# On the Origin of the Bilateral Filter and Ways to Improve It

Michael Elad

***Abstract*—Additive noise removal from a given signal is an important problem in signal processing. Among the most appealing aspects of this field are the ability to refer it to a well-established theory, and the fact that the proposed algorithms in this field are efficient and practical. Adaptive methods based on anisotropic diffusion (AD), weighted least squares (WLS), and robust estimation (RE) were proposed as iterative locally adaptive machines for noise removal. Recently, Tomasi and Manduchi proposed an alternative noniterative bilateral filter for removing noise from images. This filter was shown to give similar and possibly better results to the ones obtained by iterative approaches. However, the bilateral filter was proposed as an intuitive tool without theoretical connection to the classical approaches. In this paper we propose such a bridge, and show that the bilateral filter also emerges from the Bayesian approach, as a single iteration of some well-known iterative algorithm. Based on this observation, we also show how the bilateral filter can be improved and extended to treat more general reconstruction problems.**

***Index Terms*—Anisotropic diffusion, Bayesian methods, bilateral filtering, Jacobi algorithm, robust estimation, weighted least squares.**

## I. Introduction

ADDITIVE noise removal from a given signal is an important problem in signal and image processing [1]–[8]. This problem is the most simplified reconstruction problem in the wider field of signal restoration [9], [10]. Restoring a signal, based on corrupted measurements of it, is typically solved via the Bayesian approach, though there are other approaches [19]–[21]. The Bayesian approach uses some sort of statistical estimator applied on a Gibbs distribution resulting with a penalty functional. This functional is minimized by a numerical optimization algorithm that yields the restored signal [9], [10].

Noise removal is a practical problem raised in many systems. Apart from the trivial application of removing noise prior to presenting the signal to a human observer, pre-smoothing a signal and noise removal may help to improve the performances for many signal-processing algorithms, such as compression, detection, enhancement, recognition, and more. From this aspect, noise removal is appealing both because it relies on a well-established theory, and also because the proposed algorithms in this field are efficient and thus practical.

Manuscript received March 31, 2001; revised April 3, 2002. This work was done while the author was with Hewlett-Packard Laboratories, Israel (HPL-I). The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Scott T. Acton.

M. Elad is with the Computer Science Department (SCCM Program), Stanford University, Stanford, 94306-9025 CA USA (e-mail: elad@sccm.stanford.edu).

Digital Object Identifier 10.1109/TIP.2002.801126.

The more advanced methods for noise removal aim at preserving the signal details while removing the noise. This is achieved by a locally adaptive recovery paradigm. Such methods can be based on anisotropic diffusion (AD) [1]–[5], [22], [23], weighted least squares (WLS) [6], or robust estimation (RE) [7], [8]. The Mumford–Shah functional is a different yet resembling approach toward the same denoising task [11]. All these methods share the fact that local relations between the samples dictate the final result, and therefore, all these methods resort to an iterative algorithm. There is a solid theoretical bridge between these methods as well as to the line-process approach [12], [13].

Recently, Tomasi and Manduchi proposed an alternative noniterative bilateral filter for removing noise from images [14]. This filter is merely a weighted average of the local neighborhood samples, where the weights are computed based on temporal (or spatial in case on images) and radiometric distances between the center sample and the neighboring samples. This filter is also locally adaptive, and it was shown to give similar and possibly better results to those obtained by the previously mentioned iterative approaches. However, The bilateral filter was proposed in [14] as an intuitive tool. Thus, one important aspect that we intend to explore is its relation to the AD, WLS, and RE techniques.

In this paper we propose such a theoretical bridge, and show that the bilateral filter also emerges from the Bayesian approach, using a novel penalty functional. For this functional, we show that a single iteration of the Jacobi algorithm (also known as the diagonal normalized steepest descent—DNSD) yields the bilateral filter. Based on this observation, we also show how the bilateral filter can be improved to further speed-up its smoothing operation, and show how this filter can be extended to treat piece-wise linear signals. Also, it is shown that the bilateral filter can be extended to treat more general reconstruction problems such as image restoration, image scaling, super-resolution, and more.

This paper is organized as follows: In the next section we shortly describe the bilateral filter as was proposed in [14]. Section III describes the AD, WLS, and RE methods. In Section IV we propose a novel penalty term, strongly related to the one in Section III, and yet different. We show how this new penalty term yields the bilateral filter. Section V discusses several improvements to the bilateral filter based on our new model. In Section VI we compare the various methods discussed in this paper for simple one-dimensional (1-D) signals. We summarize this paper in Section VII.

## II. Noise Suppression Via the Bilateral Filter

We start our discussion with a description of the bilateral filter as proposed originally by Tomasi and Manduchi [14]. In order

to simplify the notations we stick to the 1-D case throughout this paper, though all derivations apply to the two-dimensional (2-D) case just as well.

An unknown signal $\underline{X}$ represented as a vector goes through a degradation stage in which a zero-mean white Gaussian noise $\underline{V}$ is added to it. The result is the corrupted signal $\underline{Y}$ given by

$$\underline{Y} = \underline{X} + \underline{V}. \tag{2.1}$$

Our task is to remove this noise and restore $\underline{X}$, given the degraded signal $\underline{Y}$. The bilateral filter suggests a weighted average of pixels in the given image $\underline{Y}$ in order to recover the image $\underline{X}$

$$\hat{X}[k] = \frac{\sum_{n=-N}^{N} W[k,n]Y[k-n]}{\sum_{n=-N}^{N} W[k,n]}. \tag{2.2}$$

This equation is simply a normalized weighted average of a neighborhood of $[2N+1]$ samples around the $k$th sample. The weights $W[k,n]$ are computed based on the content of the neighborhood. For the center sample $X[k]$, the weight $W[k,n]$ is computed by multiplying the following two factors:

$$\begin{aligned} W_S[k,n] &= \exp\left\{-\frac{d^2\{[k],[k-n]\}}{2\sigma_S^2}\right\} = \exp\left\{-\frac{n^2}{2\sigma_S^2}\right\} \\ W_R[k,n] &= \exp\left\{-\frac{d^2\{Y[k],Y[k-n]\}}{2\sigma_R^2}\right\} \\ &= \exp\left\{-\frac{[Y[k]-Y[k-n]]^2}{2\sigma_R^2}\right\}. \end{aligned} \tag{2.3}$$

The final weight is obtained by multiplying these two factors

$$W[k,n] = W_S[k,n] \cdot W_R[k,n]. \tag{2.4}$$

The weight includes two ingredients—temporal (spatial in case of images) and radiometric weights. The first weight measures the geometric distance between the center sample $[k]$ and the $[k-n]$ sample, and Euclidean metric is applied here. This way, close-by samples influence the final result more than distant ones.

The second weight measures the radiometric distance between the values of the center sample $Y[k]$ and the $[k-n]$ sample, and again, Euclidean metric is chosen. Thus, samples with close-by values tend to influence the final result more than those having distant value. Of-course, for both weights we are free to adopt any other reasonable metric. Also, instead of using the Gaussian function, other symmetric and smoothly decaying functions can be used.

Looking at the kernel applied on the input signal at the $k$th sample, this kernel has the following characteristics.

1) The sum of this kernel's coefficients is 1 due to the normalization.
2) The central value of the kernel (the coefficient multiplying the center sample) is the largest. Its size depends on the others due to the normalization.
3) Subject to the above two constraints, the kernel can take any form! We will return to this property as we describe the alternatives, and see that this phenomenon may be the basis for better performance.

The bilateral filter is controlled by three parameters: $N$ dictates the support of the filter. Larger support gives stronger smoothing. The parameters $\sigma_S, \sigma_R$ control the decay of the two weight factors. For very large (infinity) values we get a simple uniform nonadaptive filtering, which is known to degrade the signal edges. Using too small values reduces the smoothing effect.

It is interesting to note that just recently Chan *et al.* [18] proposed a new filter named digital total-variation for noise removal. As it turns out, their filter is very similar to the bilateral filter. However, they chose to use the total-variation penalty to compute the weights, as opposed to the exponential terms used by Tomasi and Manduchi [14]. More importantly, they restricted the application of the filter to a small support, thus losing the filter's prime origins of strength.

## III. Anisotropic Diffusion, WLS, and RE

For the same denoising problem described above, a known approach is to define a penalty functional that best represents our requirements from the unknown $\underline{X}$. We want the result to be as close as possible to the measured signal $\underline{Y}$ while being smooth. Smoothness should be forced in a temporally (spatially) dependent manner in order not to suppress edges in the signal $\underline{X}$. Thus, one either uses weighted least squares (WLS) [6]

$$\begin{aligned} \varepsilon_{\mathrm{WLS}}\{\underline{X}\} = &\frac{1}{2}[\underline{X}-\underline{Y}]^T[\underline{X}-\underline{Y}] \\ &+ \frac{\lambda}{2}[\underline{X}-\mathbf{D}\underline{X}]^T\mathbf{W}(\underline{Y})[\underline{X}-\mathbf{D}\underline{X}] \end{aligned} \tag{3.1}$$

or a robust estimation technique (RE), using an "$M$-function" denoted as [7], [8]

$$\varepsilon_{\mathrm{RE}}\{\underline{X}\} = \frac{1}{2}[\underline{X}-\underline{Y}]^T[\underline{X}-\underline{Y}] + \frac{\lambda}{2}\rho\{\underline{X}-\mathbf{D}\underline{X}\}. \tag{3.2}$$

The matrix $\mathbf{D}$ stands for a one-sample shift to the right (toward the origin) operation. Thus, the term $(X - \mathbf{D}X)$ is simply a discrete approximation of a backward first derivative. If $Z = X - \mathbf{D}X$, then $Z[k] = X[k] - X[k-1]$. As an example, $\mathbf{D}$ is shown for a signal with ten samples. The top line nonzero entry implies circulant operation in order to avoid boundaries effects

$$\mathbf{D} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \tag{3.3}$$

The matrix $\mathbf{W}$ is a diagonal matrix that weights the local gradients. This matrix main diagonal depends on the unknown image $\underline{Y}$. Later we show how this matrix can be constructed. In the RE, the function $\rho(\alpha)$ is symmetric nonnegative function that penalizes gradient values. The choice $\rho(\alpha) = 0.5\alpha^2$ gives the trivial LS approach.

Both these penalty functionals can be shown to emerge from the Bayesian framework and represent the MAP estimation [6]–[10]. In both cases, an iterative algorithm is typically required in order to find the signal $\underline{X}$ that minimizes the functionals [6]–[8]. A natural choice is the steepest descent (SD) algorithm due to its simplicity [15], [16]. This algorithm requires the computation of the first derivative of the functionals

$$\frac{\partial \varepsilon_{\mathrm{WLS}}\{\underline{X}\}}{\partial \underline{X}} = [\underline{X} - \underline{Y}] + \lambda[\mathbf{I} - \mathbf{D}]^T \mathbf{W}(\underline{Y})[\mathbf{I} - \mathbf{D}]\underline{X} \tag{3.4}$$

and

$$\frac{\partial \varepsilon_{\mathrm{RE}}\{\underline{X}\}}{\partial \underline{X}} = [\underline{X} - \underline{Y}] + \lambda[\mathbf{I} - \mathbf{D}]^T \rho'\{[\mathbf{I} - \mathbf{D}]\underline{X}\}. \tag{3.5}$$

The SD stage performed once with $\underline{Y}$ as initialization gives

$$\begin{aligned}\underline{\hat{X}}_1^{\mathrm{WLS}} &= \underline{\hat{X}}_0^{\mathrm{WLS}} - \mu \left.\frac{\partial \varepsilon_{\mathrm{WLS}}\{\underline{X}\}}{\partial \underline{X}}\right|_{\underline{X}=\underline{\hat{X}}_0^{\mathrm{WLS}}} \\ &= \underline{Y} - \mu\lambda(\mathbf{I} - \mathbf{D})^T \mathbf{W}(\underline{Y})(\mathbf{I} - \mathbf{D})\underline{Y}\end{aligned} \tag{3.6}$$

and

$$\begin{aligned}\underline{\hat{X}}_1^{\mathrm{RE}} &= \underline{\hat{X}}_0^{\mathrm{RE}} - \mu \left.\frac{\partial \varepsilon_{\mathrm{RE}}\{\underline{X}\}}{\partial \underline{X}}\right|_{\underline{X}=\underline{\hat{X}}_0^{\mathrm{RE}}} \\ &= \underline{Y} - \mu\lambda(\mathbf{I} - \mathbf{D})^T \rho'\{(\mathbf{I} - \mathbf{D})\underline{Y}\}.\end{aligned} \tag{3.7}$$

Looking at both iterative procedures, we see that they will produce the same solution after the first iteration provided that

$$\begin{aligned}\forall \underline{Y}, \quad \mathbf{W}(\underline{Y})(\mathbf{I} - \mathbf{D})\underline{Y} &= \rho'\{(\mathbf{I} - \mathbf{D})\underline{Y}\} \\ &\Rightarrow \mathbf{W}(\underline{Y}) = \frac{\rho'\{(\mathbf{I} - \mathbf{D})\underline{Y}\}}{(\mathbf{I} - \mathbf{D})\underline{Y}}\end{aligned} \tag{3.8}$$

where the above division is applied entry-by-entry. This equation says that for a specific sample in the signal $\underline{Y}$, denoted as $Y[k]$, its weight is computed by the formula

$$W[k] = \frac{\rho'\{Y[k] - Y[k-1]\}}{Y[k] - Y[k-1]}. \tag{3.9}$$

Black *et al.* linked between the anisotropic diffusion and the robust estimator and obtained a similar formula [12]. We see here that this formula also links between these two approaches (RE and anisotropic diffusion) and the WLS. Table I shows how the weights are obtained for several choices of the function $\rho(\alpha)$.

After the first iteration the WLS and RE depart—whereas the WLS sticks to the same weights, the RE re-computes their values based on the updated solution $\underline{\hat{X}}_1^{\mathrm{RE}}$. Thus, the RE is expected to outperform the WLS, provided that $\rho(\alpha)$ is convex. This is because convexity of $\rho(\alpha)$ implies convexity of the entire functional, and therefore convergence to the global minima point [15], [16].

So far we have been focusing on the WLS and the RE. The anisotropic diffusion (AD) is different in the sense that it uses the continuum to represent its behavior [1]–[5]. However, since eventually we work on a discrete signal, we discretize the propagation equations and get a similar equation to the one shown for the RE method. This is why we did not present this tool separately [12].

The AD, WLS, and RE algorithms are based on a solid theory of statistical estimators and regularization theory [1]–[8]. The bilateral filter, on the other hand, is an ad-hoc filter without theoretic background but nevertheless with impressive results. Both approaches give similar (but not equal) qualitative results [14], though the AD/WLS/RE require many iterations and the bilateral is a "one-pass" algorithm. There are ways to numerically speed the WLS/RE and reduce the number of iterations required [17], but still, the main difference between these two approaches is that the bilateral filter exploits all the relevant neighborhood in parallel, whereas the AD/WLS/RE apply some sort of diffusion of the neighborhood influence.

In the next section we show that the bilateral filter can also be derived from statistical estimation and regularization theories. This result is important for several reasons: First, it gives ways to better understand the bilateral filter. Second, it creates the necessary link between the AD/WLS/RE and the bilateral filter. Third, it enables us to suggest further improvements to the bilateral filter.

TABLE I
CHOICES OF $\rho(\alpha)$ AND THEIR EQUIVALENT WEIGHT FUNCTIONS

| $\rho(\alpha)$ | $\rho'(\alpha)$ | $w(\alpha)$ |
|---|---|---|
| $\frac{1}{2}\alpha^2$ | $\alpha$ | 1 |
| $\lvert\alpha\rvert$ | $\mathrm{sign}(\alpha)$ | $\frac{\mathrm{sign}(\alpha)}{\alpha}$ |
| $\sqrt{\alpha^2+\varepsilon^2}$ | $\alpha/\sqrt{\alpha^2+\varepsilon^2}$ | $1/\sqrt{\alpha^2+\varepsilon^2}$ |
| $\alpha^2/\left(\alpha^2+\varepsilon^2\right)$ | $2\alpha\varepsilon^2/\left(\alpha^2+\varepsilon^2\right)^2$ | $2\varepsilon^2/\left(\alpha^2+\varepsilon^2\right)^2$ |

## IV. Derivation of the Bilateral Filter

Since we have seen equivalence between the AD, the RE and the WLS approaches (in their first iteration), we consider hereafter the WLS due to its formulation simplicity. We propose the following new penalty functional for the unknown signal $\underline{X}$

$$\begin{aligned}\varepsilon\{\underline{X}\} = &\frac{1}{2}[\underline{X} - \underline{Y}]^T[\underline{X} - \underline{Y}] \\ &+ \frac{\lambda}{2}\sum_{n=1}^{N}[\underline{X} - \mathbf{D}^n\underline{X}]^T \mathbf{W}(\underline{Y}, n)[\underline{X} - \mathbf{D}^n\underline{X}].\end{aligned} \tag{4.1}$$

When the matrix $D$ is raised to the power $n$ it implies a shift right of $n$ samples. Thus, as opposed to the previous smoothness terms, the difference between this functional and the one presented in (3.1) is the use of several scales of derivatives, all applied directly on the unknown image. Note that we can suggest a continuous domain form of this penalty function

$$\begin{aligned}\varepsilon\{x(t)\} = &\frac{1}{2}\int_t (x(t) - y(t))^2\, dt \\ &+ \frac{\lambda}{2}\int_t \int_{\tau=0}^{T} w(y(t), \tau) \cdot ((1 - \delta(t - \tau)) \otimes x(t)^2\, d\tau\, dt\end{aligned}$$

and apply variational-calculus for its minimization. Here, we employ a convolution denoted by $\otimes$) with the Kronecker shifted delta function in order to facilitate the shift. The integral over $\tau$ gives the resulting filter its wide support from $-T$ to $T$.

Taking the first derivative of (4.1) with respect to the unknown $\underline{X}$ we get the following gradient vector

$$\begin{aligned}\frac{\partial \varepsilon\{\underline{X}\}}{\partial \underline{X}} &= [\underline{X}-\underline{Y}]+\lambda\sum_{n=1}^{N}(\mathbf{I}-\mathbf{D}^n)^T\mathbf{W}(\underline{Y},n)(\mathbf{I}-\mathbf{D}^n)\underline{X} \\ &= \left[\mathbf{I}+\lambda\sum_{n=1}^{N}(\mathbf{I}-\mathbf{D}^{-n})\mathbf{W}(\underline{Y},n)(\mathbf{I}-\mathbf{D}^n)\right]\underline{X}-\underline{Y}\end{aligned} \tag{4.2}$$

where we have used the relation $(\mathbf{D}^n)^T = (\mathbf{D}^T)^n = (\mathbf{D}^{-1})^n = \mathbf{D}^{-n}$.

If we assume again a single iteration of the SD algorithm applied with $\underline{Y}$ as the initialization, we get

$$\underline{\hat{X}}_1 = \left[\mathbf{I}-\mu\lambda\sum_{n=1}^{N}(\mathbf{I}-\mathbf{D}^{-n})\mathbf{W}(\underline{Y},n)(\mathbf{I}-\mathbf{D}^n)\right]\underline{Y}. \tag{4.3}$$

Speeding-up the above iteration can be done using locally adaptive step-sizes $\mu$. We can use the inverse of the main diagonal of the Hessian matrix—the second derivative of the penalty functional [15], [16]. This algorithm is known as the Jacobi algorithm, or the diagonal normalized steepest descent (DNSD) [15], [16]. The second derivative is the following matrix:

$$\frac{\partial^2\varepsilon\{\underline{X}\}}{\partial \underline{X}^2} = \mathbf{H}(\underline{Y}) = \mathbf{I}+\lambda\sum_{n=1}^{N}(\mathbf{I}-\mathbf{D}^{-n})\mathbf{W}(\underline{Y},n)(\mathbf{I}-\mathbf{D}^n). \tag{4.4}$$

From this matrix, we need to extract the main diagonal, which by definition is known to contain real and positive values, assuming that the weights are all positive. We define a step-size matrix $\mathbf{M}$, which extends the notion of the previously used $\mu$ by

$$\mathbf{M}(\underline{Y}) = [\xi\mathbf{I}+\mathrm{diag}\{\mathbf{H}(\underline{Y})\}]^{-1}. \tag{4.5}$$

The additional term $\xi\mathbf{I}$ relaxes the step-size matrix and ensures stability. Thus, the DNSD iteration is

$$\begin{aligned}\underline{\hat{X}}_1 &= \underline{Y}-\lambda\mathbf{M}(\underline{Y})\sum_{n=1}^{N}(\mathbf{I}-\mathbf{D}^{-n})\mathbf{W}(\underline{Y},n)(\mathbf{I}-\mathbf{D}^n)\underline{Y} \\ &= \left[\mathbf{I}-\lambda\mathbf{M}(\underline{Y})\sum_{n=1}^{N}(\mathbf{I}-\mathbf{D}^{-n})\mathbf{W}(\underline{Y},n)(\mathbf{I}-\mathbf{D}^n)\right]\underline{Y}.\end{aligned} \tag{4.6}$$

In both the SD and the DNSD we get that the solution $\underline{\hat{X}}_1$ is obtained via a linear operation on the distorted image $\underline{Y}$. Referring to this linear operation as a signal-dependent linear filer, we may ask what is the kernel applied on a neighborhood of a sample? As we show in the next analysis, the kernel is the bilateral filter. In order to show this result, we first have to specify how the weights are chosen. We can use the formula in (3.8)

$$\mathbf{W}(\underline{Y},n) = \frac{\rho'\{(\mathbf{I}-\mathbf{D}^n)\underline{Y}\}}{(\mathbf{I}-\mathbf{D}^n)\underline{Y}}. \tag{4.7}$$

However, the weights here should also reflect our decreased confidence in the smoothness penalty term as $n$ grows toward $N$. Thus, a reasonable choice is

$$\mathbf{W}(\underline{Y},n) = \frac{\rho'\{(\mathbf{I}-\mathbf{D}^n)\underline{Y}\}}{(\mathbf{I}-\mathbf{D}^n)\underline{Y}}\cdot V(n) \tag{4.8}$$

for some nonnegative symmetric and monotonically decreasing function $V(n)$ (e.g., $V(n)=\alpha^n, 0<\alpha<1$).

Note that, as far as the first iteration is involved, the iterative equation can be written alternatively as

$$\underline{\hat{X}}_1 = \underline{Y}-\lambda\mathbf{M}(\underline{Y})\sum_{n=1}^{N}V(n)\cdot(\mathbf{I}-\mathbf{D}^{-n})\rho'\{(\mathbf{I}-\mathbf{D}^n)\underline{Y}\}. \tag{4.9}$$

Looking at (4.6) and choosing the $k$th sample (assumed not to be close to the signal's boundaries) from the result, this value is computed in the following way:

$$\hat{X}_1[k] = Y[k]-\left[\lambda\mathbf{M}(\underline{Y})\sum_{n=1}^{N}(\mathbf{I}-\mathbf{D}^{-n})\times\mathbf{W}(\underline{Y},n)(\mathbf{I}-\mathbf{D}^n)\underline{Y}\right]\Bigg|_k. \tag{4.10}$$

Let us introduce several temporal notations

$$\begin{aligned}\underline{A}_n &= (\mathbf{I}-\mathbf{D}^n)\underline{Y} \\ \underline{B}_n &= \mathbf{W}(\underline{Y},n)(\mathbf{I}-\mathbf{D}^n)\underline{Y} = \mathbf{W}(\underline{Y},n)\underline{A}_n \\ \underline{C}_n &= (\mathbf{I}-\mathbf{D}^{-n})\mathbf{W}(\underline{Y},n)(\mathbf{I}-\mathbf{D}^n)\underline{Y} = (\mathbf{I}-\mathbf{D}^{-n})\underline{B}_n.\end{aligned} \tag{4.11}$$

Thus, per the sample $k$ we can write the following relations:

$$\begin{aligned}A_n[k] &= Y[k]-Y[k-n] \\ \mathbf{W}(\underline{Y},n)|_k &= V(n)\frac{\rho'\{Y[k]-Y[k-n]\}}{Y[k]-Y[k-n]} \\ B_n[k] &= V(n)\frac{\rho'\{Y[k]-Y[k-n]\}}{Y[k]-Y[k-n]}\cdot(Y[k]-Y[k-n]), \\ C_n[k] &= B_n[k]-B_n[k+n] \\ &= V(n)\frac{\rho'\{Y[k]-Y[k-n]\}}{Y[k]-Y[k-n]}\cdot(Y[k]-Y[k-n]) \\ &\quad + V(n)\frac{\rho'\{Y[k]-Y[k+n]\}}{Y[k]-Y[k+n]}\cdot(Y[k]-Y[k+n]).\end{aligned}$$

In this expression, we exploited the fact that the function $\rho(\alpha)$ is known to be symmetric and therefore its first derivative is

unsymmetric. Thus

$$\rho'\{Y[k+n]-Y[k]\} = -\rho'\{Y[k]-Y[k+n]\}.$$

Gathering together all the terms for all values of $n$ we get

$$\hat{X}_1[k] = Y[k] - \lambda \mathbf{M}(\underline{Y})|_k \cdot \sum_{n=-N}^{N} V(n) \cdot \frac{\rho'\{Y[k]-Y[k-n]\}}{(Y[k]-Y[k-n])}(Y[k]-Y[k-n]). \quad (4.12)$$

We can refer to the above expression as a time varying convolution of the form

$$\hat{X}_1[k] = \sum_{\ell=-N}^{N} f[\ell,k]\cdot Y[k-\ell] \quad (4.13)$$

where this filter's coefficients $f$ are given by (4.14), shown at the bottom of the page. We now turn to analyze the expression $\mathbf{M}(\underline{Y})|_k$. Using (4.4) and (4.5), we get

$$\mathbf{M}(\underline{Y})|_k = \frac{1}{\xi + \mathbf{H}(\underline{Y})|_{[k,k]}} = \frac{1}{\xi + 1 + \lambda\sum_{n=-N}^{N} V(n)\cdot\frac{\rho'\{Y[k]-Y[k-n]\}}{Y[k]-Y[k-n]}}. \quad (4.15)$$

Thus, the filter coefficients are shown in (4.16) at the bottom of the page. From the above results, and the description of the bilateral filter in Section II, we can draw several observations:

1) The sum of all coefficients is 1, as should be in the bilateral filter.
2) If the $M$-function $\rho(\alpha)$ is symmetric and monotonic nondecreasing (from 0 to $\infty$), all the filter coefficients are nonnegative.
3) If the $M$-function $\rho(\alpha)$ is subquadratic (i.e., $|\rho'(\alpha)| \leq |\alpha|$), and if $\lambda V(n) \leq 1$, then $\forall \ell \,|\, -N \leq \ell \leq N, f[\ell,k] \leq f[0,k]$.
4) The coefficient $f[\ell,k]$ represents the weight according to which $Y[k-\ell]$ contributes to the evaluation of the restored pixel $X[k]$. This coefficient includes two parts: the spatial weight $V(\ell)$ and the radiometric weight given by $\rho'\{Y[k]-Y[k-\ell]\}/(Y[k]-Y[k-\ell])$. These two parts are the same as described in Section II for the Tomasi–Manduchi bilateral filter. In this relation, we have the following correspondence:

$$W_S[k,\ell] \Rightarrow V(\ell)$$
$$W_R[k,\ell] \Rightarrow \frac{\rho'\{Y[k]-Y[k-\ell]\}}{(Y[k]-Y[k-\ell])}. \quad (4.17)$$

Thus, if we choose

$$V(\ell) = \exp\left\{-\frac{\ell^2}{2\sigma_S^2}\right\} \quad (4.18)$$

and

$$\frac{\rho'\{Y[k]-Y[k-n]\}}{(Y[k]-Y[k-n])} = \exp\left\{-\frac{[Y[k]-Y[k-n]]^2}{2\sigma_R^2}\right\} \Rightarrow \rho(\alpha) = -\sigma_R^2 \exp\left\{-\frac{\alpha^2}{2\sigma_R^2}\right\} \quad (4.19)$$

we get the same filter as in the bilateral filter, as described in [14].

5) Increasing the value of $\xi$ or reducing the value of $\lambda$, we get that the obtained filter tends toward the unit filter (the identity operation, leaving the image intact).

We have obtained the bilateral filter by Tomasi and Manduchi [14]. Thus, we have that the bilateral filter is merely a single iteration of the DNSD (or Jacobi) algorithm, using the penalty functional proposed in (4.1). This penalty functional is different from the one used by the WLS/RE in the regularization term. Whereas the WLS/RE penalizes smoothness with the first neighboring pixels, the new term penalizes nonsmoothness with distant neighbors as well.

## V. Improvements of the Bilateral Filter

Now that we have an origin for the bilateral filter, we can consider several improvements. In this section we will describe

1) how to speed-up the bilateral filter and increase its smoothing effect;
2) How to implement a bilateral filter for piece-wise linear signals.

In the following presentation we stick to the formulation of the bilateral filter as presented in the previous section.

Before we turn to describe the aforementioned improvements, we emphasize that the smoothness penalty term that originates the bilateral filter as given in (4.1) can also be

$$f[\ell,k] = \begin{cases} \lambda \mathbf{M}(\underline{Y})|_k \cdot V(\ell)\cdot\dfrac{\rho'\{Y[k]-Y[k-\ell]\}}{(Y[k]-Y[k-\ell])} & \ell \neq 0 \\ 1 - \lambda \mathbf{M}(\underline{Y})|_k \cdot \sum_{n=-N}^{N} V(n)\cdot\dfrac{\rho'\{Y[k]-Y[k-n]\}}{(Y[k]-Y[k-n])} & \ell = 0 \end{cases} \quad (4.14)$$

$$f[\ell,k] = \begin{cases} \dfrac{\lambda V(\ell)\cdot\frac{\rho'\{Y[k]-Y[k-\ell]\}}{(Y[k]-Y[k-\ell])}}{\xi+1+\lambda\sum_{n=-N}^{N} V(n)\cdot\frac{\rho'\{Y[k]-Y[k-n]\}}{Y[k]-Y[k-n]}} & -\begin{bmatrix} N \leq \ell \leq N, \\ \ell \neq 0 \end{bmatrix} \\ \dfrac{\xi+1}{\xi+1+\lambda\sum_{n=-N}^{N} V(n)\cdot\frac{\rho'\{Y[k]-Y[k-n]\}}{Y[k]-Y[k-n]}} & [\ell = 0] \end{cases} \quad (4.16)$$

used for a variety of restoration and reconstruction problems. Among these problems are image restoration, image scaling, super-resolution reconstruction, optical flow estimation and more [9], [10].

Also, as a consequence of our findings, we can simply suggest the application of the bilateral filter several times, implying several iterations of the DNSD. By doing so, the signal is smoothed and gets to a steady state, which will be the global minimum point of the functional in (4.1).

### A. Speeding-Up the Bilateral Filter

The bilateral filter can be speeded-up in one of two methods, and any combination of them. Given a general quadratic penalty function of the form

$$\varepsilon\{\underline{X}\} = \sum_{j=1}^{J}\left[\frac{1}{2}\underline{X}^T Q_j \underline{X} - \underline{P}_j^T \underline{X} + C_j\right] \tag{5.1}$$

the SD iteration reads

$$\underline{\hat{X}}_1 = \underline{\hat{X}}_0 - \left.\frac{\partial\varepsilon\{\underline{X}\}}{\partial\underline{X}}\right|_{\underline{\hat{X}}_0} = \underline{\hat{X}}_0 - \mu\sum_{j=1}^{J}[Q_j\underline{\hat{X}}_0 - \underline{P}_j]. \tag{5.2}$$

Clearly, the resemblance between (5.1) and (4.1) is evident if we choose the $\{Q_j, \underline{P}_j\}_j$ from (5.1) as

$$Q_1 = I, \quad Q_k = (I - \mathbf{D}^{k-1})^T \mathbf{W}(\underline{Y}, k-1)(I - \mathbf{D}^{k-1}) \quad \text{for } k = 2,3,\ldots,N+1$$

and

$$\underline{P}_1 = \underline{Y}, \quad \underline{P}_1 = \underline{P}_2 = \underline{P}_3 = \cdots = \underline{P}_{N+1} = \underline{0} \quad \text{for } k = 2,3,\ldots,N+1.$$

One way to speed the SD convergence is the Gauss-Siedel approach [15], [16]. In this approach, the samples of $\underline{\hat{X}}_1$ are computed sequentially from $\hat{X}_1[1]$ to $\hat{X}_1[L]$ (assuming $L$ scalar samples in the vector $\underline{\hat{X}}_1$), and for the calculation of $\underline{\hat{X}}_1[k]$, updated values of $\underline{\hat{X}}_1$ are used (instead of only $\underline{\hat{X}}_0$ values). This "bootstrap" method is known to be stable and converge to the same global minimum point of the penalty function given in (5.1) [15], [16]. A more systematic way to describe this process is via the decomposition of the Hessian to the upper-triangle, lower-triangle, and diagonal parts

$$\begin{aligned}
\underline{\hat{X}}_1 &= \underline{\hat{X}}_0 - \mu\sum_{j=1}^{J}[Q_j\underline{\hat{X}}_0 - \underline{P}_j] \\
&= \underline{\hat{X}}_0 + \mu\sum_{j=1}^{J}\underline{P}_j - \mu\left[\sum_{j=1}^{J}Q_j\right]\underline{\hat{X}}_0 \\
&= \underline{\hat{X}}_0 + \mu\sum_{j=1}^{J}\underline{P}_j - \mu\left[\sum_{j=1}^{J}Q_j\right]_{\text{Upper}}\underline{\hat{X}}_0 \\
&\quad - \mu\left[\sum_{j=1}^{J}Q_j\right]_{\text{Lower}}\underline{\hat{X}}_0 - \mu\left[\sum_{j=1}^{J}Q_j\right]_{\text{diag}}\underline{\hat{X}}_0.
\end{aligned} \tag{5.3}$$

In this equation applied for $\underline{\hat{X}}_1[k]$, the lower-triangle part multiplies elements of $\underline{\hat{X}}_0$ with indices smaller than $k$. Instead, we can use these entries taken from $\underline{\hat{X}}_1$. Moreover, since the last term in (5.3) is a diagonal matrix multiplying $\underline{\hat{X}}_0$, we can exploit it as well and replace the multiplication by $\underline{\hat{X}}_0[k]$ with multiplication by $\underline{\hat{X}}_1[k]$. Thus, we get

$$\begin{aligned}
\underline{\hat{X}}_1 &= \underline{\hat{X}}_0 + \mu\sum_{j=1}^{J}\underline{P}_j - \mu\left[\sum_{j=1}^{J}Q_j\right]_{\text{Upper}}\underline{\hat{X}}_0 \\
&\quad - \mu\left[\sum_{j=1}^{J}Q_j\right]_{\text{Lower}}\underline{\hat{X}}_1 - \mu\left[\sum_{j=1}^{J}Q_j\right]_{\text{diag}}\underline{\hat{X}}_1 \\
&\Downarrow \\
\underline{\hat{X}}_1 &= \left\{I + \mu\left[\sum_{j=1}^{J}Q_j\right]_{\text{diag}}\right\}^{-1} \cdot \left\{\underline{\hat{X}}_0 + \mu\sum_{j=1}^{J}\underline{P}_j\right. \\
&\quad \left. - \mu\left[\sum_{j=1}^{J}Q_j\right]_{\text{Upper}}\underline{\hat{X}}_0 - \mu\left[\sum_{j=1}^{J}Q_j\right]_{\text{Lower}}\underline{\hat{X}}_1\right\}.
\end{aligned} \tag{5.4}$$

The inversion required above is trivial since the inverted matrix is diagonal.

A different alternative for speeding the bilateral filter is to exploit the fact that the gradient is naturally sliced into several parts. Returning to (5.2) we can rewrite it as $J$ iterations of the form

$$\begin{aligned}
\underline{\hat{X}}_1 &= \underline{\hat{X}}_0 - \mu[Q_1\underline{\hat{X}}_0 - \underline{P}_1] \\
\underline{\hat{X}}_2 &= \underline{\hat{X}}_1 - \mu[Q_2\underline{\hat{X}}_1 - \underline{P}_2] \\
\underline{\hat{X}}_3 &= \underline{\hat{X}}_2 - \mu[Q_3\underline{\hat{X}}_2 - \underline{P}_3] \\
&\vdots \\
\underline{\hat{X}}_J &= \underline{\hat{X}}_{J-1} - \mu[Q_J\underline{\hat{X}}_{J-1} - \underline{P}_J].
\end{aligned} \tag{5.5}$$

This way, the final solution $\underline{\hat{X}}_J$ is closer to the global minimum point of the penalty function in (5.1). Similarly, the gradient in (4.6) can be sliced into $(N+1)$ parts and can be applied as $(N+1)$ iterations. The final result will be closer to the global minimum point of (4.1), which means a smoother result.

Note that by applying $J$ iterations, the computational load is similar to the one required with $J$ iterations of the WLS/RE methods. However, the results are expected to be totally different, since by applying different kinds of derivatives (due to the use of different neighbors) we get stronger smoothing effect. Also, in this algorithm, if we smooth the $k$th sample, if the $[k+n]$ sample does not contribute to the center sample due to its distant value, this does not imply anything about further distant neighbors behind it (e.g., $[k+n+1]$, etc.). In contrast to this, in the WLS and the RE, noncontributing samples imply that all samples behind them are masked and cannot contribute as well.

### B. Treating Piece-Wise Linear Signals

The penalty functional in (4.1) is designed for piece-wise constant signals. This is because our smoothness penalty term penalizes for the first derivative of the signal in various scales.

Instead, we propose an alternative functional designed for piece-wise linear signals

$$\varepsilon\{\underline{X}\} = \frac{1}{2}[\underline{X}-\underline{Y}]^T[\underline{X}-\underline{Y}] + \frac{\lambda}{2}\sum_{n=1}^{N}\left[\underline{X}-\frac{\mathbf{D}^n\underline{X}+\mathbf{D}^{-n}\underline{X}}{2}\right]^T \times \mathbf{W}(\underline{Y},n)\left[\underline{X}-\frac{\mathbf{D}^n\underline{X}+\mathbf{D}^{-n}\underline{X}}{2}\right]. \quad (5.6)$$

The first and the second derivatives of (5.6) with respect to the unknown $\underline{X}$ yield

$$\frac{\partial\varepsilon\{\underline{X}\}}{\partial\underline{X}} = \left[\mathbf{I}+\lambda\sum_{n=1}^{N}\left(\mathbf{I}-\frac{\mathbf{D}^n+\mathbf{D}^{-n}}{2}\right) \times \mathbf{W}(\underline{Y},n)\left(\mathbf{I}-\frac{\mathbf{D}^n+\mathbf{D}^{-n}}{2}\right)\right]\underline{X}-\underline{Y} \quad (5.7)$$

and

$$\frac{\partial^2\varepsilon\{\underline{X}\}}{\partial\underline{X}^2} = \mathbf{H}(\underline{Y}) = \mathbf{I}+\lambda\sum_{n=1}^{N}\left(\mathbf{I}-\frac{\mathbf{D}^n+\mathbf{D}^{-n}}{2}\right) \times \mathbf{W}(\underline{Y},n)\left(\mathbf{I}-\frac{\mathbf{D}^n+\mathbf{D}^{-n}}{2}\right). \quad (5.8)$$

From these expressions we can extract the DNSD iteration with $\underline{Y}$ as initialization. After several tedious algebraic steps we get that, again, the input signal $\underline{Y}$ is convolved with a locally adaptive filter of the form

$$\underline{\hat{X}}_1[k] = \sum_{\ell=-N}^{N} f[\ell,k]\cdot Y[k-\ell]. \quad (5.9)$$

The coefficients of this filter are given by (5.10), shown at the bottom of the page. This means that the spatial weight does not change, and the radiometric weights simply uses a second derivative instead of the first one.

*Remark:* In going to 2-D signals, the various penalty functions need to be updated accordingly. For example, (3.1) turns to be

$$\varepsilon_{\mathrm{WLS}}\{\underline{X}\} = \frac{1}{2}[\underline{X}-\underline{Y}]^T[\underline{X}-\underline{Y}] + \frac{\lambda}{2}[\underline{X}-\mathbf{D}_H\underline{X}]^T\mathbf{W}_H(\underline{Y})[\underline{X}-\mathbf{D}_H\underline{X}] + \frac{\lambda}{2}[\underline{X}-\mathbf{D}_V\underline{X}]^T\mathbf{W}_V(\underline{Y})[\underline{X}-\mathbf{D}_V\underline{X}]$$

where $\mathbf{D}_H$ is a horizontal shift operator by one sample, and similarly, $\mathbf{D}_V$ is a vertical shift operator by one sample. Note that the weights also reflect the derivative direction, using an extended form of (3.8). In a similar manner, (4.1) becomes

$$\varepsilon\{\underline{X}\} = \frac{1}{2}[\underline{X}-\underline{Y}]^T[\underline{X}-\underline{Y}] + \frac{\lambda}{2}\sum_{n_1=0}^{N}\sum_{\substack{n_2=0\\ n_1+n_2>0}}^{N}[\underline{X}-\mathbf{D}_H^{n_1}\mathbf{D}_V^{n_2}\underline{X}]^T \times \mathbf{W}(\underline{Y},n_1,n_2)[\underline{X}-\mathbf{D}_H^{n_1}\mathbf{D}_V^{n_2}\underline{X}]$$

and again, the weights need to be computed using the same 2-D shift.

## VI. Simulations

In this section, we demonstrate how the various discussed approaches compare with each other. All our simulations are performed on simple and synthetic 2-D signals. Fig. 1 presents a piece-wise constant test image ($\underline{X}$) and its noisy version ($\underline{Y}$). The additive noise is Gaussian with zero mean and variance $\sigma_n = 0.2$ and the values of the image $X$ are in the range [1, 7]. All the following simulations aim at removing this noise and reconstructing $\underline{X}$ from $\underline{Y}$.

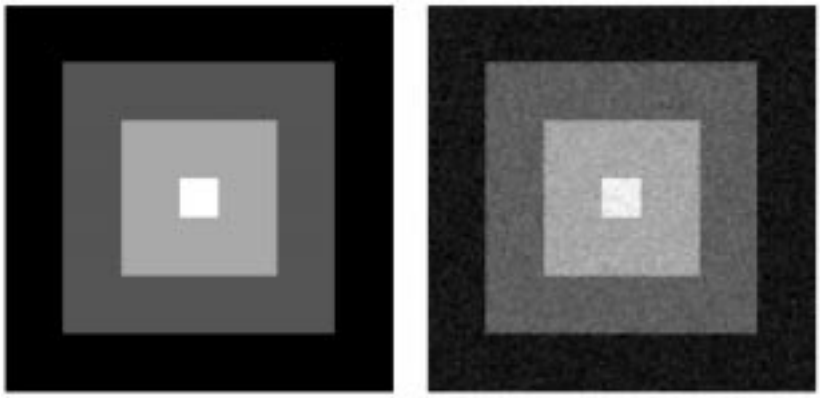
Fig. 2 shows the results obtained by the WLS and the RE methods. The WLS was applied with weights computed via the assumption $\rho(\alpha) = |\alpha|$, choosing $\lambda = 1$, and applying 50 SD iterations. Similarly, the RE used $\rho(\alpha) = |\alpha|, \lambda = 1$ and 50 iterations. The obtained mean-square-error (MSE) gain[1] in the WLS method is 3.90. The MSE gain for the RE is 10.99. It is evident that the RE method is far better compared to the WLS both visually and via the MSE assessment.

Fig. 3 shows the result obtained by the bilateral filter with weights as given in (4.18) and (4.19). The parameters in this simulation are the following: $\lambda = 1, \sigma_s = 2.5, \sigma_R = 0.5, N = 6$ (i.e., the filter support is $13 \times 13$ pixels). A single application of this filter gave an MSE gain of 23.50. This result is roughly equivalent and slightly better, compared to the RE result after 50 iterations. Fig. 3 also presents the result after ten iterations of the bilateral filter. The MSE gain in this case is 318.90.

The basic bilateral filter (single iteration) gave an MSE gain of 23.50. Applying the Gauss-Siedel with the same parameters (and thus, having the exact same complexity) we got an MSE

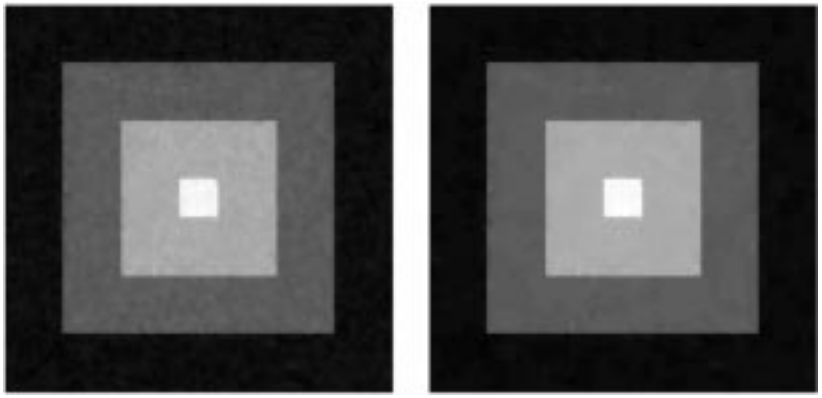
[1]This gain is defined as the ratio between the MSE before and after the filtering. Thus, this value is always positive, and larger values indicate better filtering result.

$$f[\ell,k] = \begin{cases} \dfrac{\lambda V(\ell)\cdot\dfrac{\rho'\left\{\underline{Y}[k]-\frac{\underline{Y}[k-\ell]+Y[k+\ell]}{2}\right\}}{\left(\underline{Y}[k]-\frac{\underline{Y}[k-\ell]+Y[k+\ell]}{2}\right)}}{\xi+1+\lambda\sum_{n=-N}^{N}V(n)\cdot\dfrac{\rho'\left\{\underline{Y}[k]-\frac{\underline{Y}[k-n]+Y[k+n]}{2}\right\}}{\left(\underline{Y}[k]-\frac{\underline{Y}[k-n]+Y[k+n]}{2}\right)}} & -\begin{bmatrix} N\le\ell\le N,\\ \ell\neq 0\end{bmatrix} \\ \dfrac{\xi+1}{\xi+1+\lambda\sum_{n=-N}^{N}V(n)\cdot\dfrac{\rho'\left\{\underline{Y}[k]-\frac{\underline{Y}[k-n]+Y[k+n]}{2}\right\}}{\left(\underline{Y}[k]-\frac{\underline{Y}[k-n]+Y[k+n]}{2}\right)}} & [\ell=0] \end{cases} \quad (5.10)$$



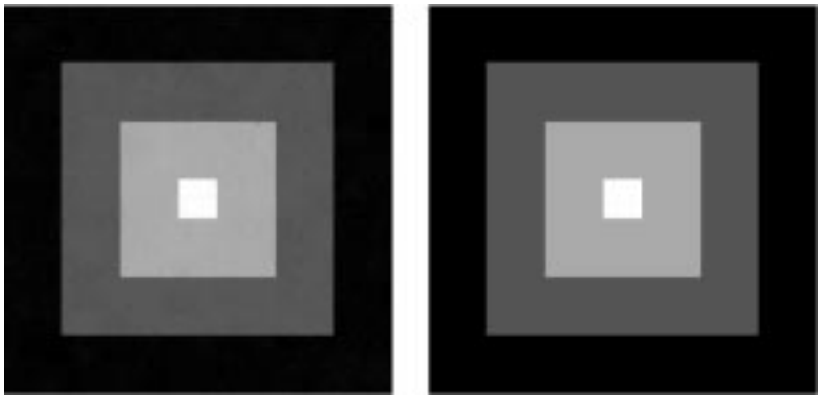
(a) (b)

Fig. 1. (a) Piece-wise constant test image and (b) its noisy version.



(a) (b)

Fig. 2. WLS (a): $\mathrm{MSE\ gain} = 3.90$ and the RE (b): $\mathrm{MSE\ gain} = 10.99$ solutions after 50 SD iterations.



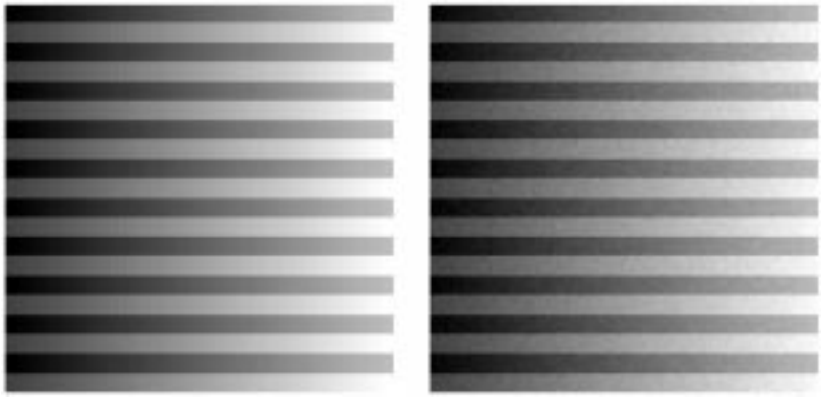
(a) (b)

Fig. 3. Bilateral filter result after one iteration (a): $\mathrm{MSE\ gain} = 23.50$ and after ten iterations (b): $\mathrm{MSE\ gain} = 318.90$.

gain of 39.44. Applying the second speed-up approach with the sliced gradient terms [as proposed in (5.5)] we got an MSE gain of 197.26! In both cases, the gain is mostly evident via the MSE since the images are very close to the original $X$. Therefore there is no point is presenting the image results in a figure.

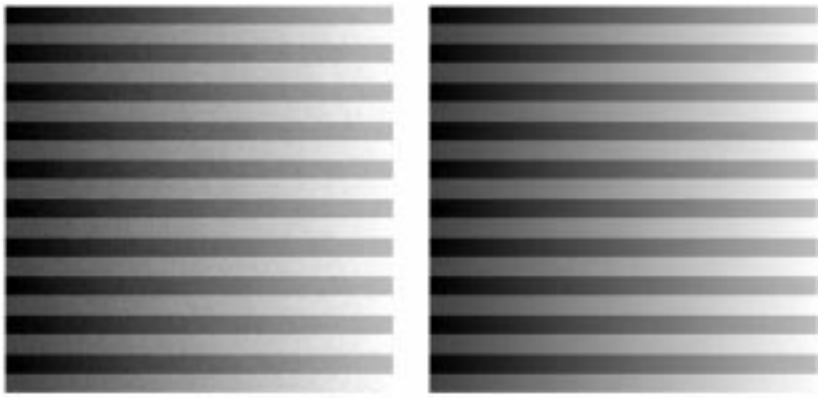
Fig. 4 shows a piece-wise linear image $\underline{X}$ and its noisy version $Y$. The noise variance is $\sigma_n = 0.2$ and the values of $X$ are in the range [0, 16]. An attempt to recover this signal using the regular bilateral gave an MSE gain of 1.53. We see that there is almost no effective filtering. A piece-wise-linear compatible bilateral filter as proposed in (5.9) gave an MSE gain of 12.91. Fig. 5 shows these results. Fig. 6 shows the reconstruction error for the two cases (using a gain of 80 in order to extend the dynamic range to a visible image), and it is quite evident that the new approach is much better.

As a final point in this section, we return to the claim about the continuity of the filter coefficients in the RE, and the fact that it is not so for the bilateral filter. We discussed this behavior at the end of Section V-A. Fig. 7 shows a noisy piece-wise image (checkerboard), where the size of every constant piece is $4 \times 4$ pixels. The original image values are in the range [0, 4] and the noise variance is $\sigma_n = 0.2$, as before.



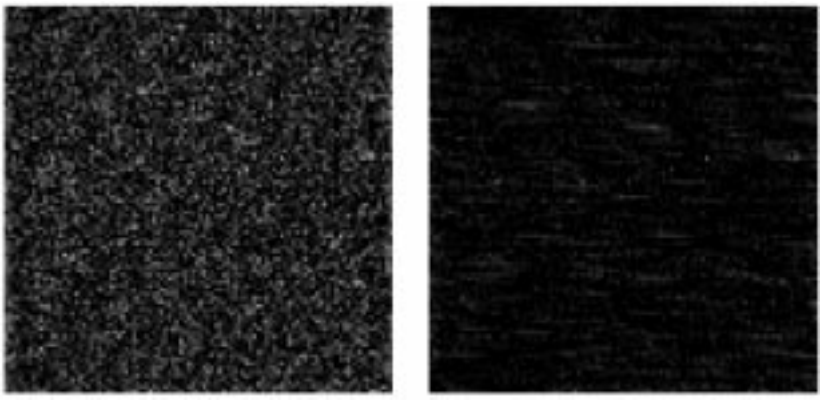
(a) (b)

Fig. 4. (a) Piece-wise linear test signal and (b) its noisy version.



(a) (b)

Fig. 5. Bilateral filtering results for the piece-wise linear case: (a) regular bilateral filter and (b) the piece-wise-linear compatible bilateral filter.



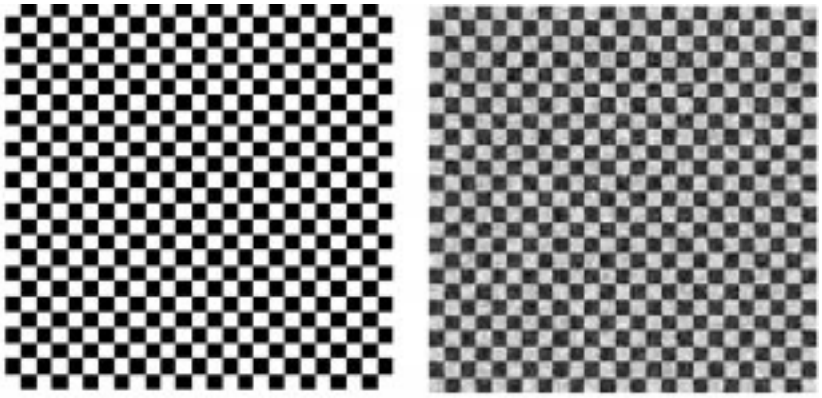
(a) (b)

Fig. 6. Bilateral filtering error images for the piece-wise linear case: (a) the regular bilateral filter and (b) the piece-wise-linear compatible bilateral filter.

This signal was filtered by the RE (same parameters as before and using 1500 iterations) resulting with an MSE gain of 2.32. A single iteration of the bilateral filter $(\lambda = 1, \sigma_s = 5, \sigma_R = 0.5, N = 6)$ gave an MSE gain of 19.97. Fig. 8 shows these results. The main reason for the much better performance with the bilateral is its ability to create a local filter that has a nonconnected structure. Fig. 9 shows the weights for a specific pixel in the image. We see that the weights contain a spatially decaying behavior due to the $W_S$, and a checkerboard structure induced by the radiometric weights $W_R$. As can be seen, the effective support of the filter extends to exploit most of the relevant neighborhood, without constraining itself to have a convex shape.

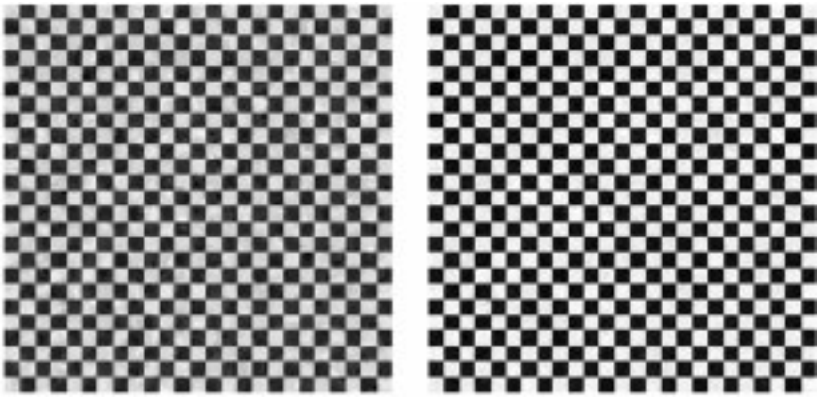
## VII. Summary

Tomasi and Manduchi proposed the bilateral filter in 1998 [14] as an appealing algorithm for noise removal from images. As such, this algorithm was posed as an alternative to locally



(a) (b)

Fig. 7. (a) Piece-wise constant test image and (b) its noisy version.



(a) (b)

Fig. 8. (a) RE and (b) the bilateral filtering reconstruction results.

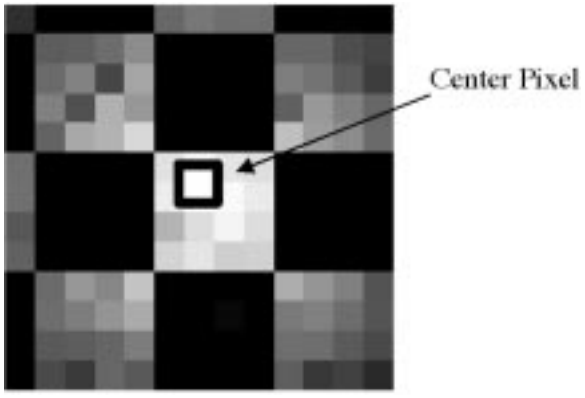



Fig. 9. Bilinear filter weights at the point [48, 48].

adaptive well-known algorithms such as the anisotropic diffusion (AD), the weighted least-squares (WLS), and the robust estimation (RE) techniques. However, no theoretical background supporting the bilateral filter was suggested. In this paper we filled this theoretical gap and proposed such a theory for explaining the origin of the bilateral filter. We have shown that the Bayesian approach is also in the core of the bilateral filter, just as it has been for the AD, WLS, and RE. We have also shown how this new insight can serve for improving the bilateral filter and extend its use for other applications.

## ACKNOWLEDGMENT

The author would like to thank Dr. R. Kimmel from the Computer Science Department, The Technion—Israel Institute of Technology, Haifa, for his substantial help in forming this paper and his numerous helpful suggestions.

## REFERENCES

[1] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 12, pp. 629–639, July 1990.

[2] Z. Lin and Q. Shi, "An anisotropic diffusion PDE for noise reduction and thin edge preservation," in *Proc. 10th Int. Conf. Image Analysis and Processing, IEEE Computer Society*, Los Alamitos, CA, 1999, pp. 102–107.
[3] N. Sochen, R. Kimmel, and R. Malladi, "A geometrical framework for low level vision," *IEEE Trans. Image Processing*, vol. 7, pp. 310–318, Feb. 1998.
[4] J. Weickert, *Anisotropic Diffusion in Image Processing*, ser. ECMI Series. Stuttgart, Germany: Teubner, 1998.
[5] G. Sapiro and D. L. Ringach, "Anisotropic diffusion of color images," *Proc. SPIE*, vol. 471, pp. 382–2657, 1996.
[6] R. L. Lagendijk, J. Biemond, and D. E. Boekee, "Regularized iterative image restoration with ringing reduction," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, pp. 1874–1887, Dec. 1988.
[7] M. E. Zervakis, "Nonlinear image restoration techniques," Ph.D. dissertation, Univ. Toronto, Toronto, ON, Canada, 1990.
[8] M. J. Black and G. Sapiro, "Edges as outliers: Anisotropic smoothing using local image statistics," in *Scale-Space Theories in Computer Vision, Second International Conference, Scale-Space '99. Proceedings (Lecture Notes in Computer Science vol. 1682)*, Berlin, Germany, 1999, pp. 259–270.
[9] R. Lagendijk and J. Biemond, *Iterative Identification and Restoration of Images*. Norwell, MA: Kluwer, 1991.
[10] A. K. Jain, *Fundamentals of Digital Image Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.
[11] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Commun. Pure Appl. Math.*, vol. 42, pp. 577–685, 1989.
[12] M. J. Black, G. Sapiro, D. H. Marimont, and D. Heeger, "Robust anisotropic diffusion," *IEEE Trans. Image Processing*, vol. 7, pp. 421–432, Mar. 1998.
[13] M. Black and A. Rangarajan, "On the unification of line processes, outlier rejection, and robust statistics with applications in early vision," *Int. J. Comput. Vis.*, vol. 19, pp. 57–92, 1996.
[14] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. 6th Int. Conf. Computer Vision*, New Delhi, India, 1998, pp. 839–846.
[15] D. Bertsekas, *Nonlinear Programming*. Belmont, MA: Athena, 1995.
[16] D. Luenberger, *Linear and Non-Linear Programming*, 2nd ed. Reading, MA: Addison-Wesley, 1987.
[17] J. Weickert, B. M. T. H. Romeny, and M. A. Viergever, "Efficient and reliable schemes for nonlinear diffusion filtering," *IEEE Trans. Image Processing*, vol. 7, pp. 398–410, Mar. 1998.
[18] T. F. Chan, S. Osher, and J. Shen, "The digital TV filter and nonlinear denoising," *IEEE Trans. Image Processing*, vol. 10, pp. 231–241, Feb. 2001.
[19] G. Ramponi, "The rational filter for image smoothing," *IEEE Signal Processing Lett.*, vol. 3, pp. 63–65, Mar. 1996.
[20] G. Ramponi and C. Moloney, "Smoothing speckled images using an adaptive rational operator," *IEEE Signal Processing Lett.*, vol. 4, pp. 68–71, Mar. 1997.
[21] M. Nagao and T. Matasuyama, "Edge preserving smoothing," *Comput. Graph. Image Process.*, vol. 9, pp. 394–407, 1979.
[22] L. Alvarez, R. Deriche, and F. Santana, "Recursivity and PDE's in image processing," in *Proc. 15th Int. Conf. Pattern Recognition (ICIP)*, Barcelona, Spain, Sept. 3–7, 2000.
[23] N. Sochen, R. Kimmel, and A. M. Bruckstein, "Diffusions and confusions in signal and image processing," *J. Math. Imag. Vis.*, vol. 14, no. 3, pp. 195–209, May 2001.

**Michael Elad** received the B.Sc., M.Sc., and D.Sc. degrees from the Electrical Engineering Department, The Technion—Israel Institute of Technology, Haifa, in 1986, 1988, and 1996, respectively.

From 1988 to 1993, he was an Officer in the Israeli Air-Force. From 1997 to 1999, he was a Researcher at the Hewlett-Packard Laboratories-Israel (HPL-I). During 2000–2001, he led the Research Division of Jigami Corporation, Israel. From 1997 to 2001, he was a Lecturer with the Electrical Engineering Department, The Technion. He is now with the Computer Science Department (SCCM Program), Stanford University, Stanford, CA, as a Research Associate. His research interests include inverse problems and numerical algorithms in the areas of signal processing, image processing, and computer vision. He worked on super-resolution reconstruction of images, motion estimation, nonlinear filtering, sparse representations for signals, target detection in images, and image compression. His current research is focused on numerical eigenvalue problems and efficient representation of signals.

Dr. Elad received the Best Lecturer Award from The Technion in 2000 and 2001.