

A New Iterated Two-Band Diffusion Equation: Theory and Its Application

Arthur Chun-Chieh Shih, Hong-Yuan Mark Liao, *Senior Member, IEEE*, and Chun-Shien Lu

Abstract—In this paper, we propose an iterated two-band filtering method to solve the selective image smoothing problem. We prove that a discrete computation step in an iterated nonlinear diffusion-based filtering algorithm is equivalent to a sequence of operations, including decomposition, regularization, and then reconstruction, in the proposed two-band filtering scheme. To correctly separate the high frequency components from the low frequency ones in the decomposition process, we adopt a dyadic wavelet-based approximation scheme. In the regularization process, we use a diffusivity function as a guide to retain useful data and suppress noises. Finally, the signal of the next stage, which is a “smoother” version of the signal at the previous stage, can be computed by reconstructing the decomposed low frequency component and the regularized high frequency component. Based on the proposed scheme, the smoothing operation can be applied to the correct targets. Experimental results show that our new approach is really efficient in noise removing.

Index Terms—Nonlinear filtering, selective image smoothing, wavelets.

I. INTRODUCTION

THE use of diffusion/conduction equations to guide image processing procedures such as smoothing has recently attracted a lot of attention [1]–[10]. In the literature [11]–[14], researchers have reported algorithms which can be used to make signals convoluted with Gaussians of varying widths so that linear diffusion-based smoothing can be executed. The goal of such algorithms is to smooth a target signal isotropically by triggering an iterated process. These algorithms repeatedly process the high-frequency components, including the data and noises of a signal, without any discrimination. Under these circumstances, the noises in the target signal can be smoothed out, but the useful data which are supposed to be retained are smoothed out as well. Nonlinear diffusion-based smoothing [1] was developed to solve the above mentioned problem. The first nonlinear diffusion filter was proposed by Perona and Malik [1]. The basic concept behind their work is to replace Gaussian smoothing with directional diffusion, which preserves useful data. In their work, a nonlinear diffusion function was embedded in a nonlinear PDE (Partial Differential Equation), which automatically guided the smoothing task. By computing the solution of the nonlinear PDE, noise reduction and edge enhancement can be

accomplished simultaneously. However, finding a closed-form solution of a nonlinear PDE is always a difficult task. Most of the time, one can only find approximate solutions by using numerical methods [1], [9]. It is understandable that different numerical methods may result in different outcomes. In recent years, many researchers have searched for an appropriate numerical method to solve the above mentioned nonlinear PDE [1], [2], [7], [9].

The major problem of a nonlinear diffusion-based process is that it is difficult to correctly separate the high frequency components from the low frequency ones. In other words, if one wishes to deal with the high frequency components only, some irrelevant low frequency components may also be included. Therefore, when one applies nonlinear diffusion to smooth the high frequency components (which should include noises and edge data only), it is actually applied to both the high frequency components and some of the low frequency ones. In this paper, we propose a new method to solve the selective image smoothing problem based on an iterated two-band filtering process. In our scheme, the aforementioned nonlinear diffusion-based filtering process is converted into a two-band filtering process. We shall prove that a discrete computation step in a nonlinear diffusion process can be decomposed into three stages: combining decomposition, regularization, and reconstruction where the first stage and the last stage are equivalent to a completely two-band filtering system.

Since we wish to correctly separate the high frequency components from the low frequency ones, an efficient and accurate tool is needed. Therefore, we propose to use a dyadic wavelet-based approach to perform the decomposition task. Since wavelets have good features for approximating function variables or operators, they can produce a better result for separating high frequency components from low frequency ones. Under these circumstances, the signal of the current stage is decomposed into high frequency components and low frequency ones by using a dyadic wavelet approximated high-pass filter and low-pass filter, respectively. In the regularization process, the high frequency components are “regularized” by means of a diffusivity function. The objective of this process is to use the diffusivity function as a guide to retain useful data and suppress noises. Finally, the signal of the next stage can be computed by reconstructing the decomposed low frequency components and the regularized high frequency components. Basically, one can consider the signal of the next stage as a “smoother” signal of the current stage. Therefore, we call the proposed iterated two-band filtering process a “modified” nonlinear diffusion-based filtering process. We shall prove that the proposed iterated system always satisfies the so-called

Manuscript received November 17, 2000; revised October 29, 2002. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Scott T. Acton.

The authors are with the Institute of Information Science, Academia Sinica, Taipei 11529, Taiwan, R.O.C. (e-mail: arthur@iis.sinica.edu.tw; liao@iis.sinica.edu.tw).

Digital Object Identifier 10.1109/TIP.2003.809017

maximum-minimum principle [1], [9] no matter what kind of wavelet basis is used. One may argue that in [15], the authors' wavelet shrinkage method was only applied once instead of many times. The reason why we made our approach an iterated one is that it is difficult to find a clear-cut boundary with only one pass. For some complicated images, the boundary between high frequency components and low frequency ones is fuzzy. Our iterated scheme can proceed in an asymptotic manner such that the high frequency components and the low frequency ones can be gradually and accurately separated. Based on our scheme, the smoothing operation could then be applied to the correct targets. Experimental results show that our new approach is really efficient in noise removing.

The rest of this paper is organized as follows. In Section II, some basic definitions and properties of the wavelet theory will be introduced. In Section III, we shall discuss the relationship between a nonlinear diffusion process and a two-band filtering process. Then, a dyadic wavelet-based diffusion equation will be derived in Section IV. Experimental results obtained using both 1-D synthetic signals and a 2-D real image will be reported in Section V. Finally, concluding remarks will be made in Section VI.

II. SOME BASIC PROPERTIES OF WAVELET THEORY

Some notations which will be used throughout the paper will be presented. All the functions considered here are in $L^2(R)$, the space of square-integrable functions over real numbers. For $f(x)$ and $g(x)$ in $L(R)$, the inner product $\langle f(x), g(x) \rangle = \int f(x)g(x)dx$ and the convolution $f(x) * g(x) = \int f(u)g(x-u)du$. The Fourier transform of $f(x)$ is denoted by $\hat{f}(\omega) = \int f(x)e^{-i\omega x}dx$.

A wavelet is a function $\psi(x) \in L^2(R)$ satisfying the admissibility condition [16]–[18]

$$C_\psi = \int \frac{|\hat{\psi}(\omega)|^2}{|\omega|} d\omega < \infty.$$

A continuous wavelet transform of a signal $f(x)$, with wavelet $\psi(x)$, is

$$\begin{aligned} W_s f(x) &= \left\langle f(u), \frac{1}{s} \psi \left(\frac{x-u}{s} \right) \right\rangle = \langle f(u), \psi_s(x-u) \rangle \\ &= f * \psi_s(x) \end{aligned} \quad (1)$$

where $(1/s)\psi(x/s)$ is denoted by $\psi_s(x)$. From the Fourier transform of $f(x)$ is

$$\hat{W}_s f(\omega) = \hat{f}(\omega) \hat{\psi}(s\omega).$$

The dyadic wavelet transform of $f(x)$ is

$$W_{2^j} f(x) = f * \psi_{2^j}(x) \quad (2)$$

for scale parameter $s = 2^j$ for $j \in Z$. Furthermore, if there exist two strictly positive constant ϱ_1 and ϱ_2 such that

$$\varrho_1 \leq \sum_{j=-\infty}^{\infty} |\hat{\psi}(2^j \omega)| \leq \varrho_2 \quad (3)$$

then $\psi(x)$ has a dual wavelet $\xi(x)$, by which the signal $f(x)$ can be reconstructed as follows:

$$f(x) = \sum_{j=-\infty}^{\infty} W_{2^j} f(x) * \xi_{2^j}(x). \quad (4)$$

From the Fourier transform of (4), we have

$$\sum_{j=-\infty}^{\infty} \hat{\psi}(2^j \omega) \hat{\xi}(2^j \omega) = 1. \quad (5)$$

Let $\phi(x)$ denote the scaling function whose Fourier transform is an aggregation of $\hat{\psi}(2^j \omega)$ and $\hat{\xi}(2^j \omega)$ for $j \geq 1$, i.e.,

$$|\hat{\phi}(\omega)|^2 = \sum_{j=1}^{\infty} \hat{\psi}(2^j \omega) \hat{\xi}(2^j \omega). \quad (6)$$

A direct implication of (6) is

$$|\hat{\phi}(2^{j-1} \omega)|^2 - |\hat{\phi}(2^j \omega)|^2 = \hat{\psi}(2^j \omega) \hat{\xi}(2^j \omega). \quad (7)$$

Equation (7) illustrates the relations among $\phi(x)$, $\psi(x)$, and $\xi(x)$ at different scales. If the scaling function $\phi(x)$ and the wavelet function $\psi(x)$ are determined beforehand, based on (7), then the corresponding dual wavelet $\xi(x)$ can be derived.

Let S_{2^j} be the smoothing operator with respect to the scaling function $\phi(x)$ defined by

$$S_{2^j} f(x) = f(x) * \phi_{2^j}(x). \quad (8)$$

Suppose that $f(x)$ is defined at the finest resolution, i.e., $f(x) = S_{2^0} f(x)$. By using the scaling function $\phi(x)$ and the wavelet function $\psi(x)$, a signal $f(x)$ can be decomposed into different components $\{S_{2^j} f(x)\}_{j=1,2,\dots}$ and $\{W_{2^j} f(x)\}_{j=1,2,\dots}$. $S_{2^j} f(x)$ is an approximation view of $f(x)$ at scale 2^j and $W_{2^j} f(x)$ represents the difference between $S_{2^{j-1}} f(x)$ and $S_{2^j} f(x)$. If $\phi(x)$ and $\psi(x)$ are considered, respectively, as the low-pass and high-pass filter, then $S_{2^j}(x)$ is equivalent to the low frequency component of $f(x)$ at scale 2^j and $W_{2^j} f(x)$ is the high frequency component, i.e., the edge information of $f(x)$ at scale 2^j . Therefore, by using the information of $\{S_{2^j} f(x)\}_{j=1,2,\dots}$ and $\{W_{2^j} f(x)\}_{j=1,2,\dots}$, the original function can be reconstructed. The reconstruction process is a recursive procedure based on (7). Multiplying both sides of (7) by $\hat{f}(\omega)$ and taking the inverse Fourier transform, we obtain

$$S_{2^{j-1}} f(x) = S_{2^j} f(x) * \bar{\phi}_{2^j}(x) + W_{2^j} f(x) * \xi_{2^j}(x) \quad (9)$$

where $\bar{\phi}(x) = \phi(x)(-x)$. By (9), $S_{2^{j-1}} f(x)$ can be recovered from $S_{2^j} f(x)$ and $W_{2^j} f(x)$. Applying the above process recursively, the original signal, $f(x) = S_{2^0} f(x)$, can be reconstructed.

In the derivation of the desired wavelet, both the quadratic and cubic spline functions are usually chosen as scaling functions. The scaling function, $\phi(x)$, in the frequency domain can be as follows [19]:

$$\hat{\phi}(x) = e^{-i\beta_1 x} \prod_{m=1}^{\infty} H(2^{-m} \omega) \quad (10)$$

where $H(\omega)$ is a 2π periodic differentiable function satisfying

$$|H(\omega)|^2 + |H(\omega + \pi)|^2 \leq 1 \text{ and } H(0) = 1. \quad (11)$$

The parameter, $0 \leq \beta_1 < 1$, is a sampling shift. With the form in (10), we have

$$\hat{\phi}(2\omega) = e^{-i\beta_1\omega} H(\omega) \hat{\phi}(\omega). \quad (12)$$

As in [19], we may impose further that the Fourier transform of the desired wavelet $\psi(x)$ and its dual wavelet $\xi(x)$ are

$$\hat{\psi}(2\omega) = e^{-i\beta_2\omega} G(\omega) \hat{\phi}(\omega) \quad (13)$$

and

$$\hat{\xi}(2\omega) = e^{-i\beta_2\omega} K(\omega) \hat{\phi}(\omega) \quad (14)$$

where $G(\omega)$ and $K(\omega)$ are two 2π periodic differentiable functions, and $0 \leq \beta_2 < 1$ is another sampling shift. Plugging (12)–(14) into (7), because $\phi(x)$ is symmetric with respect to 0, we have

$$|\hat{H}(\omega)|^2 + \hat{G}(\omega) \hat{K}(\omega) = 1. \quad (15)$$

The three 2π periodic functions $H(\omega)$, $G(\omega)$, and $K(\omega)$ can be expanded respectively as follows:

$$H(\omega) = \sum_{n=-\infty}^{\infty} h(n) e^{-in\omega}, \quad G(\omega) = \sum_{n=-\infty}^{\infty} g(n) e^{-in\omega}, \text{ and} \\ K(\omega) = \sum_{n=-\infty}^{\infty} k(n) e^{-in\omega} \quad (16)$$

where $\{h(n)\}_{n \in \mathbb{Z}}$, $\{g(n)\}_{n \in \mathbb{Z}}$, and $\{k(n)\}_{n \in \mathbb{Z}}$ are the three real sequences satisfying

$$\sum_{n=-\infty}^{\infty} |h(n)|^2 < \infty, \quad \sum_{n=-\infty}^{\infty} |g(n)|^2 < \infty, \text{ and} \\ \sum_{n=-\infty}^{\infty} |k(n)|^2 < \infty. \quad (17)$$

Based on the discrete wavelet transform, an original discrete signal $f(x)$ can be decomposed into $S_{2^1} f(x)$ and $W_{2^1} f(x)$, that is,

$$W_{2^1} f(x) = (f * G)(x) \text{ and } S_{2^1} f(x) = (f * H)(x). \quad (18)$$

The inverse wavelet transform algorithm can construct the decomposed signal $f(x)$ from $S_{2^1} f(x)$ and $W_{2^1} f(x)$

$$f(x) = (S_{2^1} f * \tilde{H})(x) + (W_{2^1} f * K)(x) \quad (19)$$

where \tilde{H} is the conjugate filter of H .

In the 2-D case, the two wavelets $\psi^1(x, y)$ and $\psi^2(x, y)$ can be characterized by the three 1-D discrete filters G , K , and L . The filters H , G , K , and L satisfy the following relation [19]:

$$\hat{G}(\omega_x) \hat{K}(\omega_x) \hat{L}(\omega_y) + \hat{G}(\omega_y) \hat{K}(\omega_y) \hat{L}(\omega_x) \\ = 1 - |\hat{H}(\omega_x)|^2 |\hat{H}(\omega_y)|^2, \quad (20)$$

$$\hat{G}(\omega_i) \hat{K}(\omega_i) + |\hat{H}(\omega_i)|^2 = 1, \text{ and}$$

$$\hat{L}(\omega_i) = \frac{1 + |\hat{H}(\omega_i)|^2}{2} \quad (21)$$

where ω_i is ω_x or ω_y .

In the 2-D discrete wavelet transform, an original discrete image $f(x, y)$ can be decomposed into $S_{2^1} f(x, y)$, $W_{2^1}^1 f(x, y)$, and $W_{2^1}^2 f(x, y)$, where

$$S_{2^1} f(x, y) = (f * HH)(x, y), \\ W_{2^1}^1 f(x, y) = (f * GD)(x, y)$$

and

$$W_{2^1}^2 f(x, y) = (f * DG)(x, y). \quad (22)$$

$A * BC$ denotes the separable convolution of rows and columns, respectively, of the image A with 1-D filters B and C . The 1-D filter D is a delta function.

The inverse 2-D discrete wavelet transform algorithm reconstructs $f(x, y)$ from $S_{2^1} f(x, y)$, $W_{2^1}^1 f(x, y)$ and $W_{2^1}^2 f(x, y)$

$$f(x, y) = (S_{2^1} f * \tilde{H} \tilde{H})(x, y) + (W_{2^1}^1 f * KL)(x, y) \\ + (W_{2^1}^2 f * LK)(x, y). \quad (23)$$

More details about the dyadic wavelet transform can be found in [16] and [19].

III. RELATIONSHIP BETWEEN A NONLINEAR DIFFUSION PROCESS AND A TWO-BAND FILTERING SYSTEM

Nonlinear diffusion PDEs have often been used to perform selective image smoothing in the past [1]–[9]. In this paper, we shall point out an interesting finding with regard to the diffusion-based system and a two-band filter bank system. Since we can find a direct mapping between the above two systems, problems which can be solved by using a nonlinear diffusion-based system can also be solved by using a two-band filtering system. In this section, we shall relate the two systems to one another. Furthermore, we will point out why the diffusion equation can be mapped into the two-band filtering format.

A nonlinear diffusion PDE in the 1-D case can be written as follows [20]:

$$\frac{\partial f_t(x)}{\partial t} = \frac{\partial}{\partial x} \left[c_t(x) \frac{\partial f_t(x)}{\partial x} \right] \quad (24)$$

where $f_t(x)$ is the status of point x at time t and $c_t(x)$ is its corresponding diffusivity. Thus, the status of point x at time $t + dt$ can be derived by

$$f_{t+dt}(x) = f_t(x) + \frac{\partial f_t(x)}{\partial t} \cdot dt \\ = f_t(x) + \frac{\partial}{\partial x} \left(c_t(x) \frac{\partial f_t(x)}{\partial x} \right) \cdot dt. \quad (25)$$

In the discrete case, dt is assumed to be Δt . Usually, we can set Δt to be 1. Therefore, we have the discrete formula of (25) as follows:

$$f_{t+1}(x) \approx f_t(x) + \frac{d}{dx} \left(c_t(x) \frac{df_t(x)}{dx} \right). \quad (26)$$

Without losing generality in a discrete case, we can assume

$$f_{t+1}(x) = f_t(x) + \frac{d}{dx} \left(c_t(x) \frac{df_t(x)}{dx} \right). \quad (27)$$

Equation (27) can be written as follows:

$$\begin{aligned} f_{t+1}(x) &= f_t(x) + \frac{d}{dx} \left((1 - (1 - c_t(x))) \frac{df_t(x)}{dx} \right) \\ &= f_t(x) + \frac{d}{dx} \left(\frac{df_t(x)}{dx} - p_t(x) \cdot \frac{df_t(x)}{dx} \right) \\ &= f_t(x) + \frac{d^2 f_t(x)}{dx^2} - \frac{d}{dx} \left(p_t(x) \cdot \frac{df_t(x)}{dx} \right) \end{aligned} \quad (28)$$

where $p_t(x) = 1 - c_t(x)$. Using the Fourier transform, the response of (28) in the ω_x domain can be written as follows:

$$\begin{aligned} \hat{f}_{t+1}(\omega_x) &= (1 - \omega_x^2) \hat{f}_t(\omega_x) \\ &\quad - (j\omega_x) \left(\frac{1}{2\pi} (\hat{p}_t(\omega_x) * (j\omega_x \hat{f}_t(\omega_x))) \right). \end{aligned} \quad (29)$$

Moreover, we can assume $A_1(x)$ and $A_2(x)$ are two functions of x and their Fourier transforms are $\hat{A}_1(\omega_x)$ and $\hat{A}_2(\omega_x)$, respectively, that can make $\hat{A}_1(\omega_x) \cdot \hat{A}_2(\omega_x) = 1 - \omega_x^2$. Besides, let $\hat{B}(\omega_x) = j\omega_x$, $\hat{D}(\omega_x) = -j\omega_x$, and $\hat{G}_t(\omega_x) = (1/2\pi) \hat{p}_t(\omega_x)$. So, we could have (29) becomes

$$\begin{aligned} \hat{f}_{t+1}(\omega_x) &= \hat{A}_2(\omega_x) \cdot \hat{A}_1(\omega_x) \cdot \hat{f}_t(\omega_x) \\ &\quad + \hat{D}(\omega_x) \cdot \left(\hat{G}_t(\omega_x) * \left(\hat{B}(\omega_x) \cdot \hat{f}_t(\omega_x) \right) \right). \end{aligned} \quad (30)$$

The block diagram illustrating how (30) operates is shown in Fig. 1(a). We have found an interesting connection among $\hat{A}_1(\omega_x)$, $\hat{A}_2(\omega_x)$, $\hat{B}(\omega_x)$, and $\hat{D}(\omega_x)$, i.e.,

$$\hat{A}_1(\omega_x) \cdot \hat{A}_2(\omega_x) + \hat{B}(\omega_x) \cdot \hat{D}(\omega_x) = 1. \quad (31)$$

Applying the inverse Fourier transform, (30) becomes

$$f_{t+1}(x) = (f_t * A_1)(x) * A_2(x) + [p_t(x) \cdot (f_t * B)(x)] * D(x). \quad (32)$$

Fig. 1(b) shows the block diagram which indicates how (32) operates. In Fig. 1(b), the input signal $f_t(x)$ is decomposed into $f_t^L(x) = (f * A_1)(x)$ and $f_t^H(x) = (f * B)(x)$ by using filters A_1 and B , respectively. Then, the component $f_t^H(x)$ is multiplied by $p_t(x)$ and becomes $f_t^{Hd}(x) = f_t^H(x) \cdot p_t(x)$. Finally, the state of point x at time $t + 1$ can be calculated by convolving $f_t^L(x)$ and $f_t^{Hd}(x)$ with A_2 and D , respectively. From the derivation shown in (30) and (32), one can find that a nonlinear diffusion process can be converted into an iterated, filter bank-based decomposition and reconstruction process. It is interesting that one can use a set of filter-based operators to find the solution of (27) directly rather than by applying any traditional numerical method. It is obvious that the solution obtained using the proposed method is dependent on the characteristics of the filter bank (A_1 , A_2 , B , and D).

However, in most conventional approaches, the filters used in a nonlinear diffusion process are nearly linear so that they cannot correctly separate the high frequency components from the low frequency component ones. In order to improve the aforementioned problem, we propose to use a set of dyadic wavelet-based filters to approximate the function variables used

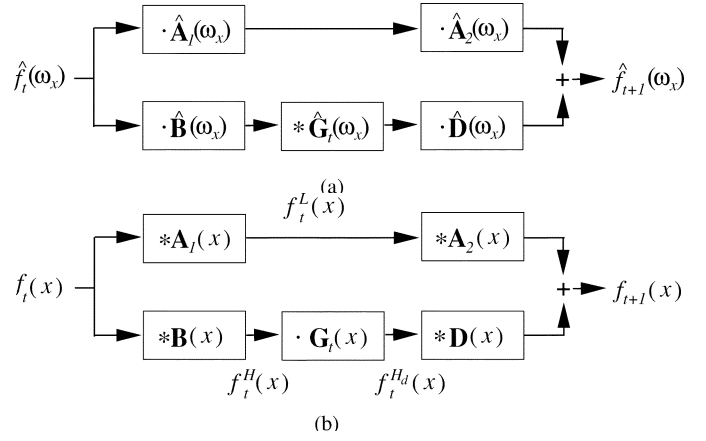


Fig. 1. (a) The block diagram showing how (30) operates. (b) The block diagram showing how (30) operates.

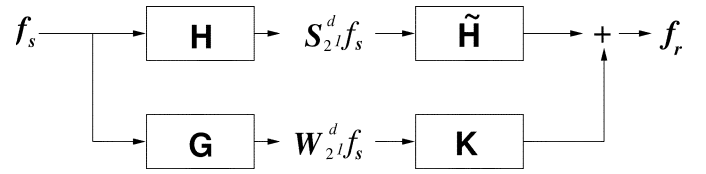


Fig. 2. General two-band filter system.

in a diffusion equation and to then solve the equation. Using the dyadic wavelet-based filters, one can separate the high frequency signals (including edges and noises) from the low frequency ones more accurately.

IV. DYADIC WAVELET-BASED DIFFUSION EQUATION

In general, a diffusion-based smoothing process repeatedly decomposes the original signal (or the low-frequency component) into a high-frequency component and a low-frequency one. The high-frequency component generated at each resolution is then processed by a nonlinear operator which can decrease the magnitude of the noises and retain that of the useful data. The signal reconstruction process can be achieved by adding the processed high-frequency component and the nonprocessed low-frequency component at the lowest resolution to form the low-frequency component of the second lowest resolution. This low-frequency component is then combined with the processed high-frequency component at the same resolution to form the low-frequency part of the resolution one level up. The process is repeated until the original signal is reconstructed. Since the noises are selectively smoothed at each resolution, a new signal that contains less noise can be obtained.

A mechanism that can characterize the above mentioned process is a two-band filtering system as shown in Fig. 2. A two-band filtering system can decompose a signal into a high-frequency component (G in Fig. 2) and a low-frequency component (H in Fig. 2). Based on the concept described in the previous paragraph, a nonlinear regularization function can be applied to the high-frequency component (i.e., G) as shown in Fig. 2. Fig. 3 shows the realization of the above mentioned concept. The dyadic wavelet transformed high-frequency component, $W_{2,l}^d f_n$, is regularized by a nonlinear function,

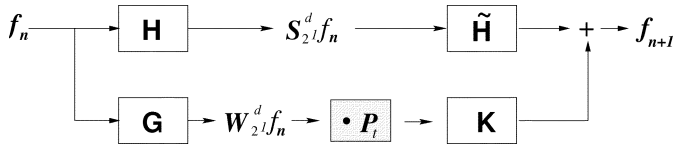


Fig. 3. Regularization module.

P_t , before it is used as a component for reconstruction. The nonlinear regularization function used can be any nonlinear function with appropriate characteristics, for example, the nonlinear diffusivity function used in [1]. The objective of regularization is to retain the values of the high-frequency components that correspond to those having larger magnitudes and to suppress those having smaller magnitudes. Experience tells us that the high-frequency components always contain edges and noises simultaneously. Since the magnitude of the intensity gradient of edges is usually much larger than that of noises, we shall use the nonlinear diffusivity function adopted in [1] to distinguish between them. The reconstruction process will be used to reconstruct a new signal by integrating the unprocessed low-frequency component and the regularized high-frequency component, iteratively.

A. Wavelet-Based Conduction Equation

In what follows, we shall show how to use the dyadic wavelet transform [19] to realize the mechanism illustrated in Fig. 3. Let $f_s(x) \in L^2(R)$. At scale 2^1 , the dyadic wavelet transform decomposes $f_s(x)$ into $S_{2^1}^d f_s(x)$ and $W_{2^1}^d f_s(x)$ [19] as follows:

$$S_{2^1}^d f_s(x) = (f_s * H)(x) \quad (33)$$

and

$$W_{2^1}^d f_s(x) = (f_s * G)(x) \quad (34)$$

where H and G are the low-pass filter and the high-pass filter at scale 2^1 , respectively. The inverse wavelet transform can reconstruct the original signal $f_s(x)$ from $S_{2^1}^d f_s$ and $W_{2^1}^d f_s$ by means of the procedure shown below

$$f_s(x) = \left(S_{2^1}^d f_s * \tilde{H} \right)(x) + \left(W_{2^1}^d f_s * K \right)(x) \quad (35)$$

where H and \tilde{H} are conjugate at scale 2^1 and K is the high-pass filter used for reconstruction at scale 2^1 . Let $p(x)$ be a continuous function depending on the absolute value of $W_{2^1}^d f_s(x)$ and $0 \leq p(x) \leq 1$. Similar to the derivation of (32), an approximately reconstructed function $f_{NR}(x)$ of $f_s(x)$ can be defined as follows:

$$f_{NR}(x) = \left(S_{2^1}^d f_s * \tilde{H} \right)(x) + [p(x) \cdot W_{2^1}^d f_s(x)] * K(x). \quad (36)$$

A new iterated function based on (36) can be written as follows:

$$f_{n+1}(x) = \left(S_{2^1}^d f_n * \tilde{H} \right)(x) + [p_n(x) \cdot W_{2^1}^d f_n(x)] * K(x). \quad (37)$$

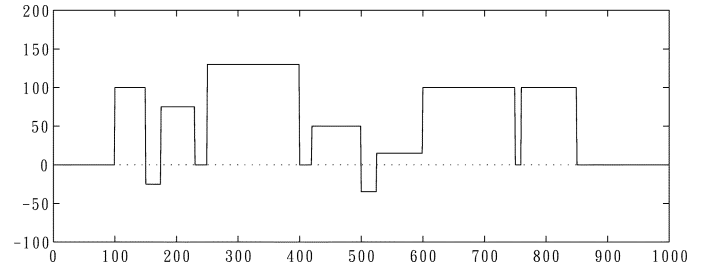


Fig. 4. Original test signal.

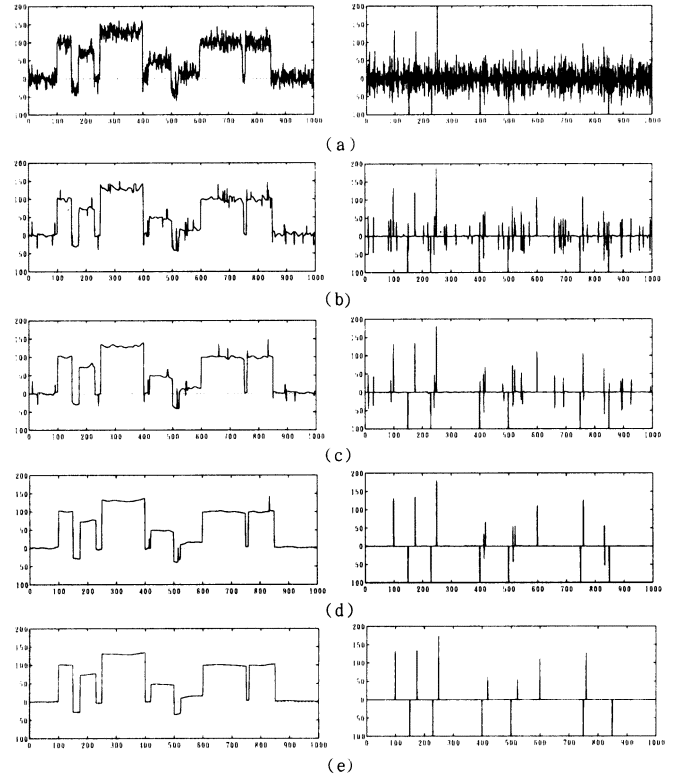


Fig. 5. (a) The first column, $f_n(x)$, shows the signal shown in Fig. 4 with added white noise and variance $\sigma = 10$. The last column shows $W_{2^1} f_n(x)$. (b)–(e) $f_n(x)$ and $W_{2^1} f_n(x)$ are obtained by applying our approach after the 30th, 60th, 120th, and 300th iteration, respectively.

In order to reduce the complexity of computing $f_{n+1}(x)$, a new dyadic wavelet-based conduction equation is derived. From (37), we can further derive the following:

$$f_{n+1}(x) = (f_n * H * \tilde{H})(x) + \{ [p_n(x) \cdot W_{2^1}^d f_n(x)] * K(x) \}. \quad (38)$$

Taking the Fourier transform of $f_{n+1}(x)$, we obtain

$$\begin{aligned} \hat{f}_{n+1}(\omega_x) &= \hat{f}_n(\omega_x) \cdot \hat{H}(\omega_x) \cdot \hat{\tilde{H}}(\omega_x) \\ &\quad + \mathcal{F} \{ p_n(x) \cdot W_{2^1}^d f_n(x) \} \cdot \hat{K}(\omega_x) \\ &= \hat{f}_n(\omega_x) \cdot |\hat{H}(\omega_x)|^2 + \hat{M}_n(\omega_x) \cdot \hat{K}(\omega_x) \end{aligned} \quad (39)$$

where $\hat{M}_n(\omega_x)$ is the Fourier transform of $p_n(x) \cdot W_{2^1}^d f_n(x)$, that is $\hat{M}_n(\omega_x) = \mathcal{F} \{ p_n(x) \cdot W_{2^1}^d f_n(x) \}$. Because $|\hat{H}(\omega_x)|^2 + \hat{G}(\omega_x) \hat{K}(\omega_x) = 1$, (39) can be rewritten as follows:

$$\begin{aligned} \hat{f}_{n+1}(\omega_x) &= \hat{f}_n(\omega_x) [1 - \hat{G}(\omega_x) \cdot \hat{K}(\omega_x)] + \hat{M}_n(\omega_x) \cdot \hat{K}(\omega) \\ &= \hat{f}_n(\omega_x) + [\hat{M}_n(\omega_x) - \hat{f}_n(\omega_x) \cdot \hat{G}(\omega_x)] \\ &\quad \cdot \hat{K}(\omega_x). \end{aligned} \quad (40)$$

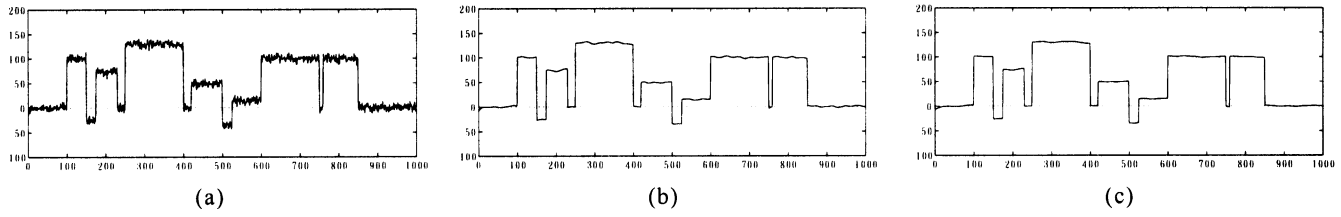


Fig. 6. (a) Signal shown in Fig. 4 corrupted by white noise with variance $\sigma = 5$. (b) Result obtained by applying our approach after 30 iterations. (c) Result obtained after 120 iterations.

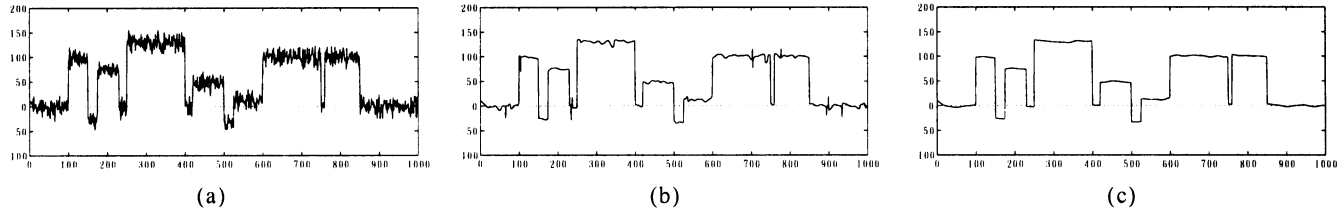


Fig. 7. (a) Signal shown in Fig. 4 corrupted by white noise with variance $\sigma = 10$. (b) Result obtained by applying our approach after 30 iterations. (c) Result obtained after 120 iterations.

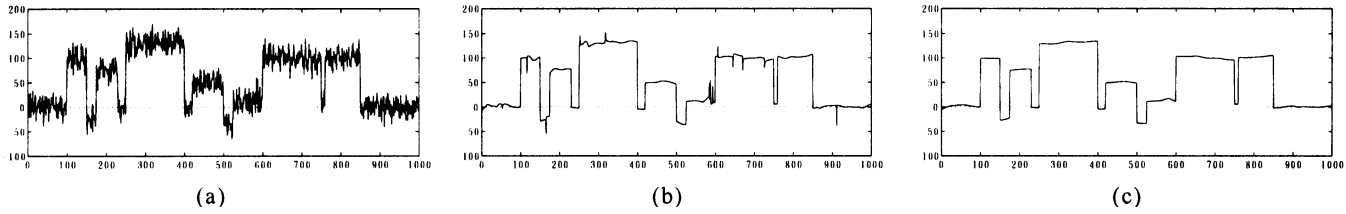


Fig. 8. (a) Signal shown in Fig. 4 corrupted by white noise with variance $\sigma = 15$. (b) Result obtained by applying our approach after 120 iterations. (c) Result obtained after 300 iterations.

Taking the inverse Fourier transform of (40), we obtain

$$\begin{aligned} f_{n+1}(x) &= f_n(x) \\ &\quad + [p_n(x) \cdot W_{2^1}^d f_n(x) - (f_n * G)(x)] * K(x) \\ &= f_n(x) + \{[p_n(x) - 1] \cdot W_{2^1}^d f_n(x)\} * K(x) \\ &= f_n(x) - \{c_n(x) \cdot W_{2^1}^d f_n(x)\} * K(x) \end{aligned} \quad (41)$$

where $c_n(x) = 1 - p_n(x)$ and $c_n(x)$ is the traditional diffusivity function [1].

When $c_n(x)$ is set to 1, (41) becomes

$$f_{n+1}(x) = f_n(x) - (W_{2^1}^d f_n * K)(x). \quad (42)$$

Again, we take the Fourier transform of (42) and obtain

$$\begin{aligned} \hat{f}_{n+1}(\omega_x) &= \hat{f}_n(\omega_x) - \hat{f}_n(\omega_x) \cdot \hat{G}(\omega_x) \cdot \hat{K}(\omega_x) \\ &= \hat{f}_n(\omega_x) [\hat{H}(\omega_x)]^2. \end{aligned} \quad (43)$$

Applying the inverse Fourier transform to both sides of (43), we obtain

$$f_{n+1}(x) = \left(S_{2^1}^d f_n * \tilde{H} \right)(x). \quad (44)$$

The high-frequency components of $f_n(x)$ can be completely removed in this case because H is a low-pass filter. On the other hand, if $c_n(x)$ is set to zero, then (41) becomes $f_{n+1}(x) = f_n(x)$. In the following, we will prove that (41) satisfies the so-called maximum and minimum principle [1]. It should be noted that the discrete maximum-minimum principle is a very restrictive stability criterion (more restrictive than the von Neumann stability principle) [9].

At scale 2^1 , $f_{n+1}(x)$ can be written as

$$f_{n+1}(x) = \left(S_{2^1}^d f_{n+1} * \tilde{H} \right)(x) + (W_{2^1} f_{n+1} * K)(x). \quad (45)$$

Because (45) is equal to (37), we have

$$\begin{aligned} W_{2^1} f_{n+1}(x) &= \left(S_{2^1}^d f_{n+1} * \tilde{H} * G \right)(x) \\ &\quad + (W_{2^1} f_{n+1} * K * G)(x) \\ &= \left(S_{2^1}^d f_n * \tilde{H} * G \right)(x) \\ &\quad + [p_n(x) \cdot W_{2^1} f_n(x)] \\ &\quad * K(x) * G(x). \end{aligned} \quad (46)$$

Because $(H * G)(x) = 0$, we have

$$(W_{2^1} f_{n+1} * K * G)(x) = [p_n(x) \cdot W_{2^1} f_n(x)] * K(x) * G(x).$$

Finally, we can derive

$$\begin{aligned} |W_{2^1} f_{n+1}(x)| &= |p_n(x) \cdot W_{2^1} f_n(x)| \\ &= p_n(x) \cdot |W_{2^1} f_n(x)| \leq |(W_{2^1} f_n x)|. \end{aligned} \quad (47)$$

Equation (47) shows that the magnitude of $W_{2^1} f_{n+1}(x)$ is always less than or equal to the magnitude of $W_{2^1} f_n(x)$. That is, there is no local extrema generated by $f_n(x)$ when the iteration increases from n to $n + 1$. As a consequence, (41) satisfies the maximum and minimum principle [1] when an arbitrary wavelet-based filter is used. But please note that (47) does not give any prediction what kind of the results we could obtain. In Section V-A, we will show that the different results are obtained by using different wavelets. But all the results satisfy (41).

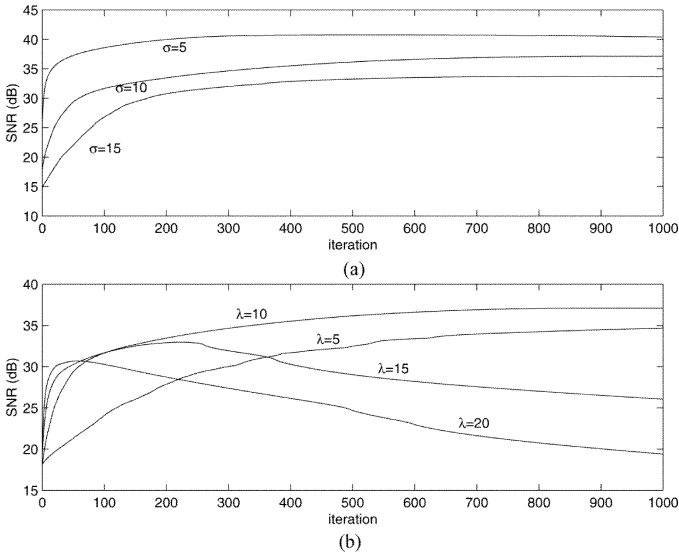


Fig. 9. (a) SNR values obtained by adding different noise levels with a fixed λ ($\lambda = 10$). (b) SNR values obtained using different λ values in the conductivity function with a noise level, $\rho = 10$.

B. Two-Dimensional Wavelet-Based Conduction Equation

In a 2-D wavelet transform at scale 2^1 , $f(x, y)$ can be decomposed into $S_{2^1}^d(x, y)$, $W_{2^1}^1 f(x, y)$, and $W_{2^1}^2 f(x, y)$, i.e.,

$$f(x, y) = \left(S_{2^1}^d f * \tilde{H}\tilde{H} \right) (x, y) + \left(W_{2^1}^1 f * KL \right) (x, y) + \left(W_{2^1}^2 f * LK \right) (x, y)$$

where $A*(B, C)$ denotes the separable convolution of rows and columns, respectively, of an image A with 1-D filters B and C [19]. Thus,

$$\begin{aligned} f_{n+1}(x, y) &= \left(S_{2^1}^d f_n * \tilde{H}\tilde{H} \right) (x, y) \\ &+ [p_n^1(x, y) \cdot W_{2^1}^1 f_n(x, y)] * KL(x, y) \\ &+ [p_n^2(x, y) \cdot W_{2^1}^2 f_n(x, y)] * LK(x, y) \end{aligned} \quad (48)$$

where $p_n^1(x, y)$ and $p_n^2(x, y)$ are continuous functions and are dependent of the absolute values of $W_{2^1}^1 f_n(x, y)$ and $W_{2^1}^2 f_n(x, y)$, respectively. Generally, we set $0 \leq p_n^1(x, y)$, $p_n^2(x, y) \leq 1$. Taking the Fourier transform of (48), we obtain

$$\begin{aligned} \hat{f}_{n+1}(\omega_x, \omega_y) &= [\hat{f}(\omega_x, \omega_y) \cdot \widehat{H}\widehat{H}(\omega_x, \omega_y)] \cdot \widehat{H}\widehat{H}(\omega_x, \omega_y) \\ &+ \hat{M}^1(\omega_x, \omega_y) \cdot \widehat{KL}(\omega_x, \omega_y) \\ &+ \hat{M}^2(\omega_x, \omega_y) \cdot \widehat{LK}(\omega_x, \omega_y) \end{aligned} \quad (49)$$

where $\hat{M}^1(\omega_x, \omega_y) = \mathcal{F}\{p_n^1(x, y) \cdot W_n^1 f(x, y)\}$ and $\hat{M}^2(\omega_x, \omega_y) = \mathcal{F}\{p_n^2(x, y) \cdot W_n^2 f(x, y)\}$. Because $\widehat{H}\widehat{H}(\omega_x, \omega_y) = \hat{H}(\omega_x)\hat{H}(\omega_y)$, $\widehat{H}\widehat{H}(\omega_x, \omega_y) = \hat{H}(\omega_x)\hat{H}(\omega_y)$, $\widehat{KL}(\omega_x, \omega_y) = \hat{K}(\omega_x)\hat{L}(\omega_y)$, and $\widehat{LK}(\omega_x, \omega_y) = \hat{L}(\omega_x)\hat{K}(\omega_y)$ [19]. The relationship among the filters $H(\omega_x)$, $G(\omega_x)$, $K(\omega_x)$, and $L(\omega_y)$ is $G(\omega_x)K(\omega_x)L(\omega_y) + G(\omega_y)K(\omega_y)L(\omega_x) = 1 - |H(\omega_x)|^2|H(\omega_y)|^2$. Thus, (49) can be rewritten as follows:

$$\begin{aligned} \hat{f}_{n+1}(\omega_x, \omega_y) &= \hat{f}_n(\omega_x, \omega_y) \cdot (|H(\omega_x)|^2|H(\omega_y)|^2) \\ &+ \hat{M}^1(\omega_x, \omega_y) \cdot \hat{K}(\omega_x) \cdot \hat{L}(\omega_y) \\ &+ \hat{M}^2(\omega_x, \omega_y) \cdot \hat{L}(\omega_x) \cdot \hat{K}(\omega_y) \end{aligned}$$

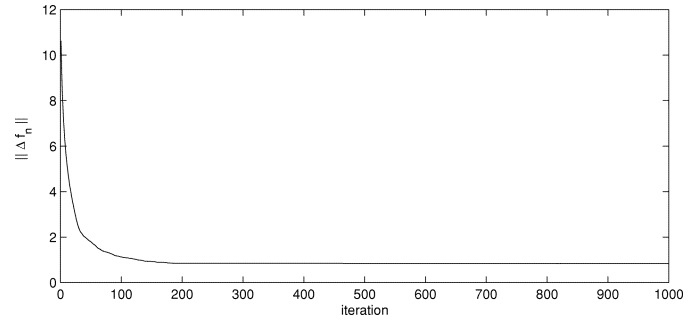


Fig. 10. Average difference $\|\Delta f_n\|$ (53) within 1000 iterations.

$$\begin{aligned} &= \hat{f}_n(\omega_x, \omega_y) \\ &\quad \times [1 - \hat{G}(\omega_x)\hat{K}(\omega_x)\hat{L}(\omega_y) \\ &\quad \quad - \hat{G}(\omega_y)\hat{K}(\omega_y)\hat{L}(\omega_x)] \\ &\quad + \hat{M}^1(\omega_x, \omega_y) \cdot \hat{K}(\omega_x) \cdot \hat{L}(\omega_y) \\ &\quad + \hat{M}^2(\omega_x, \omega_y) \cdot \hat{L}(\omega_x) \cdot \hat{K}(\omega_y) \\ &= \hat{f}(\omega_x, \omega_y) \\ &\quad + [\hat{M}^1(\omega_x, \omega_y) - \hat{f}_n(\omega_x, \omega_y) \cdot \hat{G}(\omega_x)] \\ &\quad \quad \cdot \hat{K}(\omega_x)\hat{L}(\omega_y) \\ &\quad + [\hat{M}^2(\omega_x, \omega_y) - \hat{f}_n(\omega_x, \omega_y) \cdot \hat{G}(\omega_y)] \\ &\quad \quad \cdot \hat{K}(\omega_y)\hat{L}(\omega_x). \end{aligned} \quad (50)$$

Taking the inverse Fourier transform of (50), we obtain

$$\begin{aligned} f_{n+1}(x, y) &= f_n(x, y) \\ &+ [p_n^1(x, y) \cdot W_{2^1}^1 f_n(x, y) \\ &\quad - W_{2^1}^1 f_n(x, y)] * KL(x, y) \\ &+ [p_n^2(x, y) \cdot W_{2^1}^2 f_n(x, y) - W_{2^1}^2 f_n(x, y)] \\ &\quad * LK(x, y) \\ &= f_n(x, y) \\ &\quad - [c_n^1(x, y) \cdot W_{2^1}^1 f_n(x, y)] * KL(x, y) \\ &\quad - [c_n^2(x, y) \cdot W_{2^1}^2 f_n(x, y)] * LK(x, y) \end{aligned} \quad (51)$$

where $c_n^1(x, y) = 1 - p_n^1(x, y)$ and $c_n^2(x, y) = 1 - p_n^2(x, y)$ are the traditional diffusivity functions [1] along the x direction and the y direction, respectively. We call (51) a 2-D dyadic wavelet-based diffusion equation. Following the same procedure described in Section IV-A, we can prove that (51) also satisfies the maximum and minimum principle.

V. EXPERIMENTAL RESULTS

In the first group of experiments, our method was used to apply the denoising process to a synthetic noisy signal. A set of experimental results obtained by using a 1-D synthetic signal will be given here. In the second group of experiments, several real images were used to test the effectiveness of our approach. In order to ensure that all the experiments were conducted under fair conditions, the diffusivity function used was the one proposed by Weickert [9], where

$$c(s_n) = 1 - e^{-3.315/(s_n/\lambda)^4}. \quad (52)$$

S_n in (52) is the wavelet transform of $f_n(x)$ at scale 2^1 , i.e., $W_{2^1} f_n(x)$, and λ is a constant.

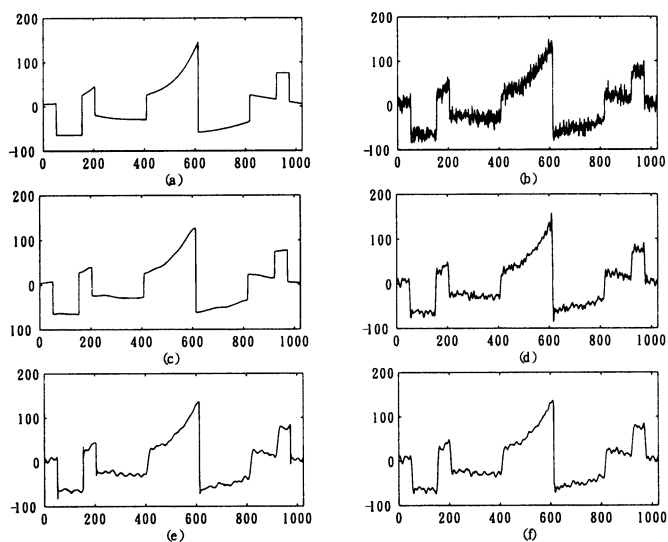


Fig. 11. (a) Original signal. (b) Noisy signal (SNR = 14.23 dB). (c) Result obtained by using the Haar wavelet after 300 iterations (27.59 dB). (d) Result obtained by using a Villasenor's biorthogonal wavelet (using the function `MakeBSFil-ter('Villasenor',1)` in `WaveLab`) after 300 iterations (17.00 dB). (e) Result obtained by using a Daubechies orthogonal wavelet with parameter 4 (using the function `MakeONFilter('Daubechies',4)` in `WaveLab`) after 300 iterations (21.01 dB). (f) Result obtained by using a Daubechies orthogonal wavelet with parameter 6 (using the function `MakeONFilter('Daubechies',6)` in `WaveLab`) after 300 iterations (18.77 dB).

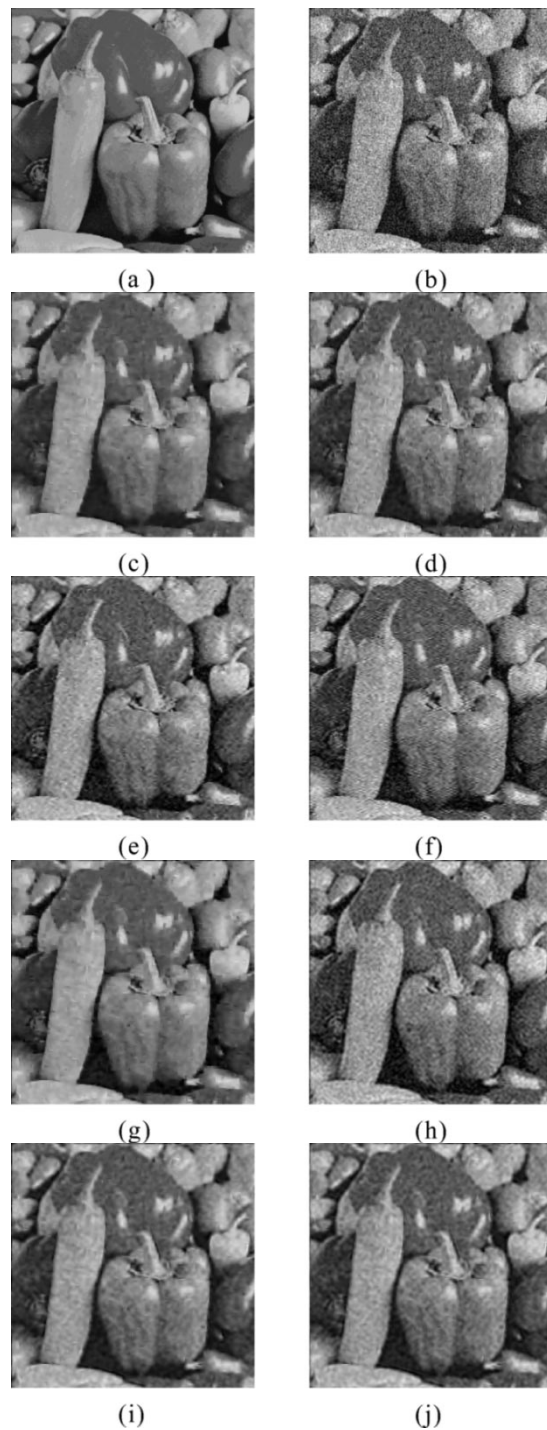


Fig. 13. (a) Original image. (b) Noisy image (SNR = 29 dB). (c) Result obtained by using a spline biorthogonal filter—CDF(1,1) after 10 iterations (SNR = 32.34 dB). (d) Result obtained by using a spline biorthogonal filter—CDF(1,3) after 10 iterations (SNR = 32.17 dB). (e) Result obtained by using a Villasenor's biorthogonal filter—`MakeBSFilter('Villasenor',1)`—after 10 iterations (SNR = 31.90 dB). (f) Result obtained by using a Villasenor's biorthogonal filter—`MakeBSFilter('Villasenor',2)`—after 10 iterations (SNR = 29.06 dB). (g) Result obtained by using the Haar orthogonal filter—`MakeONFilter('Haar')`—after 10 iterations (SNR = 32.34 dB). (h) Result obtained by using the Beylkin orthogonal filter—`MakeONFilter('Beylkin')`—after 10 iterations (SNR = 30.78 dB). (i) Result obtained by using a Daubechies orthogonal filter—`MakeONFilter('Daubechies',4)`—after 10 iterations (SNR = 32.20 dB). (j) Result obtained by using a Daubechies orthogonal filter—`MakeONFilter('Daubechies',6)`—after 10 iterations (SNR = 32.11 dB).

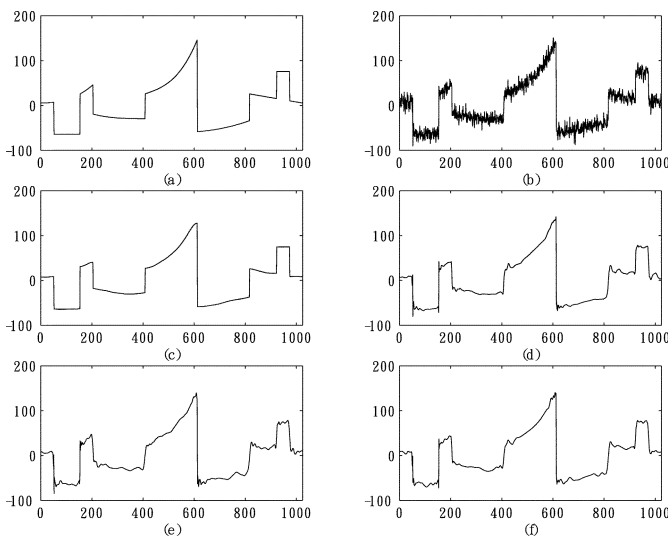


Fig. 12. (a) Original signal. (b) Noisy signal (SNR = 14.30 dB). (c) Result obtained using our approach after 300 iterations (28.83 dB). (d) Result obtained using a wavelet soft threshold (using the function `Thresh-Wave(noises,'s',1,10,5,4,MakeONFilter('Haar'))` in `WaveLab`) (21.70 dB). (e) Result obtained using a wavelet hard threshold (using the function `ThreshWave(noises,'h',1,10,5,4,MakeONFilter('Daubechies',6))` in `WaveLab`) (20.58 dB). (f) Result obtained using a wavelet hard threshold (using the function `ThreshWave(noises,'h',1,9,5,4,MakeONFilter('Symmlet',4))` in `WaveLab`) (19.56 dB).

A. Results Obtained Using the Proposed Approach

Fig. 4 shows a 1-D synthetic signal which is composed of some step functions of different magnitudes. The white noises with variance $\sigma = 10$ were added to the original signal, and the synthesized signal is shown on the left side of Fig. 5(a). After dyadic wavelet transformation, the high frequency component

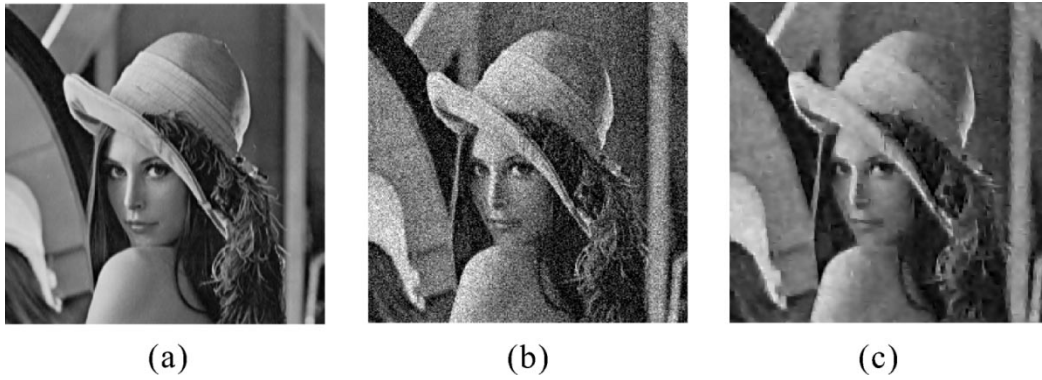


Fig. 14. (a) Original image: Lenna. (b) Noisy image (28.88 dB). (c) Result obtained by using Haar wavelet after 10 iterations (32.94 dB).

at 2^1 scale was as shown on the right side of Fig. 5(a). From the high frequency component shown in Fig. 5(a), it is not difficult to find that the edges and noises are mixed together, and that a linear approach definitely cannot separate them. The left side of Fig. 5(b)–(e) show the denoised results obtained by applying our algorithm 30, 60, 120, and 300 iterations, respectively. On the right side of Fig. 5(b)–(e), we show the corresponding low frequency components and high frequency components of the denoised results. From the right column of Fig. 5, we find that the true edges were always preserved, but that the number of noises decreased when the iteration number was increased. It is worth noting that when the iteration number reached 300, only edges were preserved.

Figs. 6(a), 7(a), and 8(a) display the signal shown in Fig. 4 after it was corrupted by white noises with variance $\sigma = 5, 10,$ and $15,$ respectively. From the results shown in Figs. 6(b) and (c), 7(b) and (c) and 8(b) and (c) it is not difficult to find that our approach worked very well in terms of removing noises.

Fig. 9(a) shows the curves based on results obtained, under different noise levels ($\sigma = 5, 10,$ and $15,$) for SNR values at different iteration numbers. All the curves were obtained with $\lambda = 10$ ((52)). It is worth noting that when the noise level was increased to $\sigma = 15,$ the SNR value still reached 30 dB after 200 iterations. Fig. 9(b) shows another set of curves based on results obtained, under different λ values ($\lambda = 5, 10, 15,$ and $20,$) for SNR values at different iteration numbers. All the curves shown in Fig. 9(b) were obtained based on a fixed noise level ($\sigma = 10$). From the four curves, it is obvious that when λ was over 10, the SNR values gradually decreased after 100 iterations.

For studying how our approach converges, we define the average difference between $W_{2^1} f_{n+1}(x)$ and $W_{2^1} f_n(x)$ as follows:

$$\|\Delta f_n\| = \frac{1}{\#x} \sum_x |W_{2^1} f_{n+1}(x) - W_{2^1} f_n(x)| \quad (53)$$

where $\#x$ is the total number of sampling points. Fig. 10 shows the average difference $\|\Delta f_n\|$ after each iteration within 1000 iterations. We can find the values of $\|\Delta f_n\|$ almost keep around ones but zeroes after 200 iterations. That means some of the coefficients $W_{2^1} f_n(x)$ are always modified before reconstruction at each iteration.

The above results were all obtained by using the quadratic spline wavelet. But it is not difficult to find, our approach also can be implemented by using biorthogonal and orthogonal wavelet bases. For fair comparison our approach by using different wavelet filters and with the other approach, we used the subroutines of the popular *MATLAB* library—*WaveLab802* (<http://www-stat.stanford.edu/wavelab/>) to implement our approach and used their purposed test signal to conduct the following results. An example program is shown in Appendix A. Fig. 11 shows another set of results obtained using different wavelet bases including the Haar wavelet, a Villasenor's biorthogonal wavelet and the Daubechies wavelet with different vanishing moments. Fig. 11(a) shows an original signal (obtained by using the function **makesignal** ('Piece-Polynomial',1024)). Fig. 11(b) shows a noisy version of (a) which SNR is 14.23 dB. Fig. 11(c)–(f) show the results obtained using the Haar wavelet bases, the Villasenor's biorthogonal wavelet(**makeBSFilter** ('Villasenor',1)), the Haar bases(**makeONFilter**('Haar')) and the Daubechies wavelet bases (**makeONFilter**('Daubechies',6)), respectively, after 300 iterations. It is apparent that the result obtained by using the Haar wavelet bases has the highest SNR value than the other ones.

B. Comparison With Other Nonlinear Smoothing Algorithms

In this section, we shall compare our results with those obtained using the well known nonlinear smoothing algorithms based on Donoho-Johnstone's approach [15]. Fig. 12(b) the test signal [shown in (a)] corrupted by noise and its SNR related to (a) is 14.30 dB. Fig. 12(c) shows the result obtained by using our approach after 300 iterations. Fig. 12(d) and (f) show the results obtained by applying the function **ThreshWave** in *WAVELAB* with different parameters. We try to find the best result by using different sets of parameters and wavelet bases. Eventually, three better results are obtained by using three different sets of parameters. It is obvious that our approach achieve a better result compared to their approach.

C. Results Obtained Using Our Approach in the 2-D Case

Fig. 13(a) shows a real test image: pepper, and Fig. 13(b) is a noisy version of (a) ($SNR = 29$ dB). By using our approach, Fig. 13(c)–(j) show the results obtained by using different

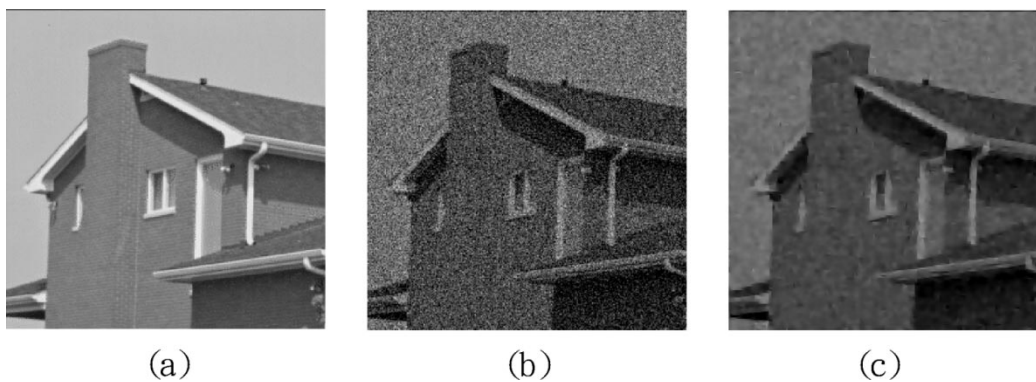


Fig. 15. (a) Original image: House. (b) Noisy image (26.76 dB). (c) Result obtained by using Haar wavelet after 10 iterations (32.04 dB).

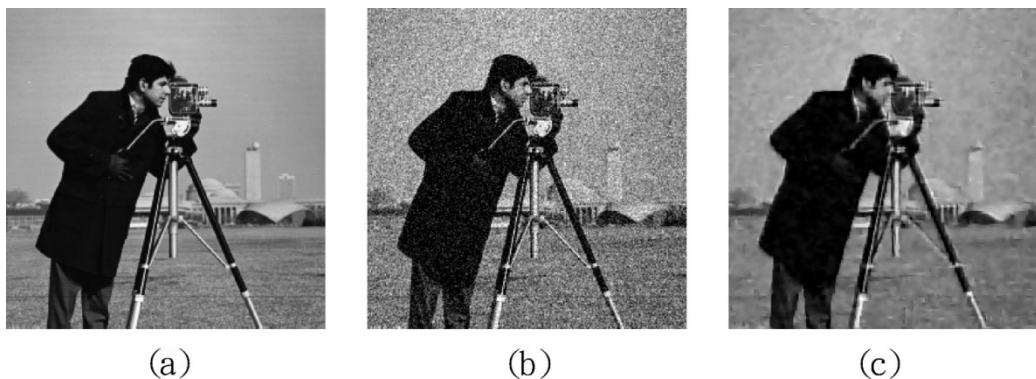


Fig. 16. (a) Original image: Photoman. (b) Noisy image (24.32 dB). (c) Result obtained by using Haar wavelet after 10 iterations (31.37 dB).

wavelet bases including biorthogonal and orthogonal wavelets after 10 iterations. The best result is also obtained by using the Harr wavelet (SNR = 32.34 dB). Figs. 14–16 are more real test images, noisy images, and denoised results obtained by applying our approach.

VI. CONCLUSIONS

In this paper, we have proposed an iterated two-band filtering method to solve the selective image smoothing problem. We have proven that a discrete computation step in an iterated non-linear diffusion-based filtering algorithm can be converted into a sequence of operations, including decomposition, regularization, and then reconstruction, in our proposed two-band filtering scheme. To correctly separate the high frequency components from the low frequency ones, we use a dyadic-wavelet based approximation scheme. In the regularization process, we use a diffusivity function to guide the processes for noise suppression and retaining useful data. Experimental results show that our approach can be used to achieve superb selective image smoothing results.

APPENDIX

This is an example code of our 1-D approach implemented by *MATLAB* language. Before running the program in *MATLAB*, please download and install *WaveLab* routines (<http://www-stat.stanford.edu/~wavelab>).

```

sigN = 1000;
sigtype = 'Piece-Polynomial';
sigrho = 7;
sigorg = MakeSignal(sigtype, sigN);
[sigx, sigy] = NoiseMaker(sigorg, sigrho);
% Inputs
% y: input noisy 1-D signal
% k: smoothing parameter
% no: iteration number
% Output
% xh: result
y = sigy;
k = 1;
no = 300;
tic
par = [1 1];
[qmf, dqmf] = MakeBSFilter('CDF', par);
xh = y(:)';
L = length(xh);
for i = 1: no
    wcoef = symmiconv(MirrorSymmFilt(dqmf), xh);
    ind = find(wcoef == 0);
    tmp = (wcoef(ind) ./ k).^4;
    c = exp(-3.315 ./ tmp);
    wcoef(ind) = (1 - c) .* wcoef(ind);
    xh = xh - symmaconv(MirrorSymmFilt(qmf), wcoef);
end
toc

```

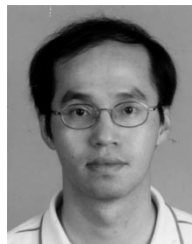
```
subplot(3,1,1); plot(sig_x);
subplot(3,1,2); plot(sig_y);
subplot(3,1,3); plot(xh);
```

ACKNOWLEDGMENT

The authors thank Mr. M.-S. Su for suggesting an efficient programming of the codes as shown in Appendix A.

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] J. Weickert, "A review of nonlinear diffusion filtering," in *Scale-Space Theory for Computer Vision*, B. H. Romeny, Ed. New York: Springer, 1997, vol. 1252, Lecture Notes in Computer Science, pp. 3–28.
- [3] F. Catté, P. Lions, J.-M. Morel, and T. Coll, "Image selective smoothing and edge detection by nonlinear diffusion," *SIAM J. Numer. Anal.*, vol. 29, no. 1, pp. 182–193, Feb. 1992.
- [4] Y. L. You, W. Xu, A. Tannenbaum, and M. Kaveh, "Behavioral analysis of anisotropic diffusion in image processing," *IEEE Trans. Image Processing*, vol. 5, pp. 1539–1553, 1996.
- [5] L. Alvarez, P. L. Lions, and J. K. Morel, "Image selective smoothing and edge detection by nonlinear diffusion II," *SIAM J. Numer. Anal.*, vol. 29, pp. 845–866, 1992.
- [6] L. Alvarez, F. Guichard, P. L. Lions, and J. K. Morel, "Axioms and fundamental equation of image processing," *Archive Ration. Mech. Anal.*, vol. 123, pp. 200–257, 1993.
- [7] M. Nitzberg and T. Shiota, "Nonlinear image filtering with edge and corner enhance," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 14, pp. 826–833, Aug. 1992.
- [8] G. Gerig, O. Kübler, R. Kikinis, and F. A. Jolesz, "Nonlinear anisotropic filtering of MRI data," *IEEE Trans. Med. Imag.*, vol. 11, pp. 221–232, 1992.
- [9] J. Weickert, B. M. ter Romeny, and M. A. Viergever, "Efficient and reliable schemes for nonlinear diffusion filtering," *IEEE Trans. Image Processing*, vol. 7, pp. 398–410, Mar. 1998.
- [10] C. J. Sze, H. Y. M. Liao, and K. C. Fan, "A new image flux conduction model and its application to selective image smoothing," *IEEE Trans. Image Processing*, vol. 10, pp. 296–306, Jan. 2001.
- [11] T. Iijima, "Basic theory of pattern normalization (for the case of a typical one-dimensional pattern)" (in Japanese), *Bull. Electrotech. Lab.*, vol. 26, pp. 368–388, 1962.
- [12] A. P. Witkin, "Scale-space filtering," in *Proc. Int. Joint Conf. Artificial Intelligence*, Karlsruhe, Aug. 1983, pp. 1019–1022.
- [13] J. J. Koenderink, "The structure of images," *Biol. Cybern.*, vol. 50, pp. 363–370, 1984.
- [14] T. Lindeberg, *Scale-Space Theory in Computer Vision*. Amsterdam, The Netherlands: Kluwer, 1994.
- [15] D. L. Donoho and I. M. Johnstone, "Ideal Spatial Adaptation by Wavelet Shrinkage," Dept. Statist., Stanford Univ., Stanford, CA, 1992.
- [16] S. Mallat, *A Wavelet Tour of signal Processing*, MA, USA: Academic, 1998.
- [17] C. K. Chui, *An Introduction to Wavelets*. New York: Academic Press, 1992.
- [18] Y. Meyer, *Wavelets and Operators*. Cambridge, U.K.: Cambridge Univ. Press, 1992.
- [19] S. Mallat and S. Zhong, "Characterization of signal from multi-scale edges," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 14, pp. 710–732, July 1992.
- [20] J. P. Holman, *Heat Transfer*, 8th ed. New York: McGraw-Hill, 1997.



Arthur Chun-Chieh Shih was born in Taipei, Taiwan, R.O.C., on September 26, 1966. He received the B.S. degree in electrical engineering from the Chinese Culture University, Taipei, in 1992, the M.S. degree also in electrical engineering from National Chung-Cheng University, Chiayi, Taiwan, in 1994 and a Ph.D. degree in computer science and information engineering from National Central University, Chung-Li, Taiwan, in 1998.

From October 1998 to July 2002, he worked for the Institute of Information Science, Academia Sinica, Taiwan, and the Department of Ecology and Evolution, the University of Chicago, Chicago, IL, as a Postdoctoral Fellow. He joined the Institute of Information Science, Academia Sinica, as an Assistant Research Fellow in July 2002. His current research interests include multimedia processing and bioinformatics.



Hong-Yuan Mark Liao received the B.S. degree in physics from National Tsing-Hua University, Hsin-Chu, Taiwan, R.O.C., in 1981, and the M.S. and Ph.D. degrees in electrical engineering from Northwestern University, Evanston, IL, in 1985 and 1990, respectively.

He was a Research Associate in the Computer Vision and Image Processing Laboratory at Northwestern University during 1990–1991. In July 1991, he joined the Institute of Information Science, Academia Sinica, Taiwan, as an assistant research fellow. He was promoted to Associate Research Fellow and then Research Fellow in 1995 and 1998, respectively. From August 1997 to July 2000, he served as the Deputy Director of the institute. Currently, he is the Acting Director of Institute of Applied Science and Engineering Research, Academia Sinica, Taiwan. His current research interests include multimedia signal processing, wavelet-based image analysis, content-based multimedia retrieval, and multimedia protection. He is on the editorial boards of the *International Journal of Visual Communication and Image Representation*; *Acta Automatica Sinica*; and the *Tamkang Journal of Science and Engineering*. He is the managing editor of the *Journal of Information Science and Engineering*.

Dr. Liao was the recipient of the Young Investigators' Award of Academia Sinica in 1998; the Excellent Paper Award of the Image Processing and Pattern Recognition Society of Taiwan in 1998 and 2000. He served as the program chair of the International Symposium on Multimedia Information Processing (ISMIP'97) and the program co-chair of the second IEEE Pacific-Rim Conference on Multimedia (2001). He will serve as a conference co-chair of the IEEE International Conference on Multimedia and Exposition at 2004. He also served on the program committees of several international and local conferences. Dr. Liao was on the Editorial Board of the IEEE TRANSACTIONS ON MULTIMEDIA (1998–2001).



Chun-Shien Lu received the Ph.D. degree in electrical engineering from National Cheng-Kung University, Tainan, Taiwan, R.O.C., in 1998. From October 1998 to July 2002, he was with Institute of Information Science, Academia Sinica, Taiwan, as a postdoctoral fellow. Since August 2002, he has been an Assistant Research Fellow at the same institute. His current research interests include multimedia security and communications, multimedia signal processing, wireless multimedia, and time-frequency analysis of signals and images.

Dr. Lu was the recipient of the Best Paper Award of the Image Processing and Pattern Recognition Society of Taiwan in 2000 for his work on digital watermarking and the Paper Award of the society in 1997, 1999, and 2001. He organized and chaired a special session on multimedia security in the 2nd and the 3rd IEEE Pacific-Rim Conference on Multimedia, respectively (2001–2002). He holds one R.O.C. patent.

Dr. Lu is a member of the IEEE Signal Processing Society.