IEEE Signal Processing Society (2003 and 2006). Between 2002 and 2006, he was an Associate Editor for the IEEE TRANSACTIONS ON IMAGE PROCESSING and, since 2006, he has been an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING.

Michael Unser (F'99) received the M.S. (summa cum laude) and Ph.D. degrees in electrical engineering in 1981 and 1984, respectively, from the Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland.

From 1985 to 1997, he was a Scientist with the National Institutes of Health, Bethesda, MD. He is now a Professor and Director of the Biomedical Imaging Group at the EPFL. His main research topics are biomedical image processing, splines, and wavelets. He is the author of over 120 published journal papers in these areas.

Dr. Unser has been actively involved with the IEEE TRANSACTIONS ON MEDICAL IMAGING, holding the positions of Associate Editor (1999–2002, 2006–present), member of steering committee, and associate Editor-in-Chief (2003–2005). He has acted as Associate Editor or member of the editorial board for eight more international journals, including the *IEEE Signal Processing Magazine*, the IEEE TRANSACTIONS ON IMAGE PROCESSING (1992–1995), and the IEEE SIGNAL PROCESSING LETTERS (1994–1998). He organized the first IEEE International Symposium on Biomedical Imaging (ISBI'2002). He currently chairs the technical committee of the IEEE-SP Society on Bio Imaging and Signal Processing (BISP), as well as the ISBI steering committee. He is the recipient of three Best Paper Awards from the IEEE Signal Processing Society.