Adaptive Smoothing via Contextual and Local Discontinuities

Ke Chen, *Senior Member*, *IEEE*

Abstract—A novel adaptive smoothing approach is proposed for noise removal and feature preservation where two distinct measures are simultaneously adopted to detect discontinuities in an image. Inhomogeneity underlying an image is employed as a multiscale measure to detect contextual discontinuities for feature preservation and control of the smoothing speed, while local spatial gradient is used for detection of variable local discontinuities during smoothing. Unlike previous adaptive smoothing approaches, two discontinuity measures are combined in our algorithm for synergy in preserving nontrivial features, which leads to a constrained anisotropic diffusion process that inhomogeneity offers intrinsic constraints for selective smoothing. Thanks to the use of intrinsic constraints, our smoothing scheme is insensitive to termination times and the resultant images in a wide range of iterations are applicable to achieve nearly identical results for various early vision tasks. Our algorithm is formally analyzed and related to anisotropic diffusion. Comparative results indicate that our algorithm yields favorable smoothing results, and its application in extraction of hydrographic objects demonstrates its usefulness as a tool for early vision.

Index Terms—Adaptive smoothing, inhomogeneity, spatial gradient, noise removal, feature preservation, anisotropic diffusion, local scale control, multiple scales, the termination problem, extraction of hydrographic objects.

1 Introduction

In order to understand an image, a multistage processing is necessary in human visual system [1]. The early stage of a vision system plays an important role in visual information processing, which critically determines the ultimate interpretation of an image. On the other hand, real images are complex since there are large numbers of diversified objects in the physical world and the imaging process is inevitably corrupted by noise from various sources. In computer vision, smoothness is a generic assumption that many types of images have approximately piece-wise constant gray levels, which characterizes the coherence and homogeneity of an object. Therefore, smoothing is viewed as a general tool in the low-level of machine vision [2], [3], [4], [5], [6], [7], [8]. In general, the purpose of smoothing is of twofold: noise is eliminated to facilitate further processing, and the features irrelevant to a given problem are ruled out to reduce the complexity for subsequent processing.

Smoothing techniques have been extensively studied in the computer vision community. In general, smoothing algorithms are roughly classified into two categories: linear and nonlinear smoothing [7]. For linear smoothing, local operators are uniformly applied to an image to form the output intensity of a pixel from a weighted summation of input intensities of its neighboring pixels. A major shortcoming of linear smoothing is that important features, e.g., boundaries between different regions, are blurred after smoothing. Nonlinear smoothing has been developed to overcome the shortcoming, which tends to preserve important features along with noise removal during smoothing.

• *The author is with the School of Informatics, The University of Manchester, PO Box 88, Sackville St., Manchester M60 1QD, United Kingdom. E-mail: Ke.Chen@manchester.ac.uk.*

Manuscript received 3 Nov. 2003; revised 2 Sept. 2004; accepted 15 Dec. 2004; published online 11 Aug. 2005.
Recommended for acceptance by A. Del Bimbo.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number TPAMI-0351-1103.

Adaptive smoothing is a class of typical nonlinear smoothing techniques that have been studied for many years [2], [5], [9], [10], [11], [12], [13], [14], [15]. The general idea underlying adaptive smoothing is to adapt pixel intensities to the local attributes of an image on the basis of discontinuity measures. Inspired by Perona and Malik's seminal work [16], there have been exploratory efforts in connecting adaptive smoothing with systems of nonlinear partial differential equations [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31], [32], which is argued to provide a theoretical foundation for adaptive smoothing [32], [33]. In this framework, adaptive smoothing is simulated as a nonuniform diffusion process, where noise is removed and important features are preserved. Adaptive smoothing has been applied to many computer vision tasks ranging from medical image processing to postprocessing of noisy data (for a review, see [34]).

In adaptive smoothing, a discontinuity measure critically determines its performance. For most adaptive smoothing approaches, spatial gradient is widely used as a discontinuity measure. Owing to overlocality, however, it is inadequate to detect significant discontinuities from a noisy image, which causes an adaptive smoothing algorithm to yield poor results. For instance, Fig. 2c illustrates a spatial gradient map of the image in Fig. 2b which is a noisy version of an image shown in Fig. 2a. It is observed from Fig. 2c that discontinuities of boundaries of the smallest circle (cf., Fig. 2a) cannot be distinguished from substantial noise. Thus, some nontrivial features would fail to survive independently during smoothing if the spatial gradient is solely used for detecting discontinuities; in other words, they either cosurvive or are smoothed out simultaneously with substantial noise while noise is removed from the image. To tackle the problem, different efforts have been made in the last decade. These efforts in the context of adaptive smoothing can be classified into two categories; one is spatial regularization and the other is alternative discontinuity measures. In spatial regularization, a Gaussian kernel is employed for removing noise prior to gradient estimation [17], [18], [32], [33].

Although noise may be removed by spatial regularization, it may cause important discontinuities to be blurred since a Gaussian operation has the linear smoothing effect. On the other hand, miscellaneous discontinuity measures have been proposed for robustness. A direct extension from spatial gradient is introducing measures of higher order differentiations and/or taking feature directions into consideration [6], [13], [14], [19], [24], [31], [32]. To some extent, however, these techniques incur significantly higher computational complexity since, in those approaches, a discontinuity measure has to be used in each iteration of adaptive smoothing. In addition, other discontinuity measures have also been developed to overcome overlocality, a weakness of spatial gradient. Unlike spatial gradient, some image-dependent morphological or structural information has been considered in these measures so that even in the presence of substantial noise important features still can be detected in a proper scale [15], [35]. However, finding a proper scale becomes a nontrivial problem for practical applications. Recently, Saha et al. proposed an algorithm that automatically determines local scales everywhere for a given image [36], which leads to an alternative image-dependent discontinuity measure. As a result, such a measure has been applied to adaptive smoothing and yields impressive performance in medical image processing [29]. Although their measure outperforms spatial gradient and other discontinuity measures, our empirical studies indicate that it might still fail to accurately detect nontrivial discontinuities in the presence of severe noise. Apart from the aforementioned efforts, Crespo and Schafer explicitly used edge information to represent significant discontinuities [37], which implies that they were using the solution to the problem that we attempt to solve. In general, identification of nontrivial discontinuities from an image is a challenging problem and, therefore, robust measures of a low computational complexity are still worth exploring for practical applications.

As a defining characteristic, iterative operations are inevitably involved in adaptive smoothing. Thus, performance of an iterative algorithm highly depends upon the termination time, which, coupled with the fact that adaptive smoothing algorithms generally converge to a uniform intensity image [3], [16], causes what we often refer to as the *termination* problem [8]. In other words, when and where to stop smoothing is a challenging problem in adaptive smoothing and no explicit stopping criterion has been found yet. Although little theoretical research has been done for solving the termination problem, there have been several heuristic methods to determine the stopping time in adaptive smoothing [17], [20], [33], [38], [39]. However, most of them are rather tricky and hard to use in practice. Recent studies reveal that the termination problem is strongly associated with the automatic control of local scales in image processing [40], [41]. Liang and Wang propose a criterion with the local scale control for stopping adaptive smoothing [41]. However, it is unclear whether such a gradient-based criterion enables adaptive smoothing to stop in a finite number of iterations. As pointed out by Saha and Udupa [29], there might be leakages on boundaries caused by local blurring artifacts. If the stopping criterion fails to detect such a leakage, adaptive smoothing would never stop at the boundary. As a consequence, smoothing runs straight across that boundary, and the intensity mass of adjacent regions would inevitably interact each other as long as adaptive smoothing runs for a sufficiently long time. In general, the termination problem is still open, which indeed becomes an obstacle for the use of adaptive smoothing in practice.

In our earlier work [35], we proposed a weight adaptation scheme for an oscillatory neural network in the context of image segmentation. In essence, this neuronal scheme tends to play a role of adaptive smoothing prior to segmentation. In this scheme, two different discontinuity measures, spatial variance of a single spatial scale (a tunable parameter) and local discontinuity defined in the immediate neighborhood of a pixel, are jointly employed for robustness. Although the scheme turns out to be effective for miscellaneous images, there are still several open problems. First, the contextual discontinuity measure of a single spatial scale leads to a difficulty in accurately capturing the intrinsic structure of a given image since the image often demands multiple local scales for characterizing its nature. Next, several parameters have to be tuned for a desirable contextual discontinuity map. If any proper parameters fail to be achieved, its performance would be considerably degraded. Furthermore, parameter tuning is rather tedious and complicated, which becomes a big hurdle for the scheme to be applied in practice. Apart from the tedious parameter tuning, our weight adaptation scheme was proposed simply based on a heuristic idea and there was no formal analysis on this scheme and its behaviors. Also, it is yet unclear about how such a scheme, as converted into an adaptive smoothing algorithm, is related to the well understood anisotropic diffusion theory.

In this paper, we further develop our previous heuristic idea [35] from an alternative perspective in order to come up with a systematic and theoretically supported adaptive smoothing approach. Unlike our earlier work, we first explore alternative discontinuity measures for detecting contextual and local discontinuities. Recently, Saha et al. proposed a scale-based affinity theory [36], which leads to a parameter-free algorithm for detecting region homogeneity based on local scales [36], [42]. Instead of the spatial variance, we modify their algorithm to come up with an alternative contextual discontinuity measure, hereinafter named *inhomogeneity*, for our purpose. The inhomogeneity measure not only avoids a tedious parameter tuning process but also offers image-dependent local scales for a given image. In addition, spatial gradient is simply employed as a local discontinuity measure to monitor variations of local discontinuities during smoothing. Furthermore, we analyze the proposed algorithm and show how our smoothing process is related to anisotropic diffusion, a generic framework for understanding adaptive smoothing. This analysis uncovers that our adaptive smoothing algorithm turns out to be an alternative implementation of the nonlinear anisotropic diffusion process, first proposed by Perona and Malik [16], by introducing two complementary discontinuity measures to diffusion coefficients. The use of a contextual discontinuity measure in adaptive smoothing is actually exploiting the intrinsic nature of a given image to sensibly constrain diffusion, which forms a genuine nonlinear anisotropic diffusion process. A formal analysis on behaviors of our algorithm with respect to a step edge is also presented. By simulations, we demonstrate that the resultant images in a wide range of iterations lead to the nearly identical results, which provides, to a great extent, a practical way to alleviate the termination problem. To evaluate its performance, we have applied our adaptive smoothing algorithm to images corrupted by a variety of noise and a special image segmentation task—extraction of hydrographic objects from satellite images. Simulations indicate that our algorithm yields favorable results in comparison with several recent smoothing algorithms, and its application in extraction of hydrographic objects demonstrates its usefulness as a tool for early vision.

The contributions of this paper are summarized as follows: First, we introduce a multiscale contextual discontinuity measure based on the scale-based affinity theory [36], [42] to our adaptive smoothing scheme and come up with a new normalization scheme to further enhance the measure. Such a multiscale measure along with its normalization scheme turns out to effectively overcome the weakness of our previous heuristic idea [35]. As a result, the proposed inhomogeneity measure of multiple local scales without the need of parameter tuning captures the intrinsic structure of a given image more accurately, which significantly distinguishes from the spatial variance of a single local scale obtained by parameter tuning and then uniformly applied to the image for encoding contextual discontinuities [35]. Second, we formally analyze stability and behaviors of our algorithm with respect to a step edge, which offers an insight into how our algorithm works during smoothing. Furthermore, we show that our algorithm provides a novel implementation of the nonlinear anisotropic diffusion process [16]; i.e., our algorithm leads to a sensibly constrained diffusion process that distinguishes from those generated by existing adaptive smoothing algorithms. This contribution relates our algorithm closely to existing adaptive smoothing algorithms and gains a clear insight into the essence of our algorithm. Finally, by an empirical study, we demonstrate the usefulness of our adaptive smoothing algorithm with its favorite features, e.g., insensitive to termination times and free of parameter tuning.

The reminder of this paper is organized as follows: Section 2 presents different discontinuity measures and our adaptive smoothing algorithm. Section 3 analyzes our algorithm and its behaviors and further relates it to anisotropic diffusion. Section 4 describes simulations including methodology, comparative results of smoothing, and an application in extraction of hydrographic objects from satellite images. Section 5 discusses related issues, and the last section draws conclusions.

2 Adaptive Smoothing by Combining Discontinuity Measures

In this section, we first describe two different discontinuity measures and then present an adaptive smoothing algorithm by integrating both discontinuity measures.

2.1 Discontinuity Measures

It is well-known that discontinuities in an image likely correspond to important features. However, noise corruption can generate discontinuities as well. Therefore, how to measure significant discontinuities is a nontrivial issue. To facilitate smoothing, two types of discontinuity are used in our approach: *local* and *contextual* discontinuity. A local measure is used to detect variable local discontinuities and acutely sensitive to any local intensity change. However, the local discontinuity measure is not always robust. If noise is substantial in an image, this measure cannot distinguish between significant features and noise due to overlocality. On the other hand, we observe that contextual information, i.e., the attributes of neighboring pixels, can be used to reduce ambiguity even in noisy circumstances. An ensemble of coupled pixels provides a basis of detecting contextual discontinuities corresponding to potentially important features, and it is anticipated that an effective contextual discontinuity measure can detect the intrinsic nature of an image. Essentially, we would like local discontinuities to indicate the details of local structures and contextual discontinuities to offer a "road map" to specify where important features are in a given image. Unlike our earlier work [35], we employ the spatial gradient [7] as a local discontinuity measure and derive a new contextual discontinuity measure from the scale-based affinity theory [36].

2.1.1 Spatial Gradient

As a common local discontinuity measure in image processing, spatial gradient is approximately estimated in a nearest neighborhood [7]. For a pixel (x, y), we define its neighborhood, $\mathcal{N}_{xy}(R)$, as

$$\mathcal{N}_{xy}(R) = \Big\{(i,j) \mid |x-i| \le R,\ |y-j| \le R\Big\}, \tag{1}$$

where R ($R \ge 0$ and $R \in \mathcal{Z}$) is a parameter that determines the size of this neighborhood. Note that here the neighborhood is defined without explicitly considering image boundaries. The neighborhood is delimited by the image boundaries. For convenience in notation, we stipulate that $\mathcal{N}_{xy}(0)$ is pixel (x, y) itself, which is actually not a neighborhood of the pixel. As described later, the parameter $R > 0$ specifies a local spatial scale associated with pixel (x, y) that critically determines results of the contextual discontinuity measure. Obviously, the nearest neighborhood of pixel (x, y) is always $\mathcal{N}_{xy}(1)$. As defined in [7], spatial gradient at pixel (x, y) is the first derivatives of its image intensity function:

$$\nabla \boldsymbol{I}_{(x,y)} = \Big[\frac{\partial I_{(x,y)}}{\partial x}, \frac{\partial I_{(x,y)}}{\partial y}\Big]^T, \tag{2}$$

where the derivatives are approximated in $\mathcal{N}_{xy}(1)$ by

$$\frac{\partial I_{(x,y)}}{\partial x} = I_{(x+1,y)} - I_{(x-1,y)}, \tag{3a}$$

$$\frac{\partial I_{(x,y)}}{\partial y} = I_{(x,y+1)} - I_{(x,y-1)}. \tag{3b}$$

Here, $I_{(x,y)}$ is the intensity of pixel (x, y). Accordingly, the magnitude of the gradient vector in (2) is given by

$$|\nabla \boldsymbol{I}_{(x,y)}| = \sqrt{\Big(\frac{\partial I_{(x,y)}}{\partial x}\Big)^2 + \Big(\frac{\partial I_{(x,y)}}{\partial y}\Big)^2}. \tag{4}$$

2.1.2 Inhomogeneity

We now derive a contextual discontinuity measure, i.e., inhomogeneity, from the scale-based affinity theory proposed by Saha et al. [36]. The basic idea underlying the scale-based affinity between two pixels (x, y) and (i, j) in a gray-level image is finding out a proper neighborhood or local scale at two pixels so that all the pixels in the neighborhood meet an intensity uniformity criterion. Using this idea, we first determine an appropriate size of neighborhood or scale R_{xy} for any pixel in a given image and further define the inhomogeneity measure.

To facilitate our presentation, we define the *boundary region* of a neighborhood of pixel (x, y) of size R by

$$\mathcal{B}_{xy}(R) = \{(i,j) \mid (i,j) \in \mathcal{N}_{xy}(R) - \mathcal{N}_{xy}(R-1)\}. \tag{5}$$

Here, $R > 0$ ($R \in \mathcal{Z}$) and the region is delimited by the image boundaries. As defined in [36], the uniformity criterion for testing the similarity between pixel (x, y) and the ensemble of pixels in the boundary region of its neighborhood of size R is

$$U_{(x,y)}(R) = \frac{\sum_{(i,j)\in\mathcal{B}_{xy}(R)} \exp\left[-\frac{(I_{(x,y)}-I_{(i,j)})^2}{2\sigma_u^2}\right]}{|\mathcal{B}_{xy}(R)|}, \tag{6}$$

where $|\mathcal{B}_{xy}(R)|$ is the cardinality of $\mathcal{B}_{xy}(R)$ and σ_u is a parameter reflecting the statistics of local intensity difference in a given image. As suggested by Saha et al. [36], this parameter is determined by estimating statistics of local intensity differences over the entire image. In other words, for each pixel (x,y) in the image, its local intensity differences, $|I_{(x,y)} - I_{(i,j)}|$, are first calculated based on its nearest neighborhood, $(i,j) \in \mathcal{B}_{xy}(1)$. After ruling out the upper 10 percent values of all the local intensity differences, which are assumed to present interobject boundaries, we estimate the mean and the variance of the remaining 90 percent values of differences, μ_d and σ_d. Suppose that the remaining local intensity difference is subject to the Gaussian distribution, the parameter σ_u is determined by

$$\sigma_u = \mu_d + 3\sigma_d, \tag{7}$$

which indicates a fact on the Gaussian distribution; i.e., three standard deviations on both sides of the mean cover 99.7 percent of the population. Intuitively, σ_u can be viewed as an estimate of noise or trivial details within intraobjects. Therefore, (6) offers a similarity measure to determine the coherence between a pixel and the ensemble of all pixels in the boundary region of its neighborhood. Thus, finding a local scale based on (6) becomes a problem to search for the maximal neighborhood for each pixel where the focused pixel is coherent with the ensemble of pixels in all possible boundary regions up to this neighborhood; i.e., for pixel (x,y) its optimal local scale is

$$\begin{aligned} R_{xy} &= \underset{r\in\mathcal{Z},\, r>1}{\operatorname{argmax}}\{U_{(x,y)}(r) \geq T_s\} \\ &s.t.\ \forall R \in \mathcal{Z}\ (1 \leq R < R_{xy}-1),\ U_{(x,y)}(r-R) \geq T_s, \end{aligned} \tag{8}$$

where T_s is a tolerance parameter and fixed to be 0.85 as recommended in [36]. Apparently, a solution to the local optimization problem in (8) is in compliance with an exhausted search procedure that checks the similarity between a pixel and its possible boundary regions in an ascent order from $r = 2$. Note that, unlike the definition in [36], we stipulate that the minimal local scale or neighborhood of a pixel be one or its nearest neighborhood rather than zero or the pixel itself for robustness thanks to our previous investigation [43].

Once optimal local scales are obtained, we further determine the disconnectedness or incoherence between two adjacent pixels where one pixel is a member of the nearest neighborhood of the other mutually. For a pair of adjacent pixels (x,y) and (i,j), two neighborhoods are defined based on their optimal local scales R_{xy} and R_{ij}:

$$\begin{aligned} \mathcal{N}_{xy;ij}(x,y) &= \{(v,w) \mid |x-v| \leq R_{xy;ij},\ |y-w| \leq R_{xy;ij}\}, \\ \mathcal{N}_{xy;ij}(i,j) &= \{(v',w') \mid |i-v'| \leq R_{xy;ij},\ |j-w'| \leq R_{xy;ij}\}. \end{aligned}$$

Here, $R_{xy;ij} = \min\{R_{xy}, R_{ij}\}$. By the definition, two adjacent pixels always have neighborhoods of the same size such that a correspondence between two neighborhoods can be established to achieve local intensity differences. Suppose that pixels (v,w) and (v',w') are an aligned pair of pixels in neighborhoods $\mathcal{N}_{xy;ij}(x,y)$ and $\mathcal{N}_{xy;ij}(i,j)$, respectively; they must satisfy the constraints: $x-v = i-v'$ and $y-w = j-w'$. As defined in [36], the intensity differences between neighborhoods $\mathcal{N}_{xy;ij}(x,y)$ and $\mathcal{N}_{xy;ij}(i,j)$ are

$$D^+_{xy;ij} = \sum_{\substack{(v,w)\in\mathcal{N}_{xy;ij}(x,y) \\ (v',w')\in\mathcal{N}_{xy;ij}(i,j) \\ s.t.\ x-v=i-v',\, y-w=j-w'}} \left[1-\exp\left(-\frac{(d^+_{vw;v'w'})^2}{2R^2_{xy;ij}}\right)\right] \exp\left(-\frac{d^2_{xy;vw}}{2R^2_{xy;ij}}\right), \tag{9a}$$

$$D^-_{xy;ij} = \sum_{\substack{(v,w)\in\mathcal{N}_{xy;ij}(x,y) \\ (v',w')\in\mathcal{N}_{xy;ij}(i,j) \\ s.t.\ x-v=i-v',\, y-w=j-w'}} \left[1-\exp\left(-\frac{(d^-_{vw;v'w'})^2}{2R^2_{xy;ij}}\right)\right] \exp\left(-\frac{d^2_{xy;vw}}{2R^2_{xy;ij}}\right), \tag{9b}$$

where

$$d^2_{xy;vw} = (x-v)^2 + (y-w)^2,$$

$$d^+_{vw;v'w'} = \begin{cases} I_{(v,w)} - I_{(v',w')}, & \text{if } I_{(v,w)} - I_{(v',w')} > 0, \\ 0, & \text{otherwise.} \end{cases}$$

$$d^-_{vw;v'w'} = \begin{cases} I_{(v,w)} - I_{(v',w')}, & \text{if } I_{(v,w)} - I_{(v',w')} < 0, \\ 0, & \text{otherwise.} \end{cases}$$

As argued by Saha and Udupa [42], $D^+_{xy;ij}$ and $D^-_{xy;ij}$ in (9) together offer a measure for detecting intra and interobject intensity variations in the vicinity of pixels (x,y) and (i,j). Therefore, we define the disconnectedness or incoherence between two adjacent pixels (x,y) and (i,j) as

$$\Psi[\mathcal{N}_{xy;ij}(x,y), \mathcal{N}_{xy;ij}(i,j)] = \frac{|D^+_{xy;ij} - D^-_{xy;ij}|}{\sum_{(v,w)\in\mathcal{N}_{xy;ij}(x,y)} \exp\left(-\frac{d^2_{xy;vw}}{2R^2_{xy;ij}}\right)}.$$

Here, we anticipate that $\Psi[\mathcal{N}_{xy;ij}(x,y), \mathcal{N}_{xy;ij}(i,j)]$ indicates the degree of nonuniformity of the regions, defined by $\mathcal{N}_{xy;ij}(x,y)$ and $\mathcal{N}_{xy;ij}(i,j)$, surrounding pixels (x,y) and (i,j). That is, the value of $\Psi[\mathcal{N}_{xy;ij}(x,y), \mathcal{N}_{xy;ij}(i,j)]$ would be high if pixels (x,y) and (i,j) are located around a boundary or a nontrivial discontinuity, otherwise, its value tends to be low. Thus, we define the inhomogeneity of pixel (x,y) as

$$H_{(x,y)} = \frac{\sum_{(i,j)\in\mathcal{B}_{xy}(1)} \Psi[\mathcal{N}_{xy;ij}(x,y), \mathcal{N}_{xy;ij}(i,j)]}{|\mathcal{B}_{xy}(1)|}. \tag{10}$$

As a robust multiscale measure defined by an ensemble of coupled pixels, inhomogeneity tends to reveal the intrinsic disconnectedness or incoherence between a pixel and its surrounding by means of a scale-based or image-dependent property.

To facilitate the use of inhomogeneity as a contextual discontinuity measure, we normalize $H_{(x,y)}$ by

$$\bar{H}_{(x,y)} = \frac{H_{(x,y)} - H_{\min}}{H_{\max} - H_{\min}}, \tag{11}$$

where $H_{\max}$ and $H_{\min}$ are the maximal and minimal inhomogeneity values across the entire image, respectively. For an image, all the normalized inhomogeneity values monotonically distribute within the interval $[0,1]$ to ensure that the maximum is one and the minimum is zero. In order to enhance the contextual discontinuities underlying an image, we further introduce a nonlinear transformation to highlight higher inhomogeneity that more likely corresponds to important features. There are a variety of nonlinear transformations to meet such a requirement. One of them used in our simulations is a *sine* transformation:

$$\hat{H}_{(x,y)} = \sin\left(\frac{\pi}{2}\bar{H}_{(x,y)}\right), \quad 0 \leq \bar{H}_{(x,y)} \leq 1. \tag{12}$$

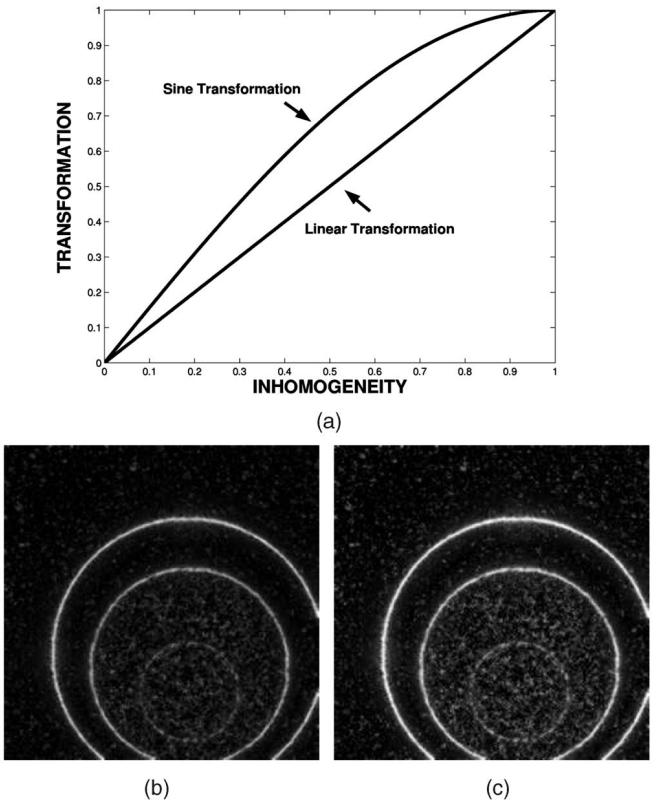

Fig. 1. Inhomogeneity transformations and their effects. (a) Linear and sine transformations. (b) The linear transformation of inhomogeneities underlying the image shown in Fig. 2b. (c) The nonlinear transformation of inhomogeneities underlying the image shown in Fig. 2b.

The effect of such a nonlinear transformation is depicted in Fig. 1a. As a demonstration, an example of the normalized inhomogeneity map and its nonlinear version are illustrated in Figs. 1b and 1c for the noisy image shown in Fig. 2b. In the maps, brightness is proportional to the degree of inhomogeneities. Thus, it is evident that the inhomogeneity maps readily indicate locations of significant discontinuities and homogeneous regions. Moreover, the map corresponding to the nonlinear version suggests that the inhomogeneities indicating most of important features, e.g., boundaries of circles, are enhanced, which may better protect significant discontinuities from being blurred during adaptive smoothing. We argue that the normalized inhomogeneity and its nonlinear version offer a multiscale contextual discontinuity measure, which forms a "road map" for guiding adaptive smoothing. Here, we emphasize that there is no parameter tuning in detection of such inhomogeneity.

2.2 Algorithm

The generic form of adaptive smoothing is to update a pixel's intensity through local weighted averaging of its neighboring pixels' intensities. A critical issue in adaptive smoothing is how to utilize local image structures to determine proper weights for averaging. In general, a large weight should be assigned to a pixel that involves low discontinuities, and vice versa. Most of adaptive smoothing algorithms simply use one discontinuity measure, e.g., spatial gradient, to generate weights. Unlike previous algorithms, we develop an alternative adaptive smoothing scheme by combining both inhomogeneity and spatial gradient described in Section 2.1 as follows:

$$I_{(x,y)}^{(t+1)} = I_{(x,y)}^{(t)} + \alpha_{(x,y)} \frac{\sum_{(i,j)\in\mathcal{B}_{xy}(1)} \alpha_{(i,j)}\beta_{(i,j)}^{(t)}(I_{(i,j)}^{(t)} - I_{(x,y)}^{(t)})}{\sum_{(i,j)\in\mathcal{B}_{xy}(1)} \alpha_{(i,j)}\beta_{(i,j)}^{(t)}}, \quad (13)$$

where $I_{(x,y)}^{(t)}$ is the intensity of pixel (x,y) at iteration t. $\alpha_{(x,y)}$ and $\beta_{(x,y)}^{(t)}$ are defined as

$$\alpha_{(x,y)} = g(\hat{\boldsymbol{H}}_{(x,y)}, h), \quad (14a)$$

$$\beta_{(x,y)}^{(t)} = g(|\nabla \boldsymbol{I}_{(x,y)}^{(t)}|, S). \quad (14b)$$

Here, $g(\cdot,\cdot)$ is a nonnegative monotonically decreasing function. There are numerous functions of such a property. Two common forms are

$$g(z,K) = \exp\left(-\frac{z^2}{2K^2}\right), \quad (15a)$$

$$g(z,K) = \frac{1}{1+\frac{z^2}{K^2}}. \quad (15b)$$

In (14), parameter h $(0<h<1)$ is used to determine to what extent potential important features should be preserved in terms of contextual discontinuities, and parameter S $(S>0)$ determines to what extent local discontinuities should be preserved during smoothing. The selection of two parameters will be discussed later on. By nonlinear transformations in (14), the α term encodes the effect of intrinsic contextual discontinuities while the β term encodes the instantaneous effect of local discontinuities during smoothing.

In (13), the intensity of pixel (x,y) is updated based on its neighboring attributes in terms of both contextual and local discontinuities. While the intensity of pixel (x,y) is updated, both $\beta_{(i,j)}^{(t)}$ and $\alpha_{(i,j)}$ associated with its nearest neighbors (cf., (13) and (14)) are jointly used to determine weights for local weighted averaging, and $\alpha_{(x,y)}$ itself is employed for the gain control of smoothing. For local weighted averaging, $\beta_{(i,j)}^{(t)}$ would play a dominant role for noise removal and feature preservation when all the adjacent pixels of (x,y) have similar contextual discontinuities. In this circumstance, pixel (x,y) should be within a relatively homogeneous region and its intensity is updated to fit its local structures in terms of the parameter S. On the other hand, if the adjacent pixels of (x,y) have different contextual discontinuities, weights for update must be determined by both contextual and local discontinuities. In this case, pixel (x,y) and its neighbors are likely near a boundary of different regions and, therefore, the role of its neighbors in update is determined by the overall effects of local and contextual discontinuities. That is, for a neighboring pixel (i,j), it will play a more important role in update if its overall discontinuities are relatively low or the value of $\alpha_{(i,j)}\beta_{(i,j)}^{(t)}$ is relatively large, and vice versa. As mentioned before, the local discontinuity measure is sensitive to any local intensity changes, while the contextual discontinuity measure is relatively not but more robust. The joint use of the two measures leads to a complementary effect for feature preservation. In addition, if pixel (x,y) resides in a relatively homogeneous region, the value of $\alpha_{(x,y)}$ is large so that fast smoothing is performed there, otherwise, the smoothing tends to be slow under control of $\beta_{(x,y)}$ so that important features can be preserved. The use of contextual discontinuity in this way introduces a novel mechanism for preserving important features along with noise removal.

Given an image, $I_{(x,y)}^{(0)}$ denotes the original intensity of pixel (x,y). We summarize the proposed adaptive smoothing algorithm as follows:

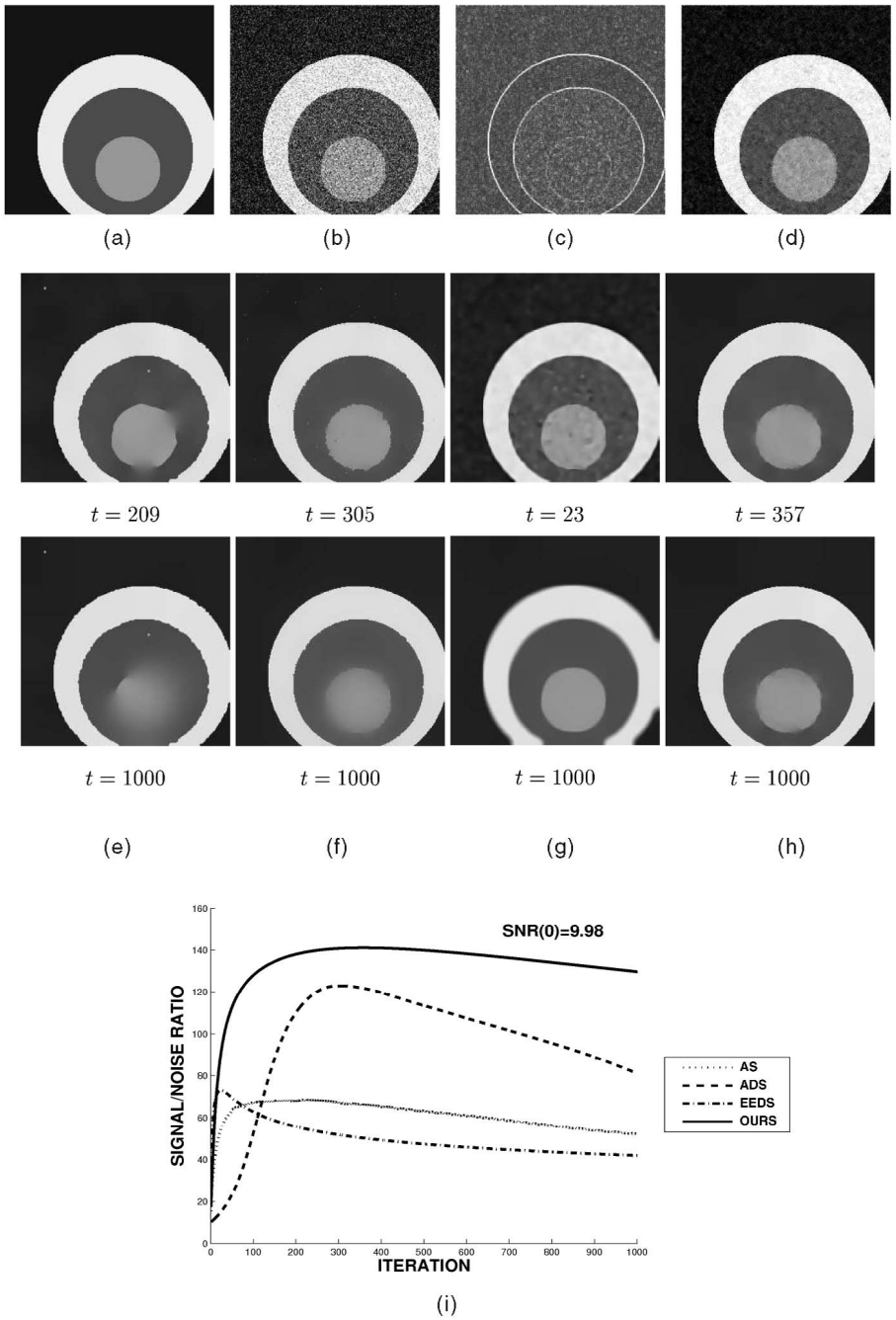

Fig. 2. Smoothing results of a synthetic image. (a) A synthetic image consisting of 256×256 pixels. (b) A noisy image of (a) where Gaussian noise of $\mu = 0$ and $\sigma = 50$ was added [7]. (c) A spatial gradient map of the image in (b). (d) The hypergraph algorithm. (e) The AS algorithm with $k = 11.5$. (f) The ADS algorithm with $K = 6.5$. (g) The EEDS algorithm with $\lambda = 4.5$ and $\sigma = 1.5$. (h) Ours with $h = 0.125$ and $S = 31.1$. (i) SNR evolution.

1. Initialization.
   - 1.1. Input a given image.
   - 1.2. Set parameters h, S for smoothing and T (maximal iteration number) for termination.
2. For each pixel (x, y), compute $\alpha_{(x,y)}$ by the inhomogeneity measure in Section 2.1.2 and (14a).
3. Iterate until $t = T$.
   - 3.1. For each pixel (x, y), compute $\beta^{(t)}_{(x,y)}$ by (3), (4), and (14b).
   - 3.2. Perform local weighted averaging to update $I^{(t)}_{(x,y)}$:

$$I^{(t+1)}_{(x,y)} = I^{(t)}_{(x,y)} + \alpha_{(x,y)} \frac{\sum_{(i,j)\in\mathcal{B}_{xy}(1)} \alpha_{(i,j)}\beta^{(t)}_{(i,j)}\left(I^{(t)}_{(i,j)} - I^{(t)}_{(x,y)}\right)}{\sum_{(i,j)\in\mathcal{B}_{xy}(1)} \alpha_{(i,j)}\beta^{(t)}_{(i,j)}}.$$

Parallelism is a desirable characteristic of an efficient image processing algorithm. Our smoothing algorithm can be recognized as a parallel process at the pixel level. By parallel computation, elementary operations, e.g., addition, multiplication, and exponential, are applied simultaneously to each pixel of the image. Note that the evaluation of α_{xy} needs to be computed only once prior to smoothing though it may

take relatively long time. Operations relevant to indexing, such as computing $\beta_{xy}^{(t)}$ and local weighted averaging, need only information on immediate neighbors of a pixel, which can be efficiently carried out on a parallel computer without difficulty. In summary, our algorithm provides a simplified yet efficient way of utilizing multiple local scales for information fusion. However, it is different from general multiple scale approaches [44]. Instead of applying the same operator of different scales to an image, statistical information is acquired through inhomogeneity in a scale-dependent contextual neighborhood and mainly employed to guide the local smoothing, and geometrical constraints in a local neighborhood are enforced through local smoothing for preserving significant boundaries precisely.

3 Algorithm Analysis

In this section, we formally analyze our adaptive smoothing algorithm. We first analyze its stability and behaviors in terms of a blurred step edge. Then, we relate our algorithm to nonlinear anisotropic diffusion from different perspectives.

3.1 Stability

For an adaptive scheme, stability is an important issue that concerns possible unbounded growth or boundness of the final result of the adaptive scheme. The essential criterion defining stability is that this numerical process must restrict the amplification of all components from the initial conditions. Here, we show that the stability of our smoothing scheme is guaranteed. Let I_{max} and I_{min} denote the maximum and the minimum of intensities across a given image. According to (13), we have

$$\begin{aligned} I_{(x,y)}^{(t+1)} &= I_{(x,y)}^{(t)} + \alpha_{(x,y)} \frac{\sum_{(i,j)\in\mathcal{B}_{xy}(1)} \alpha_{(i,j)}\beta_{(i,j)}^{(t)}(I_{(i,j)}^{(t)} - I_{(x,y)}^{(t)})}{\sum_{(i,j)\in\mathcal{B}_{xy}(1)} \alpha_{(i,j)}\beta_{(i,j)}^{(t)}} \\ &= (1-\alpha_{(x,y)})I_{(x,y)}^{(t)} \\ &\quad + \alpha_{(x,y)} \sum_{(i,j)\in\mathcal{B}_{xy}(1)} \frac{\alpha_{(i,j)}\beta_{(i,j)}^{(t)}}{\sum_{(i,j)\in\mathcal{B}_{xy}(1)} \alpha_{(i,j)}\beta_{(i,j)}^{(t)}} I_{(i,j)}^{(t)}. \end{aligned}$$

Therefore, we can achieve

$$\begin{aligned} I_{(x,y)}^{(t+1)} &= (1-\alpha_{(x,y)})I_{(x,y)}^{(t)} \\ &\quad + \alpha_{(x,y)} \sum_{(i,j)\in\mathcal{B}_{xy}(1)} \frac{\alpha_{(i,j)}\beta_{(i,j)}^{(t)}}{\sum_{(i,j)\in\mathcal{B}_{xy}(1)} \alpha_{(i,j)}\beta_{(i,j)}^{(t)}} I_{(i,j)}^{(t)} \\ &\geq (1-\alpha_{(x,y)})I_{(x,y)}^{(t)} + \alpha_{(x,y)} I_{min} \geq I_{min}. \end{aligned}$$

Similarly, we can also obtain

$$\begin{aligned} I_{(x,y)}^{(t+1)} &= (1-\alpha_{(x,y)})I_{(x,y)}^{(t)} \\ &\quad + \alpha_{(x,y)} \sum_{(i,j)\in\mathcal{B}_{xy}(1)} \frac{\alpha_{(i,j)}\beta_{(i,j)}^{(t)}}{\sum_{(i,j)\in\mathcal{B}_{xy}(1)} \alpha_{(i,j)}\beta_{(i,j)}^{(t)}} I_{(i,j)}^{(t)} \\ &\leq (1-\alpha_{(x,y)})I_{(x,y)}^{(t)} + \alpha_{(x,y)} I_{max} \leq I_{max}. \end{aligned}$$

Thus, the intensity of any pixel (x, y) in a smoothed image is always bounded by $I_{min} \leq I_{(x,y)}^{(t+1)} \leq I_{max}$, which guarantees the stability of our algorithm.

3.2 Behaviors on Step Edge

As a desirable characteristic, adaptive smoothing is capable of feature preservation along with noise removal. A step edge with the significant contrast is one of the most important discontinuities. Now, we analyze the behaviors of our algorithm with respect to a blurred step edge or a continuous version of a hard step edge.

Without loss of generality, here, we consider only a one-dimensional case. Thus, a blurred step edge can be modeled by a sigmoid function with a steep slope. For such a blurred step edge, we have the following properties in the vicinity of its inflection: $\frac{\partial I_x}{\partial x} > 0$, $\frac{\partial^2 I_x}{\partial x^2} = 0$, and $\frac{\partial^3 I_x}{\partial x^3} < 0$. Recall that weights of pixels for adaptive smoothing depend upon inhomogeneityand spatial gradient with a nonlinear transformation in (15). For analysis, we adopt (15a) given that the same analysis can be extended to (15b) without difficulty. Based on (15a), we rewrite the weight of point x by dropping the normalization term, which does not affect the analysis, as follows:

$$C_x^{(t)} = \alpha_x \beta_x^{(t)} = \exp\left\{ -\left[\frac{\hat{H}_x^2}{2h^2} + \frac{1}{2S^2}\left(\frac{\partial I_x^{(t)}}{\partial x} \right)^2 \right] \right\}. \quad (16)$$

Here, we approximate the response of $\hat{H}_x$ to a step edge by a Gaussian function with the center at the point of inflection, which is differentiable with the property: $\frac{\partial^2 \hat{H}_x}{\partial x^2} < 0$. To observe the behaviors of our algorithm with respect to a step edge, we examine the spatial gradient change in the point of inflection along the time dimension. In the Appendix, we derive an expression on such an iteration-dependent change as follows:

$$\begin{aligned} \frac{\partial}{\partial t}\left(\frac{\partial I_x^{(t)}}{\partial x} \right) &= C_x^{(t)} \frac{\partial^3 I_x^{(t)}}{\partial x^3}\left[1 - \frac{1}{S^2}\left(\frac{\partial I_x^{(t)}}{\partial x} \right)^2 \right] \\ &\quad + \frac{C_x^{(t)}}{h^2}\frac{\partial I_x^{(t)}}{\partial x}\left[(\hat{H}_x^2 - h^2)\left(\frac{\partial \hat{H}_x}{\partial x} \right)^2 - \hat{H}_x \frac{\partial^2 \hat{H}_x}{\partial x^2} \right]. \end{aligned} \quad (17)$$

Based on the properties of sigmoid and Gaussian functions, we have $\frac{\partial^3 I_x^{(t)}}{\partial x^3} < 0$, $\frac{\partial I_x^{(t)}}{\partial x} > 0$, and $\frac{\partial^2 \hat{H}_x}{\partial x^2} < 0$. In addition, $C_x^{(t)}$ of the exponential form and $\hat{H}_x$ are always positive. Thus, the sign of $\frac{\partial}{\partial t}(\frac{\partial I_x^{(t)}}{\partial x})$ in (17) is critically determined by $[1 - \frac{1}{S^2}(\frac{\partial I_x^{(t)}}{\partial x})^2]$ and $(\hat{H}_x^2 - h^2)$. Obviously, the sign of $\frac{\partial}{\partial t}(\frac{\partial I_x^{(t)}}{\partial x})$ would be always positive if $[1 - \frac{1}{S^2}(\frac{\partial I_x^{(t)}}{\partial x})^2] < 0$ and $(\hat{H}_x^2 - h^2) > 0$. That is, the magnitude of spatial gradient in the vicinity of the step edge will always increase during smoothing when $\frac{\partial I_x^{(t)}}{\partial x} > S$ and $\hat{H}_x > h$. It manifests that a step edge would be always sharpened if significant spatial gradient and inhomogeneity are detected there. Naturally, parameters S and h play the role specifying significant discontinuities. Note that all the terms associated with $\hat{H}_x$ in (17) are determined once an image is given and, therefore, are regardless of iterations, which plays a regularization role in identifying significant discontinuities in the presence of substantial noise. Moreover, (17) provides a basis for explaining behaviors of our algorithm, which will be further discussed later on.

3.3 Smoothing and Diffusion

Anisotropic diffusion has become a framework to build a theoretical foundation for adaptive smoothing. Here, we show how our smoothing process closely relates to

anisotropic diffusion and further present an analysis that distinguishes our algorithm from classical anisotropic diffusion algorithms in light of the energy-based framework for image reconstruction [4]. First, we consider the one-dimensional case for intuition. As a result, we reformulate our smoothing scheme as follows:

$$I_x^{(t+1)} = I_x^{(t)} + \alpha_x \frac{\sum_{i\in\mathcal{B}_x(1)} \alpha_i\beta_i^{(t)}(I_i^{(t)} - I_x^{(t)})}{\sum_{i\in\mathcal{B}_x(1)} \alpha_i\beta_i^{(t)}}.$$

Here, $\mathcal{B}_x(1) = \{x-1, x+1\}$. Therefore, the iterative update process is

$$I_x^{(t+1)} = I_x^{(t)} + \frac{\alpha_x\alpha_{x-1}\beta_{x-1}^{(t)}(I_{x-1}^{(t)} - I_x^{(t)}) + \alpha_x\alpha_{x+1}\beta_{x+1}^{(t)}(I_{x+1}^{(t)} - I_x^{(t)})}{\alpha_{x-1}\beta_{x-1}^{(t)} + \alpha_{x+1}\beta_{x+1}^{(t)}}.$$

To facilitate the presentation, we use a term $c_{i;x}^{(t)}$ to denote the weights of point i in the nearest neighborhood of point x, $i \in \{x-1, x+1\}$, at iteration t as

$$\begin{aligned} c_{i;x}^{(t)} &= \frac{\alpha_x\alpha_i\beta_i^{(t)}}{\alpha_{x-1}\beta_{x-1}^{(t)} + \alpha_{x+1}\beta_{x+1}^{(t)}} \\ &= \frac{g(\hat{\boldsymbol{H}}_x, h)g(\hat{\boldsymbol{H}}_i, h)g(|\nabla\mathbf{I}_i^{(t)}|, S)}{g(\hat{\boldsymbol{H}}_{x-1}, h)g(|\nabla\mathbf{I}_{x-1}^{(t)}|, S) + g(\hat{\boldsymbol{H}}_{x+1}, h)g(|\nabla\mathbf{I}_{x+1}^{(t)}|, S)}. \end{aligned}$$

Thus, we have

$$\begin{aligned} I_x^{(t+1)} &= I_x^{(t)} + c_{x-1;x}^{(t)}(I_{x-1}^{(t)} - I_x^{(t)}) + c_{x+1;x}^{(t)}(I_{x+1}^{(t)} - I_x^{(t)}) \\ &= I_x^{(t)} + c_{x+1;x}^{(t)}(I_{x+1}^{(t)} - I_x^{(t)}) - c_{x-1;x}^{(t)}(I_x^{(t)} - I_{x-1}^{(t)}), \end{aligned}$$

which can be rewritten as

$$I_x^{(t+1)} - I_x^{(t)} = c_{x+1;x}^{(t)}(I_{x+1}^{(t)} - I_x^{(t)}) - c_{x-1;x}^{(t)}(I_x^{(t)} - I_{x-1}^{(t)}). \quad (18)$$

Apparently, (18) is approximating the anisotropic diffusion equation proposed by Perona and Malik [16]:

$$\frac{\partial I}{\partial t} = \frac{\partial}{\partial x}\left(c_{i;x}^{(t)}\frac{\partial I}{\partial x}\right) = \nabla(c_{i;x}^{(t)}\nabla I), \quad (19)$$

where $c_{i;x}^{(t)}$ becomes the diffusion coefficients.

For the two-dimensional case, our smoothing scheme is a discretization on a 3×3 square lattice. Unlike Perona and Malik's implementation, we adopt an eight-nearest-neighbors discretization of Laplacian operator. That is,

$$\begin{aligned} I_{(x,y)}^{(t+1)} = I_{(x,y)}^{(t)} + \Big(&c_E\triangle_E I + c_{NE}\triangle_{NE} I + c_N\triangle_N I + c_{NW}\triangle_{NW} I \\ &+ c_W\triangle_W I + c_{SW}\triangle_{SW} I + c_S\triangle_S I + c_{SE}\triangle_{SE} I\Big)_{(x,y)}^{(t)}. \end{aligned}$$

Here, E, NE, N, NW, W, SW, S, and SE are the mnemonic subscripts for eight directions, i.e., East, North-East, North, North-West, West, South-West, South, and South-West. The superscript and subscripts on the parenthesis are applied to all the terms enclosed, and the symbol $\triangle$ stands for nearest-neighbor differences. Let $\triangle I_{ij;xy}$ denote the local intensity difference $I_{(i,j)} - I_{(x,y)}$. We have

$$\begin{aligned} &\triangle_E I_{(x,y)} = \triangle I_{x+1y;xy}, \quad \triangle_{NE} I_{(x,y)} = \triangle I_{x+1y-1;xy}, \\ &\triangle_N I_{(x,y)} = \triangle I_{xy-1;xy}, \quad \triangle_{NW} I_{(x,y)} = \triangle I_{x-1y-1;xy}, \\ &\triangle_W I_{(x,y)} = \triangle I_{x-1y;xy}, \quad \triangle_{SW} I_{(x,y)} = \triangle I_{x-1y+1;xy}, \\ &\triangle_S I_{(x,y)} = \triangle I_{xy+1;xy}, \text{ and } \triangle_{SE} I_{(x,y)} = \triangle I_{x+1y+1;xy}. \end{aligned}$$

The diffusion coefficients are updated at every iteration as a nonlinear function of the magnitude of local discontinuities under control of contextual discontinuities or inhomogeneity. For pixel $(i,j) \in \mathcal{B}_{xy}(1)$, its diffusion coefficient is

$$c_{ij;xy}^{(t)} = \frac{\alpha_{(x,y)}\alpha_{(i,j)}\beta_{(i,j)}^{(t)}}{\sum_{(i,j)\in\mathcal{B}_{xy}(1)} \alpha_{(i,j)}\beta_{(i,j)}^{(t)}}. \quad (20)$$

Accordingly, we have

$$\begin{aligned} &c_{E_{(x,y)}}^{(t)} = c_{x+1y;xy}^{(t)}, \quad c_{NE_{(x,y)}}^{(t)} = c_{x+1y-1;xy}^{(t)}, \quad c_{N_{(x,y)}}^{(t)} = c_{xy-1;xy}^{(t)}, \\ &c_{NW_{(x,y)}}^{(t)} = c_{x-1y-1;xy}^{(t)}, \quad c_{W_{(x,y)}}^{(t)} = c_{x-1y;xy}^{(t)}, \quad c_{SW_{(x,y)}}^{(t)} = c_{x-1y+1;xy}^{(t)}, \\ &c_{S_{(x,y)}}^{(t)} = c_{xy+1;xy}^{(t)}, \text{ and } c_{SE_{(x,y)}}^{(t)} = c_{x+1y+1;xy}^{(t)}. \end{aligned}$$

As an alternative implementation, our smoothing scheme provides an approximate discretization of (19) for a gray-level image.

The energy-based methodology is a unified framework for early vision [4]. The basic idea in this framework is that an image processing process, e.g., edge detection and image segmentation, is performed by minimizing an energy function in the following form:

$$E(\boldsymbol{I}) = \sum_{(x,y)\in\mathbf{I},(i,j)\in\mathcal{N}_{xy}(R)} P(I_{(x,y)}, I_{(i,j)}) + \sum_{(x,y)\in\mathbf{I}} Q(I_{(x,y)}). \quad (21)$$

Here, $\boldsymbol{I}$ is the intensity set of all the pixels in a gray image. In (21), the first sum term is the a priori term encoding the a priori knowledge of the image space [2], while the second sum term is encoding the knowledge on the data available. $P(\cdot,\cdot)$ is usually an even function with respect to only the value of difference of its arguments and, therefore, can be written as $P(I_{(x,y)}, I_{(i,j)}) = P(I_{(x,y)} - I_{(i,j)})$. As shown by Perona and Malik [16], the approximation of anisotropic diffusion is simply a gradient descent of the a priori part of the energy function in (21) as $R = 1$. That is,

$$\frac{\partial I_{(x,y)}^{(t)}}{\partial t} = \eta \sum_{(i,j)\in\mathcal{N}_{xy}(1)} c(I_{(x,y)}^{(t)} - I_{(i,j)}^{(t)})(I_{(x,y)}^{(t)} - I_{(i,j)}^{(t)}), \quad (22)$$

where η $(0<\eta<1)$ is a "speed" factor to control the gradient decent search and $c(I_{(x,y)}^{(t)} - I_{(i,j)}^{(t)})$ is the iteration-dependent or time-dependent diffusion coefficients. In our algorithm, a diffusion coefficient (c.f., (20)) is determined by three components, $\alpha_{(x,y)}$, $\alpha_{(i,j)}$, and $\beta_{(i,j)}^{(t)}$. The first two components are iteration-independent or time-independent while the last one is the same as those in classical diffusion schemes, e.g., that of Perona and Malik [16], that use only spatial gradient to detect discontinuities. Let us denote $\eta\alpha_{(x,y)}$ as $\eta_{(x,y)}$ and $\beta_{(i,j)}^{(t)}$ as $c(I_{(x,y)}^{(t)} - I_{(i,j)}^{(t)})$. According to (22), our algorithm may be seen as approximating anisotropic diffusion in a constrained way by the minimization of the a priori part of the energy function in (21):

$$\frac{\partial I_{(x,y)}^{(t)}}{\partial t} = \eta_{(x,y)} \sum_{(i,j)\in\mathcal{N}_{xy}(1)} \alpha_{(i,j)} c(I_{(x,y)}^{(t)} - I_{(i,j)}^{(t)})(I_{(x,y)}^{(t)} - I_{(i,j)}^{(t)}).$$

Along with local discontinuity, the use of contextual discontinuity via inhomogeneity reshapes the landscape of the energy function with respect to a given image by taking account of the intrinsic local structural information

in the a priori term of (21) so that its search space can be limited to a great extent. Unlike other diffusion schemes, our gradient descent search is further under the control of image-dependent "speed" factors, i.e., $\eta_{(x,y)}$, which yields a genuine anisotropic diffusion process. In light of the energy-based framework, we conclude that the joint use of contextual and local discontinuity measures in our algorithm leads to a constrained diffusion process. Such a constrained diffusion generates a favorite feature for practical use; i.e., our algorithm is insensitive to termination conditions. In other words, important features can be preserved in a wide range of iterations, which will be demonstrated with simulation results reported in Section 4.

4 Simulations

In this section, we first describe the methodology used in our simulations then report comparative smoothing results for a variety of noisy images. Finally, we demonstrate that as a useful tool for early vision, our algorithm incorporated into a simple region growing algorithm effectively extracts hydrographic objects from satellite images, a special image segmentation task.

4.1 Methodology

It is a nontrivial issue to evaluate performance of an image processing algorithm, and there is less agreement on which method to use for perfectly evaluating all imageries [7], [8]. Based on an extensive literature search in the context of image smoothing, we can classify the existing performance evaluation methods into three categories; i.e., *objective*, *subjective*, and *application-based* methodologies. By the objective methodology, an evaluation is performed by comparing the smoothed image and its ground-truth to see how much noise has been removed from a noisy image. Based on the ground-truth, a gain or noise-reduction measure is defined for evaluating the improvement after smoothing. As long as the ground-truth of a test image is available, the objective methodology has become the first choice for performance evaluation [7], [9], [10], [11], [15], [20], [23], [28], [31], [40], [45], [46], [47], [48]. Unfortunately, the ground truth of a noisy image is often unavailable. Under the circumstance, almost all researchers adopt the subjective methodology for performance evaluation. By the subjective methodology, a noisy image used for test and its smoothed images are illustrated. Thus, the performance of an algorithm is evaluated by human's common sense gained from very much sophisticated visual perception experience [2], [5], [12], [13], [14], [15], [16], [17], [18], [19], [21], [22], [24], [25], [26], [27], [30], [33], [35], [37], [38], [39], [41], [43], [49], [50]. In addition, there is an application-based methodology for performance evaluation while two aforementioned methodologies dominate performance evaluation of smoothing algorithms. By the application-based methodology, images in a certain application domain are used for test and the smoothing results are assessed by either a specialist who has expertise in the domain or a comparison with an anticipated result set up prior to the test [29].

In order to demonstrate the favorite feature of our algorithm, *insensitive to termination times*, it is anticipated that multiple smoothed results in a wide range of iterations are presented. Due to limited space here, it is unrealistic to entirely adopt a subjective method for performance evaluation by illustrating many resultant images. Therefore, an objective method would be a proper choice where we can define a gain measure to encode smoothed results in a concise form, which is further backed by illustrating few typical snapshots. Here, a snapshot refers to a resultant image at a specific iteration. As a consequence, we first use synthetic and benchmark gray-level images for a test where their ground-truth is known and we add a variety of noise into them to generate various noisy images. Thus, we can evaluate its performance by an objective method as an adaptive smoothing algorithm is applied to such a noisy image. In order to demonstrate the effectiveness, we compare our algorithm to three classical adaptive smoothing algorithms: adaptive smoothing (AS) [5], anisotropic diffusion smoothing (ADS) [16], and edge-enhanced diffusion smoothing (EEDS) [33]. We also apply the adaptive hypergraph model [47], a noniterative algorithm, recently developed based on a generic hypergraph imaging theory [48] to the same images for further comparison. The ultimate goal of image smoothing is to facilitate the subsequent processing for early vision. To demonstrate the usefulness of our algorithm in an early vision task, we apply our algorithm to a special image segmentation task, extraction of hydrographic objects from satellite images, for an application-based evaluation.

Now, we define a gain measure, signal-to-noise ratio (SNR), for our objective evaluation. SNR measure has been widely used in evaluating performance of a smoothing algorithm in the objective methodology [7], [9], [10], [11], [15], [20], [23], [28], [31], [40], [45], [46], [47], [48]. For a given noisy image $\boldsymbol{I}$, $I^{(t)}_{(x,y)}$ denotes the intensity of pixel $(x,y) \in \boldsymbol{I}$ at iteration t while an adaptive smoothing algorithm is applied to the noisy image. $\boldsymbol{I}^{(gt)}$ is its ground-truth. As a result, an SNR measure is defined as follows:

$$\mathrm{SNR}(t, \boldsymbol{\Phi}) = \frac{\sum_{(x,y)\in \boldsymbol{I}} (I^{(gt)}_{(x,y)})^2}{\sum_{(x,y)\in \boldsymbol{I}} (I^{(gt)}_{(x,y)} - I^{(t)}_{(x,y)})^2}, \tag{23}$$

where $\boldsymbol{\Phi}$ is the set of parameters in an algorithm and, here, the number of iterations is viewed as another parameter. As the ground-truth of a noisy image is available, this measure converts parameter tuning into an optimization problem, which is highly consistent with a parameter tuning procedure for anisotropic diffusion suggested in [6]. Thus, parameter tuning in our simulations becomes finding a solution:

$$(t^*, \boldsymbol{\Phi}^*) = \arg\max_{(t,\boldsymbol{\Phi})} \mathrm{SNR}(t, \boldsymbol{\Phi}).$$

Note that the optimality definition is based on theoretical studies as follows: diffusive smoothing generally has the property of causality; no spurious detail should be generated going from finer to coarser scale corresponding to temporal evolution, and a diffusion algorithm generally converges to a uniform intensity image [3], [16]. It is also worth mentioning that the use of such a measure also gives us an insight into the selection of two tunable parameters, S and h, in our algorithm, which will be discussed later on.

4.2 Comparative Results

Fig. 2a shows a noise-free synthetic image containing a set of circles of different sizes and the boundary sharpness varies considerably around different circles. The intensities of four homogeneous regions are 20, 75, 150, and 235,

respectively. Fig. 2b shows a noisy version of this image corrupted by Gaussian noise [7]. The noisy image shown in Fig. 2b is challenging to an adaptive smoothing algorithm with only a local discontinuity measure, e.g., the AS and ADS algorithms, because weak edges, e.g., boundaries of the smallest circle, may either be detected along with many noisy details in a small scale or hardly be detected at all in a large scale, as illustrated in Fig. 2c, where its spatial gradient map is shown. Fig. 2d depicts the optimal result with an SNR of 66.3 by the adaptive hypergraph model [47], which will be discussed later on. Figs. 2e, 2f, 2g, and 2h show a set of snapshots for four adaptive smoothing algorithms. In each column, the upper image corresponds to an optimal snapshot and the lower one is the snapshot after $1,000$ iterations. Fig. 2i shows evolution of SNRs corresponding to different algorithms during smoothing. It is evident from the comparative results that our algorithm outperforms the others in restoration of the noisy image. In terms of SNRs and the snapshots after $1,000$ iterations, the optimal result by our algorithm only slightly alters, whereas the results by the other three adaptive smoothing algorithms evolve towards a uniform intensity image and some important features, e.g., boundaries of the smallest circle, have been smoothed out or severely blurred.

Fig. 3a is another noisy version of Fig. 2a generated by further adding salt-and-pepper noise [7] to the image in Fig. 2b. The composite noise leads to a more challenging image for an adaptive smoothing algorithm. It is well-known that median filters are particularly effective in the presence of salt-and-pepper noise. Hence, we first apply a set of median filters and adaptive hypergraph models [47] of different windows or neighborhood orders to the image. In light of (23), Figs. 3b and 3c shows the optimal results by the median filter with the SNR of 53.2 and the adaptive hypergraph model with the SNR of 41.1. Figs. 3d, 3e, 3f, and 3g show a set of snapshots for four adaptive smoothing algorithms arranged in the same order as Fig. 2. By comparison, our algorithm manages to smooth out both Gaussian and impulse noise and is evidential by a considerable SNR gain from 8.25 to 187.45 (see also discussions later on). However, other algorithms fail to completely remove noise and gains of their SNRs are far lower than that of our algorithm. Snapshots corresponding to $1,000$ iterations along with evolution of SNRs depicted in Fig. 3i show that, to a great extent, our algorithm maintains the optimal snapshot in a wide range of iterations.

Fig. 4a shows a benchmark image, *Pepper*, where there are different regions including large homogeneous regions and fine details. For test, we add Gaussian noise to this benchmark image and the noisy version is shown in Fig. 4b. We apply five algorithms to the noisy image and illustrate resultant images in Figs. 4c, 4d, 4e, 4f, and 4g. By the adaptive hypergraph model [47] we achieve the optimal resultant image with the SNR of 87.3 as illustrated in Fig. 4c. For four adaptive smoothing algorithms, it is observed that, except the AS algorithm, the other three adaptive smoothing algorithms produce reasonably good performance after a number of iterations as illustrated in their optimal snapshots (the upper image in each column); important features including most of fine details are preserved, while noise is readily removed. It should be mentioned that the AS algorithm fails to yield the satisfactory performance; noise still survives in its optimal shot. After $1,000$ iterations, however, many important features are missing or blurred for the three adaptive smoothing algorithms used for comparison. In contrast, our algorithm performs well; most of important features are still preserved well even after $1,000$ iterations. The above outcome is clearly confirmed by evolution of SNRs as illustrated in Fig. 4h. The AS algorithm leads to a small SNR gain from 38.74 to 76.28 at the optimal snapshot, while other three algorithms raise SNRs to at least 224.36. Although the SNR of the EEDS algorithm at the optimal snapshot slightly outperforms ours, further, few iterations make its SNRs drop quickly. In contrast, our algorithm does not suffer from such a quick SNR drop and is capable of maintaining a reasonably high SNR even after 500 iterations, as seen in Fig. 4g.

4.3 Extraction of Hydrographic Objects

As high resolution satellite images are increasingly available, the effective image processing techniques are highly demanded to provide *up-to-date* and *accurate* geographic information. For example, the United States Geological Survey (USGS) is seeking effective strategies to improve map revision capabilities based on satellite images [51]. An ultimate goal is to automatically extract semantically meaningful objects with a certain accuracy from images for map revision [52]. Satellite images are often corrupted by noise from various sources. In general, pixels within a semantically meaningful entity are not homogeneous, while pixels belonging to different entities may have similar attributes. As a result, most of traditional segmentation techniques have limited success and other image processing techniques like smoothing are incorporated into a segmentation method in order to extract accurate geographic information [49].

Extracting a hydrographic object refers to grouping the pixels corresponding to a water body, e.g., river, together and putting other objects into the background, which can be viewed as a special image segmentation task. Hydrographic objects tend to be more homogeneous in comparison with other kinds of objects. As a region growing algorithm is applied, we can take advantage of this property to generate seeds so that only pixels belonging to hydrographic objects can be identified as seeds, and other objects are naturally put into a background. The images used in our simulations are provided by the USGS. These high resolution satellite images are processed using a nonlinear transformation to compensate for variations in actual pixel sizes on the ground due to perspective projection. In addition, a set of original topographic maps are available for some areas so that we can compare the extracted hydrographic objects with the corresponding maps, the relative ground-truth, for testing our algorithm. Incorporated by a simple region growing algorithm [7], our smoothing algorithm has been applied to hydrographic object extraction. In our simulations, we fix the parameters used in our algorithm for all the images and only 20 iterations are run for smoothing.

Fig. 5a shows a satellite image, where intensity values and local attributes change considerably. Fig. 5b shows the smoothing result, and Fig. 5c shows the extraction result. To facilitate comparisons, we display the river in Fig. 5c by marking it as white and superimposing it on the original image. For evaluating our result, Fig. 5d shows the corresponding part of the USGS 1:24,000 topographic map. The simulation result shows that the boundaries of small islands are localized accurately even though they are covered by forests. Similarly, the forests along the river

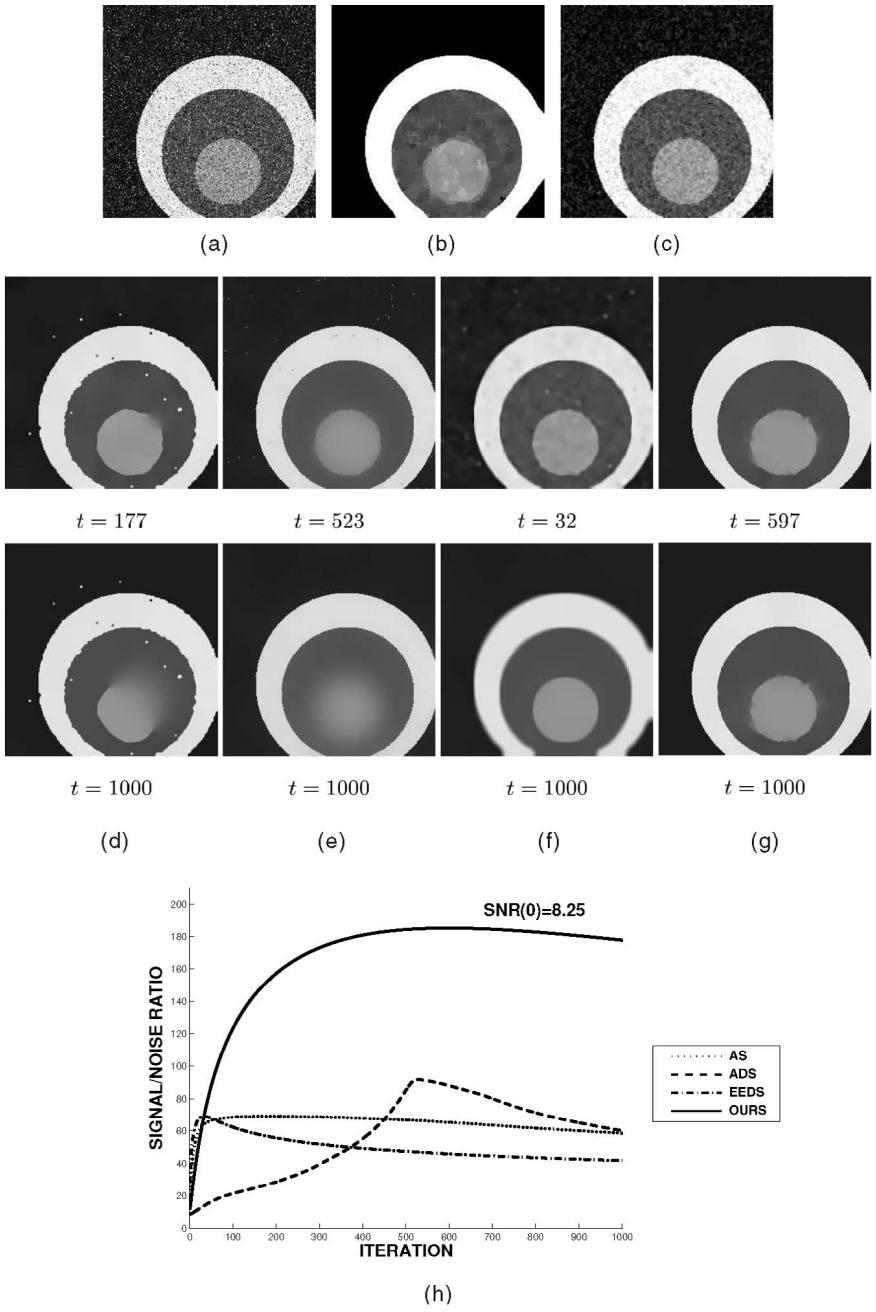

Fig. 3. Smoothing results for a synthetic image. (a) A noisy image by adding salt-and-pepper noise with probabilities $P_a = 0.03$ and $P_b = 0.03$ [7]. (b) Optimal result by the median filter. (c) The hypergraph algorithm. (d) The AS algorithm with $k = 12.0$. (e) The ADS algorithm with $K = 7.0$. (f) The EEDS algorithm with $\lambda = 4.5$ and $\sigma = 2.0$. (g) Ours with $h = 0.125$ and $S = 34.6$. (h) SNR evolution.

banks are preserved well. A bridge connects the lake and the river is also preserved. As shown in Fig. 5a, the bridge is spatially small and it would be very difficult for nonlinear smoothing algorithms to preserve this cartographic feature. Furthermore, we can see that hydrographic regions have changed from the map by comparing Fig. 5d with Fig. 5a. Note, for example, the lower part of the left branch. This geographical change illustrates the constant nature of such changes and the need for frequent map revision. With precise region boundaries achieved, we trust that our method is suited for frequent map revision purposes. While the major features are the same, the lake has shrunk in size, and such shrinkage is captured by our algorithm (see Fig. 5c). This suggests that our method can be used for monitoring changes of hydrographic features.

Fig. 6a shows another satellite image, where almost all the boundaries are irregular and the pixels within a hydrographic object are not homogeneous. Fig. 6b shows the smoothing image, where the pixels within a hydrographic object tend to be homogeneous and their boundaries are accurately preserved. Fig. 6c shows the extraction result. We observe that the major hydrographic objects are accurately extracted even along the noisy banks and, in particular, a small narrow branch in the middle of the image is extracted as well. Moreover, important details are preserved, such as the small islands in the lake at the left of the image. Fig. 6d shows the corresponding part of the

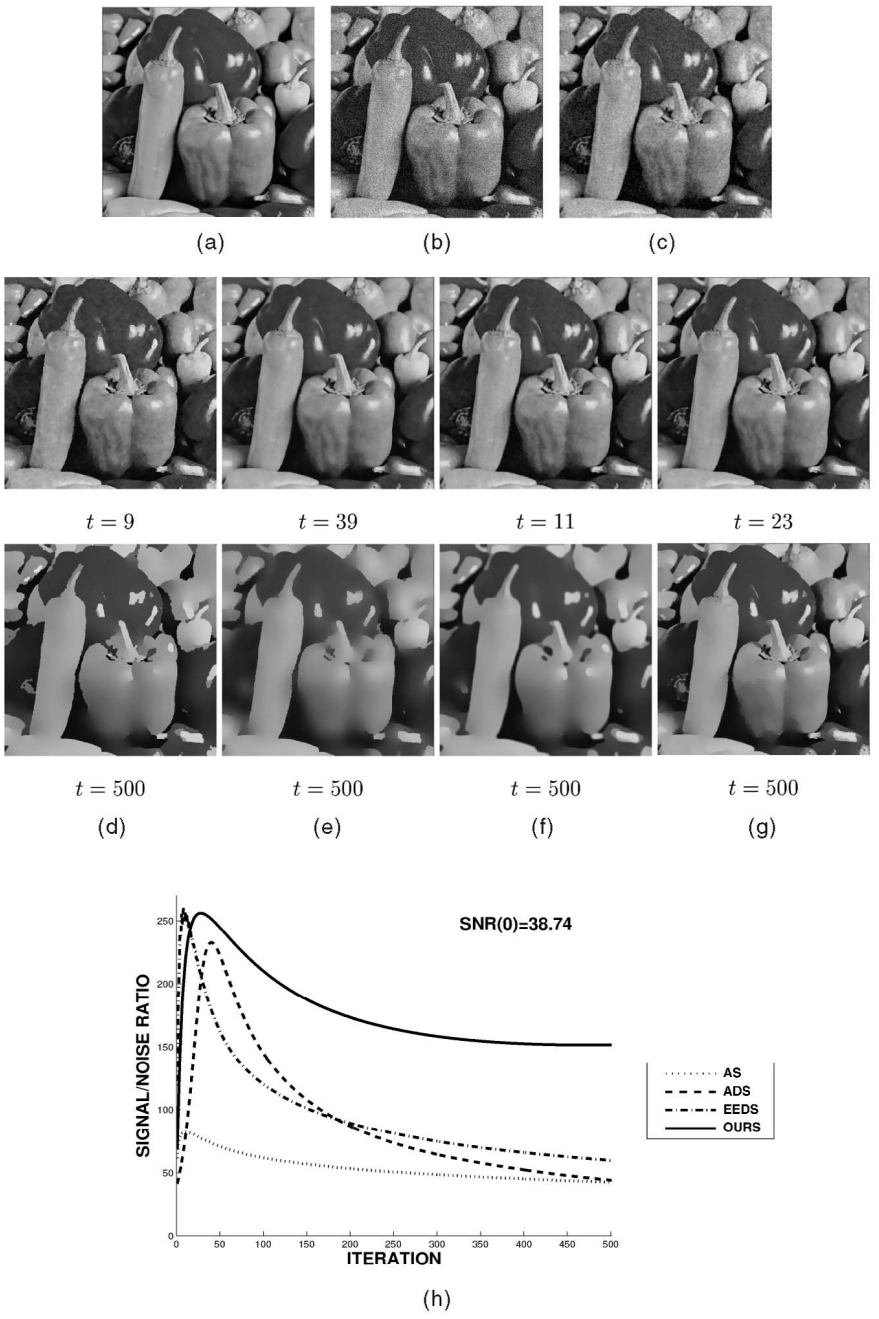

Fig. 4. Smoothing results for a benchmark image. (a) The original image consisting of 512×512 pixels. (b) A noisy version of the image in (a) with Gaussian noise of $\mu = 0$ and $\sigma = 20$. (c) The hypergraph algorithm. (d) The AS algorithm with $k = 7.5$. (e) The ADS algorithm with $K = 5.5$. (f) The EEDS algorithm with $\lambda = 1.0$ and $\sigma = 0.5$. (g) Ours with $h = 0.091$ and $S = 18.5$. (h) SNR evolution.

USGS 1:24,000 topographic map. The alignment between Figs. 6c and 6d indicates that all the important hydrographic objects extracted match those in the map very well.

Here, we mention that, without the smoothing, their performance of most region-growing algorithms [7] is rather poor; the boundaries of water bodies are missing and some water bodies are unexpectedly oversegmented so that the correct water bodies fail to be extracted. In contrast, the above simulation results, as well as others not reported here due to limited space, show that the extracted regions are comparable with existing topographic maps.

Unlike traditional map revision methods based on aerial photogrammetry [52], the use of our smoothing algorithm and a simple growing algorithm offers an effective and efficient underpinning technique to develop a map revision system. For a large image, such a system automatically partitions it into a set of image patches of the same size, e.g., 640×640. All patches can be processed independently and then all the extraction results are seamlessly collected to form a segmented image by an algorithm [49]. An alignment of the segmented image with its corresponding topographic map reveals any changes of hydrographic objects (see Figs. 5c and 5d, for example), which provides an alternative yet efficient computational procedure for map revision on a general-purpose computer [49], [52].

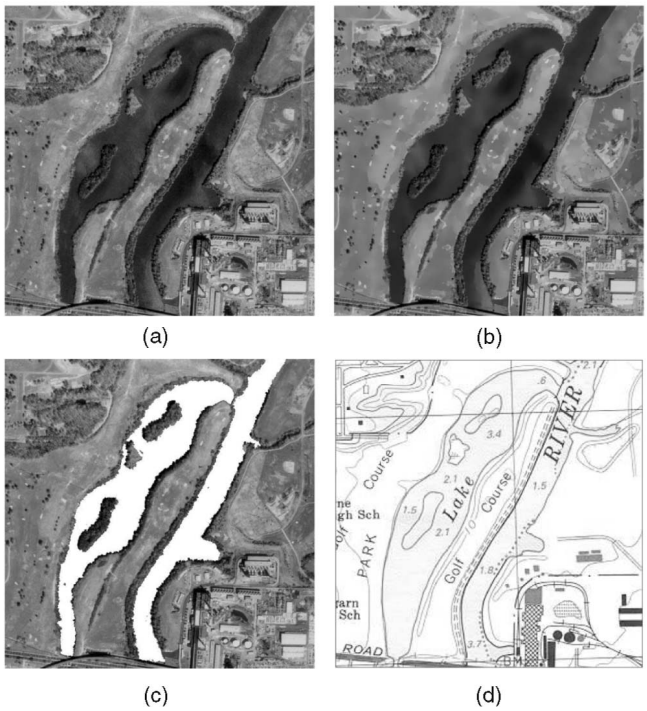

Fig. 5. The result of extracting hydrographic objects for a satellite image. (a) Raw image consisting of 640×640 pixels. (b) The smoothed image. (c) The extraction result. (d) A topographic map corresponding to the image in (a).

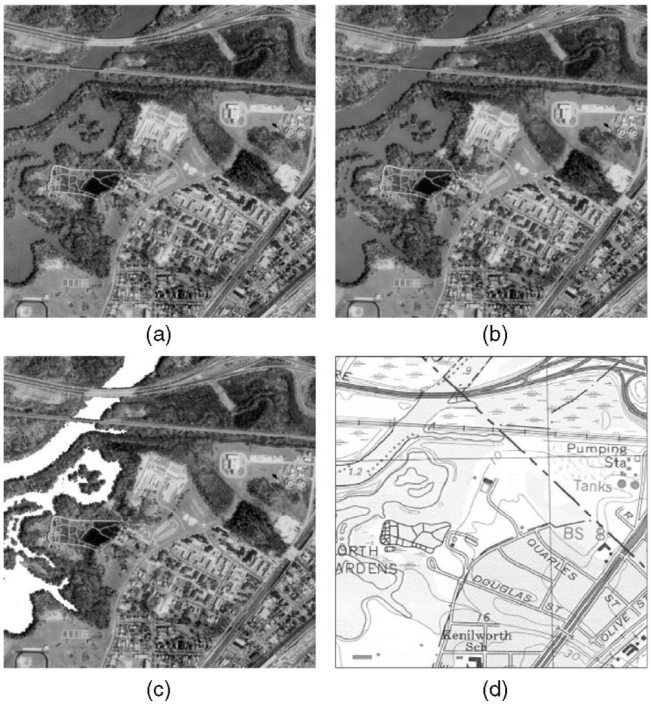

Fig. 6. The result of extracting hydrographic objects for a satellite image. (a) Raw image consisting of 640×640 pixels. (b) The smoothed image. (c) The extraction result. (d) A topographic map corresponding to the image in (a).

5 Discussions

In our algorithm, there are two tunable parameters S and h whose values critically determine its performance. As argued previously based on (17), they are responsible for setting up thresholds in order to identify important features from potential discontinuities detected by two different measures. As our algorithm is applied in practice, how to tune the parameters would be a nontrivial issue. Fortunately, the SNR objective function in (23) allows us to have an empirical study based on the images whose ground-truth is known or available. By an exhausted search, we always find a set of parameters which yields the optimal performance for a given image. Our simulations on various images including those not reported here uncover that the optimal value of S is always around the mean of local intensity differences on homogeneous regions, μ_d, in (7) while the optimal value of h is around the mean of the lowest 20 percent inhomogenity values across a whole image. This suggests that we can fix two parameters in our algorithm once we have measured its inhomogeneity for a given image. In fact, parameters used in our simulations on satellite images are actually selected in this way, which results in the satisfactory performance (cf., Figs. 5 and 6).

In general, there is an implicit condition for adaptive smoothing; i.e., an image has a piecewise constant structure. Therefore, an adaptive smoothing algorithm often works well in the presence of noise subject to the uniformed or Gaussian distribution but fails for images violating the piecewise constant condition, e.g., those full of severe salt-and-pepper noise or irregular texture. The use of two complementary discontinuities enables our algorithm to remove salt-and-pepper noise to a certain extent. Now, we analyze the behavior of our algorithm in the presence of impulse noise based on (17). Although a local discontinuity measure, e.g., spatial gradient, indicates a significant discontinuity in the presence of impulse noise, a contextual discontinuity measure, e.g., inhomogeneity, may not report the same due to its ensemble nature, in particular, as impulse noise appears inside a homogeneous region. Under such a circumstance, there are $|\nabla \mathbf{I}^{(t)}_{(x,y)}| > S$ but $\hat{H}_{(x,y)} < h$. Thus it is likely that the sign of

$$\frac{\partial}{\partial t}\left(\frac{\partial I^{(t)}_{(x,y)}}{\partial x}\right)$$

in (17) is negative, which eventually leads to the effect of impulse noise removal.

Unlike adaptive smoothing, there are alternative smoothing techniques for impulsive noise removal without iterations, e.g., order-statistics filters [7]. Hypergraph imaging model [48] was proposed based on the finite combinatorial set concepts and has been applied in image analysis. Derived from the aforementioned theory, an adaptive hypergraph model [47] was recently developed for impulsive noise detection and removal with a subsequent processing. By observations from results in Figs. 2, 3, and 4 and others not reported here, we found that the hypergraph model effectively filters out impulsive noise, not only salt-and-pepper noise but also high frequency components of Gaussian noise. In essence, this hypergraph model is an impulsive noise detector with a certain spatial scale, i.e., the neighborhood order [47], and actually never alters any properties of the image, which distinguishes from other smoothing-like methods. This salient feature would make this hypergraph model become another discontinuity

measure to overcome the weakness of adaptive smoothing. However, finding a proper neighborhood order for a given image inevitably incurs a parameter tuning process and our simulations indicate that this parameter critically determines its performance of the hypergraph model. As argued by Saha et al. [36] (see also discussions below), there are miscellaneous local structures in a given image that indicate the intrinsic nature of the image. Therefore, this hypergraph model should be extended by the use of multiple neighborhood orders corresponding to appropriate local scales. Our preliminary simulations not reported here suggest that such an extension based on the automatic local scale selection procedure described in (5)-(8) can lead to better results in impulsive noise detection. As a consequence, we would regard such an improved adaptive hypergraph model as another discontinuity measure responsible for impulsive noise detection and somehow incorporate it into our algorithm in our ongoing work.

Spatial scale or size of window for operations in image processing is one of the most important factors in determining resultant images. For adaptive smoothing, most of algorithms adopt only a local discontinuity measure, e.g., spatial gradient, in a 3×3 window to detect discontinuities. As noise substantially appears in an image, these algorithms fail to distinguish significant discontinuities from noise due to overlocality. Using a window of larger size for discontinuity detection, the SUSAN smoothing algorithm overcomes the weakness but encounters a new problem on how to select a proper spatial scale [15]. On the other hand, many spatial regularization methods have been proposed to tackle the overlocality problem, where a Gaussian filter of larger scale is used isotropically to remove noise prior to estimation of local discontinuities [17], [18], [19], [25], [33]. Thus, two different spatial scales may be used in adaptive smoothing. Although doing so indeed alleviates the noise influence, these algorithms may blur discontinuities (for instance, see simulation results of the EEDS algorithm [33] in Section 4) and lead to a higher computational complexity since the Gaussian filter of large scale has to operate at every iteration. Alternatively, Meer et al. employed spatial variance as a discontinuity measure in their multiresolution adaptive image smoothing algorithm [50], where three local windows of different sizes were used simultaneously to estimate spatial variance at every iteration. Undoubtedly, their algorithm benefits from the use of multiple spatial scales but suffers from a high computational burden. Apart from the expensive computation, a fatal weakness of the aforementioned algorithms using multiple spatial scales is a lack of consideration on local structures in a given image. As argued by Saha et al. [36], there are miscellaneous local structures in a given image that indicate the intrinsic nature of the image. Therefore, those miscellaneous local structures underlying an image can be detected only by image-specific spatial scales. Although multiple scales are used in aforementioned algorithms, an individual spatial scale is uniformly used everywhere, which simply leads to the *image-independent* multiscale smoothing.

Local scale control is inevitable for exploring the intrinsic nature of an image for early vision [40]. Recently, Saha and Udupa proposed a scale-based adaptive smoothing algorithm where they explicitly consider the local scale control based on their scale-based affinity theory [42]. In their algorithm, however, the weight for a pixel is determined by a pair-wise intensity difference measure based on a local spatial scale. Since a single pixel may be corrupted by impulse noise, the pair-wise intensity difference may fail to distinguish impulse noise from a significant discontinuity. Like other classical anisotropic diffusion algorithms [33], the sole use of such a single discontinuity measure inevitably causes their algorithm to suffer from the termination problem. In our algorithm, the proposed contextual discontinuity measure, inhomogeneity, derived from their scale-based affinity theory [36], takes advantage of the ensemble properties of coupled pixels, which is insensitive to impulse noise. The contextual discontinuities detected are viewed as a "road map" on local scales for a given image and employed for control of the smoothing speed and feature preservation via cooperating with the local discontinuity measure. Here, we emphasize that the simultaneous use of two discontinuity measures in our algorithm leads to the robust performance and the *image-dependent* multiscale smoothing, which effectively alleviates the termination problem in adaptive smoothing.

Although local scale control yields favorite effects, an algorithm for this purpose often suffers from a high computational complexity. Due to its nature of locally exhausted search, the automatic scale selection for detecting inhomogeneity takes most of the time in our algorithm. In our earlier studies in image segmentation [35], [43], we found that a type of images, e.g., medical images, often have similar statistical properties in spite of miscellaneous contents contained in images. Before our algorithm is applied to a large number of images of the same type, we would first obtain inhomogeneity maps for only a couple of images based on the algorithm in Section 2. Then, we convert achieved inhomogeneity maps into a training set based on their corresponding local structures. A supervised learning algorithm is employed to encode the knowledge. For other images of the same category, their inhomogeneity maps are generated by the learner from image data instead. Previous studies [39], [49] show that such an idea turns out to be useful in determination of parameters for image processing, which, we believe, is a feasible way to speed up our algorithm in applications.

6 Conclusions

We have presented a novel adaptive smoothing algorithm, where two different discontinuity measures are jointly used for noise removal and feature preservation. Moreover, we have shown that the proposed smoothing scheme provides an alternative implementation of the anisotropic diffusion proposed by Perona and Malik [16] and further leads to a constrained anisotropic diffusion process controlled by both contextual and local discontinuities. Simulation results show its effectiveness and demonstrate its potential for real world applications. In contrast to other adaptive smoothing algorithms, our algorithm is relatively immune to the termination problem thanks to the use of scale-based intrinsic contextual discontinuity.

Appendix

Here, we derive the expression of the gradient change of a blurred step edge along the time dimension.

According to the anisotropic diffusion equation in (19), we have

$$\begin{aligned}\frac{\partial I_x^{(t)}}{\partial t} &= \frac{\partial}{\partial x}\left(C_x^{(t)}\frac{\partial I_x^{(t)}}{\partial x}\right)\\ &= C_x^{(t)}\frac{\partial^2 I_x^{(t)}}{\partial x^2}+\frac{\partial C_x^{(t)}}{\partial x}\frac{\partial I_x^{(t)}}{\partial x},\end{aligned} \tag{24}$$

where $C_x^{(t)}$ is specified in (16) and its first derivative with respect to x is

$$\frac{\partial C_x^{(t)}}{\partial x} = -C_x^{(t)}\left(\frac{1}{S^2}\frac{\partial I_x^{(t)}}{\partial x}\frac{\partial^2 I_x^{(t)}}{\partial x^2}+\frac{\hat{H}_x}{h^2}\frac{\partial \hat{H}_x}{\partial x}\right). \tag{25}$$

Inserting (25) into (24) yields

$$\begin{aligned}\frac{\partial I_x^{(t)}}{\partial t} &= C_x^{(t)}\frac{\partial^2 I_x^{(t)}}{\partial x^2}-C_x^{(t)}\frac{\partial I_x^{(t)}}{\partial x}\left(\frac{1}{S^2}\frac{\partial I_x^{(t)}}{\partial x}\frac{\partial^2 I_x^{(t)}}{\partial x^2}+\frac{\hat{H}_x}{h^2}\frac{\partial \hat{H}_x}{\partial x}\right)\\ &= \underbrace{C_x^{(t)}\frac{\partial^2 I_x^{(t)}}{\partial x^2}\left[1-\frac{1}{S^2}\left(\frac{\partial I_x^{(t)}}{\partial x}\right)^2\right]}_{T_1}-\underbrace{C_x^{(t)}\frac{\hat{H}_x}{h^2}\frac{\partial I_x^{(t)}}{\partial x}\frac{\partial \hat{H}_x}{\partial x}}_{T_2}.\end{aligned} \tag{26}$$

Given that the sigmoid function is differentiable, we rewrite the spatial gradient change along the time dimension as

$$\frac{\partial}{\partial t}\left(\frac{\partial I_x^{(t)}}{\partial x}\right) = \frac{\partial}{\partial x}\left(\frac{\partial I_x^{(t)}}{\partial t}\right) = \frac{\partial T_1}{\partial x}-\frac{\partial T_2}{\partial x}. \tag{27}$$

Based on (26) and (27), we derive the expression by considering two terms in (26) separately. Therefore, we have

$$\begin{aligned}\frac{\partial T_1}{\partial x} &= \frac{\partial}{\partial x}\left(C_x^{(t)}\frac{\partial^2 I_x^{(t)}}{\partial x^2}\left[1-\frac{1}{S^2}\left(\frac{\partial I_x^{(t)}}{\partial x}\right)^2\right]\right)\\ &= C_x^{(t)}\frac{\partial^3 I_x^{(t)}}{\partial x^3}\left[1-\frac{1}{S^2}\left(\frac{\partial I_x^{(t)}}{\partial x}\right)^2\right]\\ &\quad+\frac{\partial^2 I_x^{(t)}}{\partial x^2}\frac{\partial}{\partial x}\left\{C_x^{(t)}\left[1-\frac{1}{S^2}\left(\frac{\partial I_x^{(t)}}{\partial x}\right)^2\right]\right\}\\ &= C_x^{(t)}\frac{\partial^3 I_x^{(t)}}{\partial x^3}\left[1-\frac{1}{S^2}\left(\frac{\partial I_x^{(t)}}{\partial x}\right)^2\right].\end{aligned} \tag{28}$$

In the last step, we apply the property of the sigmoid function at the point of inflection, $\partial^2 I_x/\partial x^2 = 0$. Similarly, we have

$$\begin{aligned}\frac{\partial T_2}{\partial x} &= \frac{\partial}{\partial x}\left(C_x^{(t)}\frac{\hat{H}_x}{h^2}\frac{\partial I_x^{(t)}}{\partial x}\frac{\partial \hat{H}_x}{\partial x}\right)\\ &= \frac{\partial I_x^{(t)}}{\partial x}\frac{\partial}{\partial x}\left(C_x^{(t)}\frac{\hat{H}_x}{h^2}\frac{\partial \hat{H}_x}{\partial x}\right)+\frac{\partial^2 I_x^{(t)}}{\partial x^2}\left(C_x^{(t)}\frac{\hat{H}_x}{h^2}\frac{\partial \hat{H}_x}{\partial x}\right)\\ &= \frac{\partial I_x^{(t)}}{\partial x}\frac{\partial}{\partial x}\left(C_x^{(t)}\frac{\hat{H}_x}{h^2}\frac{\partial \hat{H}_x}{\partial x}\right)\\ &= -C_x^{(t)}\frac{\partial I_x^{(t)}}{\partial x}\left(\frac{\hat{H}_x}{h^2}\frac{\partial \hat{H}_x}{\partial x}\right)^2+C_x^{(t)}\frac{\partial I_x^{(t)}}{\partial x}\frac{\partial}{\partial x}\left(\frac{\hat{H}_x}{h^2}\frac{\partial \hat{H}_x}{\partial x}\right)\\ &= -C_x^{(t)}\frac{\partial I_x^{(t)}}{\partial x}\left(\frac{\hat{H}_x}{h^2}\frac{\partial \hat{H}_x}{\partial x}\right)^2\\ &\quad+C_x^{(t)}\frac{\partial I_x^{(t)}}{\partial x}\left[\frac{\hat{H}_x}{h^2}\frac{\partial^2 \hat{H}_x}{\partial x^2}+\frac{1}{h^2}\left(\frac{\partial \hat{H}_x}{\partial x}\right)^2\right]\\ &= -\frac{C_x^{(t)}}{h^2}\frac{\partial I_x^{(t)}}{\partial x}\left[(\hat{H}_x^2-h^2)\left(\frac{\partial \hat{H}_x}{\partial x}\right)^2-\hat{H}_x\frac{\partial^2 \hat{H}_x}{\partial x^2}\right].\end{aligned} \tag{29}$$

Assembling (28) and (29) in (27), we obtain

$$\begin{aligned}\frac{\partial}{\partial t}\left(\frac{\partial I_x^{(t)}}{\partial x}\right) &= C_x^{(t)}\frac{\partial^3 I_x^{(t)}}{\partial x^3}\left[1-\frac{1}{S^2}\left(\frac{\partial I_x^{(t)}}{\partial x}\right)^2\right]\\ &\quad+\frac{C_x^{(t)}}{h^2}\frac{\partial I_x^{(t)}}{\partial x}\left[(\hat{H}_x^2-h^2)\left(\frac{\partial \hat{H}_x}{\partial x}\right)^2-\hat{H}_x\frac{\partial^2 \hat{H}_x}{\partial x^2}\right].\end{aligned}$$

ACKNOWLEDGMENTS

The partial work reported in this manuscript was done while the author was with The Ohio State University. The author is grateful to D.L. Wang and X.W. Liu for their valuable discussions and the anonymous reviewers for their useful comments. The author would also like to thank The Center for Mapping at The Ohio State University for providing the USGS satellite image database used in the simulations.

REFERENCES

[1] D. Marr, *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information.* San Francisco: W.H. Freeman, 1982.
[2] S. Geman and D. Geman, "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 6, pp. 721-741, 1984.
[3] J. Koenderink, "The Structure of Images," *Biological Cybernetics,* vol. 50, pp. 363-370, 1984.
[4] A. Blake and A. Zisserman, *Visual Reconstruction.* Cambridge, Mass.: MIT Press, 1987.
[5] P. Saint-Marc, J.S. Chen, and G. Medioni, "Adaptive Smoothing: A General Tool for Early Vision," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 13, pp. 514-529, 1991.
[6] J. Weickert, "A Review of Nonlinear Diffusion Filtering," *Scale-Space Theory in Computer Vision,* B. Romeny, L. Florack, J. Koenderink, and M. Viergever, eds., pp. 3-28, Berlin: Springer, 1997.
[7] R.C. Gonzalez and R.E. Woods, *Digital Image Processing,* second ed., Upper Saddle River, N.J.: Prentice Hall, 2002.
[8] D.A. Forsyth and J. Ponce, *Computer Vision: A Model Approach.* Upper Saddle River, N.J.: Prentice Hall, 2003.
[9] A. Lev, S. Zucker, and A. Rosenfeld, "Iterative Enhancement of Noisy Images," *IEEE Trans. Systems, Man, and Cybernetics,* vol. 7, pp. 435-442, 1977.
[10] D.C.C. Wang, A.H. Vagnucci, and C.C. Li, "Gradient Inverse Weighted Smoothing Scheme and the Evaluation of Its Performance," *Computer Vision, Graphics, and Image Processing,* vol. 15, pp. 167-181, 1981.
[11] G.A. Mastin, "Adaptive Filters for Digital Image Noise Smoothing: An Evaluation," *Computer Vision, Graphics, and Image Processing,* vol. 31, pp. 103-121, 1985.
[12] C.H. Li and C.K. Lee, "Image Smoothing Using Parametric Relaxation," *Graphical Models and Image Processing,* vol. 31, pp. 161-174, 1995.
[13] M.I. Gürelli and L. Onural, "A Class of Adaptive Directional Image Smoothing Filters," *Pattern Recognition,* vol. 29, pp. 1995-2004, 1996.
[14] R. Legault and C.Y. Suen, "Optimal Local Weighted Averaging Methods in Contour Smoothing," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 19, pp. 801-817, 1997.
[15] S.M. Smith and J.M. Brady, "SUSAN—A New Approach to Low Level Image Processing," *Int'l J. Computer Vision,* vol. 23, pp. 45-78, 1997.
[16] P. Perona and J. Malik, "Scale-Space and Edge Detection Using Anisotropic Diffusion," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 12, pp. 629-639, 1990.
[17] F. Catté, P.L. Lions, J.M. Morel, and T. Coll, "Image Selective Smoothing and Edge Detection by Nonlinear Diffusion," *SIAM J. Numerical Analysis,* vol. 29, pp. 182-193, 1992.
[18] L. Alvarez, P.L. Lions, and J.M. Morel, "Image Selective Smoothing and Edge Detection by Nonlinear Diffusion II," *SIAM J. Numerical Analysis,* vol. 29, pp. 845-866, 1992.

[19] M. Nitzberg and T. Shiota, "Nonlinear Image Filtering with Edge and Corner Enhancement," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 14, pp. 826-833, 1992.

[20] R.T. Whitaker and S.M. Pizer, "A Multi-Scale Approach to Non-Uniform Diffusion," *CVGIP: Image Understanding,* vol. 57, pp. 99-110, 1993.

[21] L. Alvarez and L. Mazorra, "Signal and Image Restoration Using Shock Filters and Anisotropic Diffusion," *SIAM J. Numerical Analysis,* vol. 31, pp. 590-605, 1994.

[22] J. Kacur and K. Mikula, "Solution of Nonlinear Diffusion Appearing in Image Smoothing and Edge Detection," *Applied Numerical Math.,* vol. 50, pp. 47-59, 1995.

[23] Y.L. You, W.Y. Xu, A. Tannenbaum, and M. Kaveh, "Behavioral Analysis of Anisotropic Diffusion in Image Processing," *IEEE Trans. Image Processing,* vol. 5, pp. 1539-1553, 1996.

[24] R.A. Carmona and S. Zhong, "Adaptive Smoothing Respecting Feature Directions," *IEEE Trans. Image Processing,* vol. 7, pp. 353-358, 1998.

[25] J. Weickert, B.M. ter Haar Romeny, and M.A. Viergever, "Efficient and Reliable Schemes for Nonlinear Diffusion Filtering," *IEEE Trans. Image Processing,* vol. 7, pp. 398-410, 1998.

[26] M.J. Black, G. Sapiro, D.H. Marimont, and D. Heeger, "Robust Anisotropic Diffusion," *IEEE Trans. Image Processing,* vol. 7, pp. 421-432, 1998.

[27] J. Monteil and A. Beghdadi, "A New Interpretation and Improvement of the Nonlinear Anisotropic Diffusion for Image Enhancement," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 21, pp. 940-946, 1999.

[28] S.T. Acton, "Locally Monotonic Diffusion," *IEEE Trans. Signal Processing,* vol. 48, pp. 1379-1389, 2000.

[29] P.K. Saha and J.K. Udupa, "Scale-Based Diffusive Image Filtering Preserving Boundary Sharpness and Fine Structures," *IEEE Trans. Medical Imaging,* vol. 20, pp. 1140-1155, 2001.

[30] D. Barash, "A Fundamental Relationship between Bilateral Filtering, Adaptive Smoothing, and the Nonlinear Diffusion Equation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 24, pp. 844-847, 2002.

[31] A.C. Shih, H.Y. Liao, and C.H. Liu, "A New Iterated Two-Band Diffusion Equation: Theory and Its Application," *IEEE Trans. Image Processing,* vol. 12, pp. 466-476, 2003.

[32] B.M. ter Harr Romeny, *Geometry Driven Diffusion in Computer Vision.* Boston: Kluwer, 1994.

[33] J. Weickert, "Theoretical Foundations of Anisotropic Diffusion in Image Processing," *Computing,* vol. 11, pp. 221-236, 1996.

[34] J. Weickert, "Applications of Nonlinear Diffusion in Image Processing and Computer Vision," *Acta Mathematica Universitatis Comenianae,* vol. 70, pp. 33-50, 2001.

[35] K. Chen, D.L. Wang, and X.W. Liu, "Weight Adaptation and Oscillatory Correlation for Image Segmentation," *IEEE Trans. Neural Networks,* vol. 11, pp. 1106-1123, 2000.

[36] P.K. Saha, J.K. Udupa, and D. Odhner, "Scale-Based Fuzzy Connected Image Segmentation: Theory, Algorithms, and Validation," *Computer Vision and Image Understanding,* vol. 77, pp. 145-174, 2000.

[37] J. Crespo and R.W. Schafer, "Edge-Based Adaptive Smoothing," *Optical Eng.,* vol. 36, pp. 3081-3092, 1997.

[38] X. Li and T. Chen, "Nonlinear Diffusion with Multiple Edginess Thresholds," *Pattern Recognition,* vol. 27, pp. 1029-1037, 1994.

[39] B. Fischl and E.L. Schwartz, "Learning an Integral Equation Approximation to Nonlinear Anisotropic Diffusion in Image Processing," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 19, pp. 342-353, 1997.

[40] J.H. Elder and S.W. Zucker, "Local Scale Control for Edge Detection and Blur Estimation," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 20, pp. 699-716, 1998.

[41] P. Liang and Y.F. Wang, "Local Scale Controlled Anisotropic Diffusion with Local Noise Estimate for Image Smoothing and Edge Detection," *Proc. IEEE Int'l Conf. Computer Vision,* pp. 193-200, 1998.

[42] P.K. Saha and J.K. Udupa, "Optimum Image Thresholding via Class Uncertainty and Region Homogeneity," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 23, pp. 689-706, 2001.

[43] K. Chen and D.L. Wang, "A Dynamically Coupled Neural Oscillator Network for Image Segmentation," *Neural Networks,* vol. 15, pp. 423-439, 2002.

[44] T. Lindeberg and B.M. ter Haar Romeny, "Linear Scale-Space," *Geometry-Driven Diffusion in Computer Vision,* B.M. ter Haar Romeny, ed., pp. 1-41, Dordrecht: Kluwer Academic, 1994.

[45] F. Torkamani-Azar and K.E. Tait, "Image Recovery Using the Anisotropic Diffusion Equation," *IEEE Trans. Image Processing,* vol. 5, pp. 1573-1578, 1996.

[46] S.T. Acton, "Multigrid Anisotropic Diffusion," *IEEE Trans. Image Processing,* vol. 7, pp. 280-291, 1998.

[47] S. Rital, A. Bretto, D. Aboutajdine, and H. Cherifi, "Application of Adaptive Hypergraph Model to Impulsive Noise Detection," *Lecture Notes in Computer Science 2124,* W. Skarbek ed., pp. 555-562, Berlin: Springer-Verlag, 2001.

[48] A. Bretto, H. Cherifi, and D. Aboutajdine, "Hypergraph Imaging: An Overview," *Pattern Recognition,* vol. 35, pp. 651-658, 2002.

[49] X.W. Liu, K. Chen, and D.L. Wang, "Extraction of Hydrographic Regions from Remote Sensing Images Using an Oscillator Network with Weight Adaptation," *IEEE Trans. Geoscience and Remote Sensing,* vol. 39, pp. 207-211, 2001.

[50] P. Meer, R.H. Park, and K. Cho, "Multi-Resolution Adaptive Image Smoothing," *CVGIP: Graphical Models and Image Processing,* vol. 56, pp. 140-148, 1994.

[51] USGS, "U.S. Geological Survey Strategic Plan: 1997—2003," The United States Geological Survey, 1997, available online at http://www.usgs.gov/strategic/index.html.

[52] L. Moore, "The U.S. Geological Survey's Revision Program for 7.5-Minute Topographic Maps," The United States Geological Survey Open-File Report 00-325, 2000.

Ke Chen received the BS and MS degrees from Nanjing University in 1984 and 1987, respectively, and the PhD degree from the Harbin Institute of Technology in 1990, all in computer science. He has been with The University of Manchester since 2003, where he is now a senior lecturer in computer science. He was with Birmingham University, Peking University, Ohio State University, Kyushu Institute University, and Tsinghua University. He was a visiting professor at Microsoft Research Asia in 2000 and Hong Kong Polytechnic University in 2001. He serves as an associate editor of the *IEEE Transactions on Neural Networks* and the *International Journal of Image and Graphics*, a technical program cochair of the 2005 International Conference on Natural Computation, and has been a member of the technical program committee of numerous international conferences. At present, he chairs the IEEE Computational Intelligence Society NNTC Audio & Speech Processing Task Force and the ETTC Multimedia & Biometrics Task Force, and is also the vice chair of the IEEE Computational Intelligence Society UKRI Chapter. He is a recipient of several academic awards including the NSFC Distinguished Principal Young Investigator Award and JSPS Research Award. He is a senior member of the IEEE and a member of the IEEE Computational Intelligence Society. His current research interests include pattern recognition, machine learning, and machine perception.