Weight Adaptation and Oscillatory Correlation for Image Segmentation

Ke Chen, *Member, IEEE*, DeLiang Wang, *Member, IEEE*, and Xiuwen Liu

***Abstract*—We propose a method for image segmentation based on a neural oscillator network. Unlike previous methods, weight adaptation is adopted during segmentation to remove noise and preserve significant discontinuities in an image. Moreover, a logarithmic grouping rule is proposed to facilitate grouping of oscillators representing pixels with coherent properties. We show that weight adaptation plays the roles of noise removal and feature preservation. In particular, our weight adaptation scheme is insensitive to termination time and the resulting dynamic weights in a wide range of iterations lead to the same segmentation results. A computer algorithm derived from oscillatory dynamics is applied to synthetic and real images and simulation results show that the algorithm yields favorable segmentation results in comparison with other recent algorithms. In addition, the weight adaptation scheme can be directly transformed to a novel feature-preserving smoothing procedure. We also demonstrate that our nonlinear smoothing algorithm achieves good results for various kinds of images.**

***Index Terms*—Desynchronization, image segmentation, LEGION, nonlinear smoothing, oscillatory correlation, synchronization, weight adaptation.**

I. INTRODUCTION

IMAGE segmentation refers to the process of partitioning an image into a set of coherent regions. As a major aspect of visual perception, it is central to various kinds of tasks. Although humans often perform it effortlessly, image segmentation is still one of the main hurdles on the path from acquisition to understanding of images for machine vision.

Image segmentation has been extensively studied in computer vision community, and many techniques have been proposed (for reviews of the subject see [12], [13], [23], [43], and [44]). Basically, all of algorithms can be classified into four broad categories: pixel classification, edge-based segmentation, region-based segmentation, and hybrid techniques. Pixel classification, e.g., thresholding, is a technique that associates a pixel with a specific label if a measure of the pixel is within a certain range. This technique has been extended to more complicated forms such as multiple-threshold test [13], [17], and a recursive test [5]. Edge-based segmentation generally consists of two consecutive stages, i.e., edge detection and contour completion, and regions embraced by closed contours produce segmentation results [7], [9], [30]. Region-based techniques operate directly on regions, and classical techniques include region growing and split and merge [1], [14], [26], [44]. The main idea of these methods is to iteratively group (or split) pixels into connected regions in accordance with some prespecified criteria, e.g., homogeneity. Finally, hybrid techniques combine two or more kinds of aforementioned techniques to achieve improved segmentation [2], [4], [11], [24], [43]. Computationally, most of these algorithms are of serial nature [20] though some partially parallel algorithms have been developed [20]–[22]. Due to many uncertain factors in image segmentation [42], good computational solutions are often difficult to obtain [10] and the segmentation problem is, to a great extent, viewed as unsolved.

In image segmentation, unexpected noise often causes poor performance. To tackle the problem, smoothing techniques are widely used prior to segmentation to improve the performance of a segmentation algorithm. Nonlinear smoothing methods have shown their usefulness in facilitating image segmentation, and most of them are of iterative nature. For iterative smoothing, it is well known that the performance highly depends upon the termination time, and these algorithms generally cause the original image to evolve toward a uniform intensity image [15], [25], which we refer to as the termination problem. Therefore, those smoothing techniques are difficult to use in practice.

Neural networks have been successfully applied in pattern recognition [3], [27]. However, relatively little work has been reported on image segmentation, which is generally viewed as part of preprocessing in neural networks. For image segmentation, a nontrivial issue is how to represent the outcome of segmentation. Most of neural network based methods pose image segmentation as pixel classification [16], [23], where a pixel is associated with one class label representing a region or a segment of the given image. In general, however, such a representational mechanism results in the need of prior knowledge of a given image. That is, these classification-based methods require prior knowledge of the number of regions existing in the image. Traditional neural networks, such as associative memories and multilayer perceptrons, are based on attractor dynamics; that is, they respond to an input with a single pattern previously stored. Such networks lack an effective representation for encoding multiple objects simultaneously, and as a result, image segmentation is a challenging problem for neural networks.

Theoretical studies of brain functions and feature binding suggest the mechanism of temporal correlation as a representational framework. According to von der Malsburg [35], an

Manuscript received August 25, 1998; revised February 11, 2000. This work was supported in part by the National Science Foundation under Grant IRI-9 423 312 and a China National Key Fundamental Research Grant G1999032708. The work of D. L. Wang was supported by the ONR Young Investigator Award N00014-96-1-0676.

K. Chen was with the Department of Computer and Information Science and Center for Cognitive Science, The Ohio State University, Columbus, OH 43210 USA. He is now with the National Laboratory of Machine Perception and Center for Information Science, Peking University, Beijing 100871, China (email: chen@cis.pku.edu.cn).

D. L. Wang and X. Liu are with the Department of Computer and Information Science and Center for Cognitive Science, The Ohio State University, Columbus, OH 43210-1277 USA (email: dwang.liux@cis.ohio-state.edu).

Publisher Item Identifier S 1045-9227(00)04301-0.

object is represented by the temporal correlation of the firing activities of the scattered cells encoding different features of an object. Neural oscillations provide a natural encoding scheme for temporal correlation. In such a scheme, each oscillator encodes some features of an object, and each object is represented by a group of synchronized oscillators, whereas different objects are represented by different oscillator groups whose oscillations are desynchronized. This special form of temporal correlation is called *oscillatory correlation* [38]. Furthermore, Terman and Wang proposed *locally excitatory globally inhibitory oscillator networks* (LEGION) [33], [38], and analytically showed that LEGION can rapidly achieve both synchronization in a locally coupled oscillator group and desynchronization among a number of oscillator groups.

LEGION has recently been applied to segment real images [39]. However, their LEGION algorithm is sensitive to noise since an oversimplified local coupling scheme is used. For example, it does not work for the noisy image in Fig. 4(a), which contains four blocks and a background. Real images are inevitably noisy due to the complexity of the physical world and various kinds of noise introduced by imaging apparatus.

In this paper, we propose an image segmentation method using weight adaptation and oscillatory correlation on the basis of LEGION. A weight adaptation scheme is proposed to remove noise and irrelevant features to the given task in images. A logarithmic grouping rule is used to replace both the summation and the maximization rules in [39]. The weight adaptation scheme and the new grouping rule lead to an extended LEGION algorithm. We have applied the algorithm to noisy synthetic images and real images. Simulation results show that our extended LEGION algorithm achieves considerably better performance. On the other hand, the weight adaptation scheme can be directly transformed to a feature-preserving smoothing algorithm. In contrast to recent adaptive smoothing techniques [41], our smoothing algorithm is relatively immune to the termination problem and important features can be preserved in a wide range of iterations. Comparison results demonstrate its effectiveness.

The rest of the paper is organized as follows. Section II briefly reviews the architecture and dynamics of LEGION. Section III presents the weight adaptation scheme, the extended LEGION algorithm, and the feature-preserved selective smoothing algorithm. Section IV reports simulation results, and comparisons are presented in Section V. Further discussions are given in Section VI.

II. LEGION MODEL

In LEGION, each oscillator (i, j) is defined as a feedback loop between an excitatory unit x_{ij} and an inhibitory unit y_{ij} [39]

$$\frac{dx_{ij}}{dt} = f(x_{ij}, y_{ij}) + I_{ij} H(p_{ij} - \theta) + S_{ij} + \rho \tag{1a}$$

$$\frac{dy_{ij}}{dt} = \epsilon g(x_{ij}, y_{ij}) \tag{1b}$$

where $f(x, y) = 3x - x^3 + 2 - y$ and $g(x, y) = \alpha(1 + \tanh(\beta x)) - y$ are used. The detailed forms of f and g do not matter for LEGION to function (for details, see [33], [39]). $H(\cdot)$ is the Heaviside step function, which is defined as $H(\nu) = 1$ if $\nu \geq 0$ and $H(\nu) = 0$ if $\nu < 0$. I_{ij} represents external stimulation to the oscillator and S_{ij} represents overall coupling from the network. The parameter ρ denotes the amplitude of a Gaussian noise term which is introduced to test the robustness of the system and, more importantly, to actively desynchronize different oscillator groups. The parameter ϵ is chosen $0 < \epsilon \ll 1$. In this case, (1) without any coupling or noise, corresponds to a standard relaxation oscillator [34]. The dynamics of a single relaxation oscillator is summarized as follows. To simplify the presentation, we drop all the subscripts. The x-nullcline of (1), $dx/dt = 0$ is a cubic curve, while the y-nullcline, $dy/dt = 0$ is a sigmoid. The parameter β controls the steepness of the sigmoid function and is chosen to be large, $\beta \gg 1$. For $I > 0$, the two nullclines intersect only at the middle branch of the cubic and (1) gives rise to a stable periodic orbit for all sufficiently small values ϵ. In this case, the oscillator is referred to as enabled [see Fig. 1(a)]. The periodic solution alternates between a phase called the *active phase* of the oscillator and a phase called the *silent phase*, respectively. Within these two phases, (1) exhibits near steady state behavior. In contrast, the transition between two phases takes place on a fast time scale. The parameter α is introduced to control the relative times that a stimulated oscillator spends in the two phases. For $I < 0$, the two nullclines intersect on the left branch of the cubic, and (1) produces a stable fixed point as illustrated in Fig. 1(b). In this case, the oscillator is referred to as excitable. Fig. 1(c) illustrates the enabled behavior.

In (1a), the Heaviside term provides a mechanism to distinguish between major objects and noisy fragments. Wang and Terman [39] suggested that a major object must contain at least one oscillator (called a leader), which is located at the center of a large, homogeneous region. A noisy fragment does not contain such an oscillator. Essentially, a leader receives large lateral excitation from its neighborhood. The variable p_{ij} denotes the lateral potential of oscillator (i, j) and, through the threshold θ, determines whether oscillator (i, j) is a leader.

For image segmentation, the LEGION network generally has two-dimensional (2-D) architecture, as illustrated in Fig. 2, where each oscillator corresponds to a pixel in the given image and is connected to its eight nearest neighbors except for the boundaries where there is no wrap around. The global inhibitor is connected with all the oscillators on the 2-D grid. It receives excitation from each oscillator and in turn exerts inhibition to each oscillator.

In LEGION, the overall coupling term S_{ij} in (1a) is defined by

$$S_{ij} = S_{ij}^a - W_z H(z - \theta_z), \tag{2}$$

where S_{ij}^a is the total coupling from the adjacent active neighbors of oscillator (i, j) to be defined later on. W_z is the weight of inhibition from the global inhibitor z, whose activity is defined as

$$\frac{dz}{dt} = \phi(\sigma_\infty - z) \tag{3}$$

where $\sigma_\infty = 0$ if $x_{ij} < \theta_z$ for every oscillator and $\sigma_\infty = 1$ if $x_{ij} \geq \theta_z$ for at least one oscillator. θ_z is a threshold and ϕ is a parameter. The computation of LEGION can be briefly summarized as follows. Once an oscillator enters the active phase,

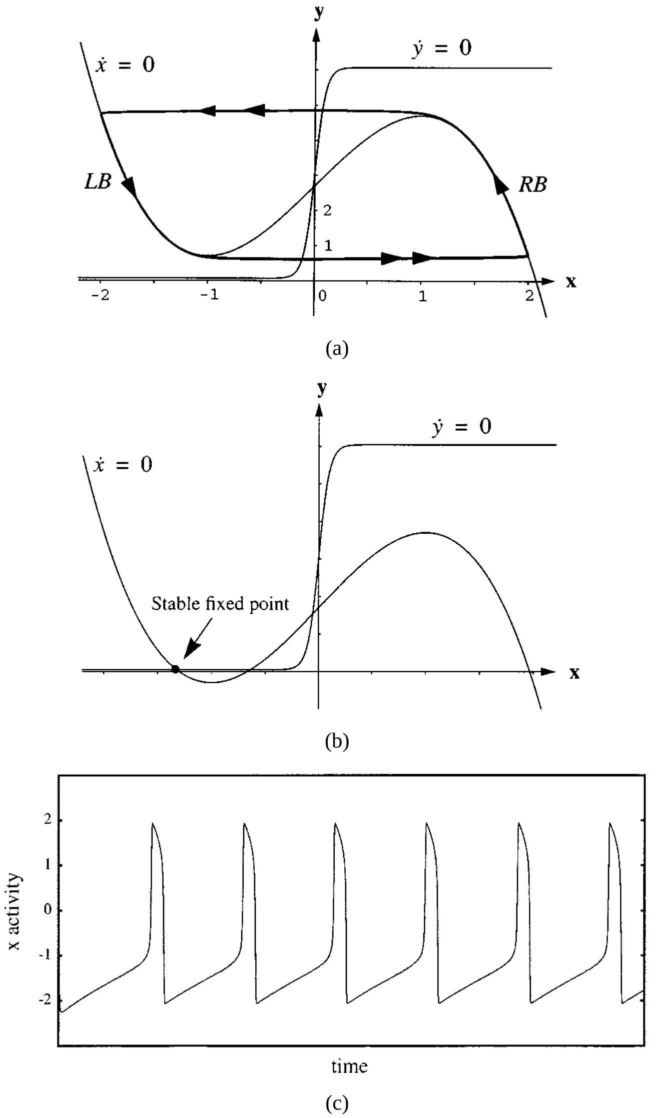


Fig. 1. Nullclines and trajectories of a single relaxation oscillator. (a) Behavior of an enabled oscillator. The bold curve shows the limit cycle of the oscillator, whose direction of motion is indicated by arrowheads with double arrowheads indicating jumping. (b) Behavior of an excitable oscillator. The oscillator approaches a stable fixed point. (c) Temporal activity of the oscillator. The x value of the oscillator is plotted. The parameter values used are $I = 0.8$, $\rho = 0.02$, $\epsilon = 0.04$, $\alpha = 9.0$, and $\beta = 10.0$.

it triggers the global inhibitor. As a result, the global inhibitor attempts to inhibit the entire network. On the other hand, an active oscillator propagates its excitation to its nearest neighbors, which is stronger than the inhibition. The propagation of the activation rapidly continues until all the oscillators representing the same object are active. Thus, the dynamics underlying LEGION is a process of both synchronization by local cooperation through excitatory coupling and desynchronization by global competition via the global inhibitor.

III. WEIGHT ADAPTATION AND EXTENSION OF LEGION ALGORITHM

In this section, we define fixed and dynamic connection weights and present a weight adaptation scheme for noise removal and feature preservation. A new grouping rule is also proposed to achieve better segmentation. Using the weight adaptation scheme and the new grouping rule, we extend the LEGION algorithm for gray-level image segmentation. In addition, we point out that our weight adaptation scheme can be directly transformed to a nonlinear smoothing algorithm.

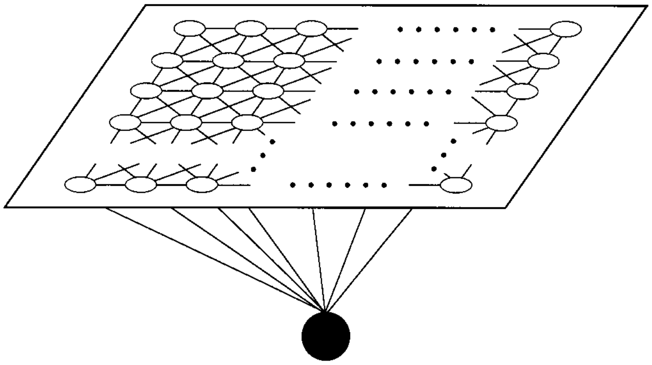


Fig. 2. Architecture of a 2-D LEGION network for image segmentation. An oscillator is represented by an open circle and the global inhibitor is represented by the filled circle.

A. Dynamic Weight Adaptation

To facilitate weight adaptation, two kinds of connections, i.e., *fixed* and *dynamic* connection, are assumed in our neural oscillator network. For a specific oscillator, the fixed connectivity specifies a group of neighboring oscillators connecting to the oscillator. Corresponding to one pixel in the image, the lateral attributes associated with the oscillator can be measured from such an ensemble of fixed connections. On the other hand, the dynamic connectivity encodes the instantaneous relationship between two adjacent oscillators during weight adaptation. Therefore, fixed connection weights are directly determined based on the image, while dynamic connection weights adapt themselves for noise removal and feature preservation.

For oscillator (i, j), the weight of its fixed connection from oscillator (k, l), $T_{ij;kl}$, is defined as the difference between the external stimuli received by (i, j) and (k, l) in its lateral neighborhood, $N(i, j; R)$, i.e.,

$$T_{ij;kl} = I_{kl} - I_{ij}. \tag{4}$$

Here I_{ij} and I_{kl} are the intensities of pixel (i, j) and pixel (k, l), respectively, and $N(i, j; R)$ is defined as

$$\begin{aligned} N(i,j;R) = \{(k,l) \,|\, i - R \le k \le i + R, \\ j - R \le l \le j + R, (k,l) \ne (i,j)\} \end{aligned}$$

where R $(R \ge 1)$ is a parameter that determines the size of the lateral neighborhood. For oscillator (i, j), the fixed connections exist only in $N(i, j; R)$ and $T_{ij;kl} = -T_{kl;ij}$, where $(k, l) \in N(i, j; R)$. On the other hand, a dynamic connection weight from oscillator (k, l) to oscillator (i, j), $W_{ij;kl}$ is defined only within the nearest neighborhood of (i, j) and initialized to the corresponding fixed weight, i.e., $W_{ij;kl}^{(0)} = T_{ij;kl}$, where $(k, l) \in N(i, j; 1)$. Dynamic weights are also anti-symmetrical: $W_{ij;kl}^{(t)} = -W_{kl;ij}^{(t)}$ as we shall see later, and $|W_{ij;kl}^{(t)}|$ encodes the dissimilarity between oscillators (i, j) and (k, l) at time t.

The basic idea in our proposed weight adaptation scheme is to update $W_{ij;kl}$ using all the dynamic weights associated with oscillators (i,j) and (k,l), where $(k,l) \in N(i,j;1)$. For noise removal and feature preservation, intuitively, dynamic weights should be adapted so that the absolute dynamic weight becomes small if two oscillators correspond to two adjacent pixels in a homogeneous region, while the weight becomes large if the two pixels cross a boundary between different homogeneous regions. We observe that most of the discontinuities in a lateral neighborhood correspond to significant features and such discontinuities should remain unchanged and may be used to control the speed of weight adaptation for preserving features. We call such discontinuities corresponding to a lateral neighborhood *lateral discontinuities*. Furthermore, proximity is a major grouping principle, which suggests another measure that reflects *local discontinuities* sensitive to the changes of local attributes among adjacent oscillators. Therefore, we adopt two different methods to measure fixed lateral discontinuities corresponding to potentially significant features and variable local discontinuities corresponding to local changes. Two discontinuity measures are jointly used in weight adaptation.

First, we use the variance of all the fixed weights associated with an oscillator to measure its lateral discontinuities in the following manner. For oscillator (i,j), we first calculate the mean of its fixed weights on $N(i,j;R)$, $\mu_{ij}(R)$, as

$$\mu_{ij}(R) = \frac{\sum_{(k,l)\in N(i,j;R)} T_{ij;kl}}{|N(i,j;R)|}. \tag{5}$$

Accordingly, the variance of its fixed weights $\sigma_{ij}^2(R)$ is calculated as

$$\begin{aligned}\sigma_{ij}^2(R) &= \frac{\sum_{(k,l)\in N(i,j;R)} (T_{ij;kl} - \mu_{ij}(R))^2}{|N(i,j;R)|} \\ &= \frac{\sum_{(k,l)\in N(i,j;R)} T_{ij;kl}^2}{|N(i,j;R)|} \\ &\quad - \left(\frac{\sum_{(k,l)\in N(i,j;R)} T_{ij;kl}}{|N(i,j;R)|}\right)^2.\end{aligned} \tag{6}$$

Moreover, we normalize the variance $\sigma_{ij}^2(R)$ as

$$\tilde{\sigma}_{ij}^2(R) = \frac{\sigma_{ij}^2(R) - \sigma_{\min}^2(R)}{\sigma_{\max}^2(R) - \sigma_{\min}^2(R)} \tag{7}$$

where $\sigma_{\max}^2(R)$ and $\sigma_{\min}^2(R)$ are the maximal and minimal variance across the entire image, respectively. Intuitively, $\tilde{\sigma}_{ij}^2(R)$ reflects the relative degree of the lateral discontinuities for oscillator (i,j). A large value of $\tilde{\sigma}_{ij}^2(R)$ indicates that the oscillator's lateral discontinuities likely correspond to significant features and vice versa. This suggests that the local attributes of an oscillator with a high lateral discontinuity should be preserved and those of an oscillator with a low lateral discontinuity should be adapted toward homogeneity.

To detect local discontinuities during weight adaptation, we define four detectors for oscillator (i,j) along four directions, as illustrated Fig. 3. These four directions are vertical (V), horizontal (H), diagonal (D), and counter-diagonal (C), respectively. Accordingly, four detectors are defined as

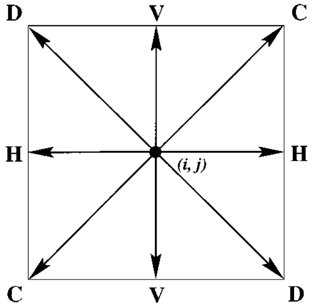


Fig. 3. Four directions used to detect local discontinuities at pixel (i,j). Coordinates of pixels increase from left to right in horizontal direction and from top to bottom in vertical direction.

TABLE I
RESPONSES OF FOUR DETECTORS TO A STEP EDGE ALONG ONE OF FOUR DIRECTIONS

orientation	$D_{H_{ij}}$	$D_{V_{ij}}$	$D_{C_{ij}}$	$D_{D_{ij}}$
H	h	0	h	h
V	0	h	h	h
C	h	h	h	0
D	h	h	0	h

$$D_{H_{ij}} = |W_{ij;i-1,j} - W_{ij;i+1,j}| \tag{8a}$$

$$D_{V_{ij}} = |W_{ij;i,j-1} - W_{ij;i,j+1}| \tag{8b}$$

$$D_{C_{ij}} = |W_{ij;i-1,j-1} - W_{ij;i+1,j+1}| \tag{8c}$$

$$D_{D_{ij}} = |W_{ij;i-1,j+1} - W_{ij;i+1,j-1}|. \tag{8d}$$

If there is a step edge through (i,j) in one of these four directions, the corresponding detector will respond strongly. Assume that h is the steepness of a step edge, Table I summarizes the responses to a possible edge through pixel (i,j) along one of those four directions. Based on the four detectors, we define a measure of local discontinuity as

$$D_{ij} = \frac{D_{H_{ij}} + D_{V_{ij}} + D_{C_{ij}} + D_{D_{ij}}}{4}. \tag{9}$$

D_{ij} is sensitive to local discontinuity regardless of local orientation.

Using both $\tilde{\sigma}_{ij}^2(R)$ and D_{ij}, we introduce $V_{ij}^{(t)}$ that integrates the local attributes of oscillator (i,j) to realize noise removal and feature preservation, as shown in (10) at the bottom of the next page, where s $(s > 0)$ is a scale used to determine to what extent local discontinuities should be preserved during weight adaptation and κ $(\kappa > 0)$ is used to determine to what extent features should be preserved in terms of lateral discontinuities. The function $\Phi(\nu, \theta)$ is defined as $\Phi(\nu, \theta) = 0$ if $\nu < \theta$ and $\Phi(\nu, \theta) = \nu$ if $\nu \geq \theta$. θ_σ $(0 \leq \theta_\sigma \leq 1)$ is a threshold used to alleviate the influence of noise in the estimation of lateral discontinuities (discussed later).

In general, the degree of lateral discontinuities in an image indicates the significance of the corresponding features. In (10), if all the oscillators in $N(i,j;1)$ have similar lateral discontinuities their local discontinuities should play a dominant role in the update of $V_{ij}^{(t)}$. In this case, the local discontinuities $D_{mn}^{(t)}$ of oscillator (m,n) in $N(i,j;1)$ determine the contribution of the dynamic weight $W_{ij;mn}^{(t)}$ in the update of $V_{ij}^{(t)}$. That is, a small value of $D_{mn}^{(t)}$ implies that $W_{ij;mn}^{(t)}$ have a large contribution and vice versa. Intuitively, the update of $V_{ij}^{(t)}$ makes oscillator (i,j) change its local attributes so that the dissimilarity between (i,j) and its adjacent oscillators is reduced in terms of the scale s. The reduction of dissimilarity results in noise removal along with feature preservation in terms of local discontinuities. When adjacent oscillators of (i,j) have different lateral discontinuities, the contribution from $W_{ij;mn}^{(t)}$ to $V_{ij}^{(t)}$ must depend upon both lateral and local discontinuities. In this case, $W_{ij;mn}^{(t)}$ makes a large contribution only if the overall discontinuities associated with oscillator (m,n) are relatively small. Intuitively, both lateral and local discontinuities cooperatively provide a robust way to fulfill feature preservation and noise removal for those oscillators associated with high lateral discontinuities, i.e., $\tilde{\sigma}_{mn}^2(R) \geq \theta_\sigma$. The local attributes of oscillator (i,j) tends to change toward reduction of the dissimilarity between (i,j) and the adjacent oscillators with relatively low overall discontinuities, while the dissimilarity between (i,j) and those with relatively high overall discontinuities tends to remain unchanged. The above analysis indicates that $V_{ij}^{(t)}$ plays a role of updating the local attributes of oscillator (i,j) for noise removal and feature preservation.

Based on (10), weight adaptation for $W_{ij;kl}$ is defined as

$$W_{ij;kl}^{(t+1)} = W_{ij;kl}^{(t)} + \Big[\exp\big(-\kappa\Phi\left(\tilde{\sigma}_{kl}^2(R), \theta_\sigma\right)\big) V_{kl}^{(t)} - \exp\big(-\kappa\Phi\left(\tilde{\sigma}_{ij}^2(R), \theta_\sigma\right)\big) V_{ij}^{(t)}\Big]. \tag{11}$$

In (11), $W_{ij;kl}^{(t+1)}$ is adapted based on the local attributes associated with (i,j) and (k,l) in terms of both lateral and local discontinuities. The lateral discontinuity further plays a role of gain control during weight adaptation. Thus, (11) incorporates both noise removal and feature preservation.

With respect to our weight adaptation scheme, an additional issue is worth mentioning. For an iterative system, the termination problem should be considered. For our adaptation scheme, the termination criteria can be defined in a similar way as in unsupervised learning methods, e.g., self-organizing feature maps [18]. In the next section, we shall empirically show that the weight adaptation scheme is insensitive to termination criteria and, therefore, an upper bound on iteration numbers can be readily given as a termination condition.

B. Extension of LEGION Algorithm

As described before, the size of a LEGION network for image segmentation is the same as the given image; each oscillator of the network corresponds to a pixel of the image. Due to a large number of pixels in a real image, numerical integration of hundreds of thousands of differential equations is prohibitively expensive. To reduce numerical computation on a serial computer, a simplified LEGION algorithm was recently proposed by Wang and Terman [39]. The algorithm preserves essential dynamic properties of LEGION, such as two time scales and the properties of synchronization and desynchronization. By a similar argument, we extend this algorithm by incorporating weight adaptation and a new grouping rule to be discussed below. In addition, a new leader generation method is proposed.

When a LEGION network is applied to gray-level image segmentation, every oscillator is assumed to be stimulated [39]. It implies that every oscillator can enter the active phase once the coupling term, S_{ij}^a, in (2) is strong enough. As mentioned before, coupling is critical for determining whether an oscillator can synchronize with other oscillators. Previously, two grouping rules called summation and maximization were proposed [39]. The former rule summates the dynamic weights of those neighboring oscillators that are active, while the latter takes the maximal one. It is observed that both of them suffer from either losing important boundary detail or yielding small noisy regions. To overcome these shortcomings, we propose an alternative grouping rule using a logarithmic operation

$$S_{ij}^a = \frac{W_{\max} \sum_{(m,n)\in N(i,j;1)} H(x_{mn})/(1+|W_{ij;mn}|)}{\log\left(\sum_{(m,n)\in N(i,j;1)} H(x_{mn})+1\right)} \tag{12}$$

where x_{mn} is the activity of oscillator (m,n). $W_{\max} = I_{\max} - I_{\min}$, and $I_{\max}$ and $I_{\min}$ are the maximal and minimal intensity across the entire image. Note that the reciprocal of $W_{ij;mn}$ is used in (12) as $W_{ij;mn}$ encodes dissimilarity.

In order to lead an oscillator group, a leader is always required to be stimulated in LEGION. Here, by extending the concept of lateral potentials (see (1a)), we observe that an oscillator corresponding to the pixel located near the center of a homogeneous region tends to have a high potential. Based on the observation, we identify such oscillators directly in the following way. For oscillator (i,j), we calculate $\mu_{ij}(1)$, $\sigma_{ij}^2(1)$, $\mu_{ij}(R_p)$, and $\sigma_{ij}^2(R_p)$ using (5) and (6). Oscillator (i,j) is a leader if and only if $|\mu_{ij}(R_p) - \mu_{ij}(1)| \leq T_\mu$ and $|\sigma_{ij}^2(R_p) - \sigma_{ij}^2(1)| \leq T_\sigma$. Here, T_μ and T_σ are two thresholds to reflect homogeneity and

$$V_{ij}^{(t)} = \frac{\sum_{(m,n)\in N(i,j;1)} \exp\Big[-\Big(\kappa\Phi\ \left(\tilde{\sigma}_{mn}^2(R), \theta_\sigma\right) + D_{mn}^{(t)}\Big/ s\Big)\Big]\, W_{ij,mn}^{(t)}}{\sum_{(m,n)\in N(i,j;1)} \exp\Big[-\Big(\kappa\Phi\ \left(\tilde{\sigma}_{mn}^2(R), \theta_\sigma\right) + D_{mn}^{(t)}\Big/ s\Big)\Big]} \tag{10}$$

R_p ($R_p > 1$) is a parameter to reflect whether an oscillator is within a homogeneous region. Intuitively, a large value of R_p results in leaders that are generated from large homogeneous regions, while a more rigorous test on the mean and the variance results in leaders that are from more homogeneous regions. Note that R_p may be different from R used to determine the size of the fixed neighborhood.

By incorporating all the proposed mechanisms described above, an extended LEGION algorithm is generated. In summary, the following aspects distinguish our extended LEGION algorithm from the original one. First, a new method is used to generate leaders. Second, weight adaptation is performed for a given number of iterations T_0 in the initialization step. Third, the logarithmic grouping rule is used to replace the summation and maximization rules. All the other parts in the original algorithm still remain the same (see [39] for details). We point out that, like the original LEGION algorithm, our extended algorithm is consistent with LEGION dynamics established by Terman and Wang [33].

C. Feature-Preserving Smoothing Algorithm

Our weight adaptation scheme can be converted into a form of nonlinear smoothing, which we call feature-preserving smoothing (FPS). For consistency, the previous notations are still utilized to describe the algorithm. First of all, we calculate the variance $\sigma^2_{ij}(R)$ on the lateral neighborhood of pixel (i, j)

$$\sigma^2_{ij}(R) = \frac{\sum_{(m,n)\in N(i,j;R)\cup\{(i,j)\}} I^2_{mn}}{|N(i,j;R)|+1} - \left(\frac{\sum_{(m,n)\in N(i,j;R)\cup\{(i,j)\}} I_{mn}}{|N(i,j;R)|+1}\right)^2. \qquad (13)$$

The normalized variance $\tilde{\sigma}^2_{ij}(R)$ is calculated accordingly using (7). In addition, the measure of local discontinuities in (9) is still used, but those detectors along four directions as illustrated in Fig. 3 are rewritten in terms of pixel values as

$$D_{H_{ij}} = |I_{i-1,j} - I_{i+1,j}| \qquad (14a)$$

$$D_{V_{ij}} = |I_{i,j-1} - I_{i,j+1}| \qquad (14b)$$

$$D_{C_{ij}} = |I_{i-1,j-1} - I_{i+1,j+1}| \qquad (14c)$$

$$D_{D_{ij}} = |I_{i-1,j+1} - I_{i+1,j-1}|. \qquad (14d)$$

According to (10) and (11), the iterative scheme is (15), as shown at the bottom of the page. In (15), $I^{(0)}_{ij}$ denotes the original intensity of pixel (i, j). Note that the FPS algorithm is equivalent to the weight adaptation scheme if we define $W^{(t)}_{ij;kl} = I^{(t)}_{kl} - I^{(t)}_{ij}$, where $(k, l) \in N(i, j; 1)$.

There is often an issue of bounding in an adaptive scheme, which concerns possible unbounded growth or boundness of the final result of the adaptive scheme. The essential criterion defining boundedness is that this numerical process must restrict the amplification of all components from the initial conditions. Here, we show the boundedness of our proposed smoothing method. Let $\eta_{ij} = \exp(-\kappa\Phi(\tilde{\sigma}^2_{ij}(R), \theta_\sigma))$ and $\gamma^{(t)}_{ij} = \exp(-D^{(t)}_{ij}/s)$ to simplify the presentation. We rewrite the above iterative scheme as

$$\begin{aligned} I^{(t+1)}_{ij} &= I^{(t)}_{ij} + \eta_{ij} \frac{\sum_{(k,l)\in N(i,j;1)} \eta_{kl}\gamma^{(t)}_{kl}\left(I^{(t)}_{kl} - I^{(t)}_{ij}\right)}{\sum_{(k,l)\in N(i,j;1)} \eta_{kl}\gamma^{(t)}_{kl}} \\ &= (1-\eta_{ij})I^{(t)}_{ij} \\ &\quad + \eta_{ij} \sum_{(k,l)\in N(i,j;1)} \frac{\eta_{kl}\gamma^{(t)}_{kl}}{\sum_{(k,l)\in N(i,j;1)} \eta_{kl}\gamma^{(t)}_{kl}} I^{(t)}_{kl}. \end{aligned}$$

Apparently, $I_{\min} \le I^{(t)}_{ij} \le I_{\max}$ for any pixel (i, j) in the image. In other words, pixel values are bounded during smoothing.

According to the definition of dynamic coupling weight, $W^{(t)}_{ij;kl}$ can be represented as

$$W^{(t)}_{ij;kl} = I^{(t)}_{kl} - I^{(t)}_{ij}$$

Utilizing the above result, we can obtain $W_{\min} \le W^{(t)}_{ij;kl} \le W_{\max}$, where $W_{\min} = I_{\min} - I_{\max}$ and $W_{\max} = I_{\max} - I_{\min}$. As a result, any updated value of dynamic coupling weight is bounded within $[W_{\min}, W_{\max}]$.

IV. Simulation Results

In this section, we report segmentation results of applying our extended LEGION algorithm to noisy synthetic images, aerial images, and magnetic resonance imaging (MRI) images. We give all the parameter values in Table II. Note that a figure appearing in this section corresponds to more than one simulation. Therefore, an item with a single value in Table II means that the same parameter value is used in all the simulations illustrated by the figure, while multiple parameter values in an item indicate that the parameter changes in different simulations. For discussions on parameter choices see Section IV-D below.

$$\begin{aligned} I^{(t+1)}_{ij} &= I^{(t)}_{ij} + \exp\!\left(-\kappa\Phi\left(\tilde{\sigma}^2_{ij}(R), \theta_\sigma\right)\right) \\ &\quad \times \frac{\sum_{(m,n)\in N(i,j;1)} \exp\!\left[-\left(\kappa\Phi\left(\tilde{\sigma}^2_{mn}(R), \theta_\sigma\right) + D^{(t)}_{mn}\big/ s\right)\right]\left(I^{(t)}_{mn} - I^{(t)}_{ij}\right)}{\sum_{(m,n)\in N(i,j;1)} \exp\!\left[-\left(\kappa\Phi\left(\tilde{\sigma}^2_{mn}(R), \theta_\sigma\right) + D^{(t)}_{mn}\big/ s\right)\right]}. \end{aligned} \qquad (15)$$

TABLE II
PARAMETER VALUES USED IN SIMULATIONS REPORTED SECTION IV

Parameter	Figure 4	Figure 5	Figure 6	Figure 7	Figure 8
R_p	9	9	12	4	2
T_μ	2	2	0.1	0.2	0.1
T_σ	10	10	1	2	2
R	9	9	2	2	1
s	12	12	8	8	$(6, 5, 7)$
θ_σ	0.7	0.7	$(0.01, 0.03)$	$(0.02, 0.03)$	$(0.01, 0.009, 0.02)$
κ	4	10	40	$(60, 40)$	$(80, 100)$
T_0	$(40, 1000)$	$(650, 2000)$	20	20	40
W_z	$(90, 65)$	$(90, 65)$	$(70, 80)$	$(95, 70)$	$(45, 65, 42)$

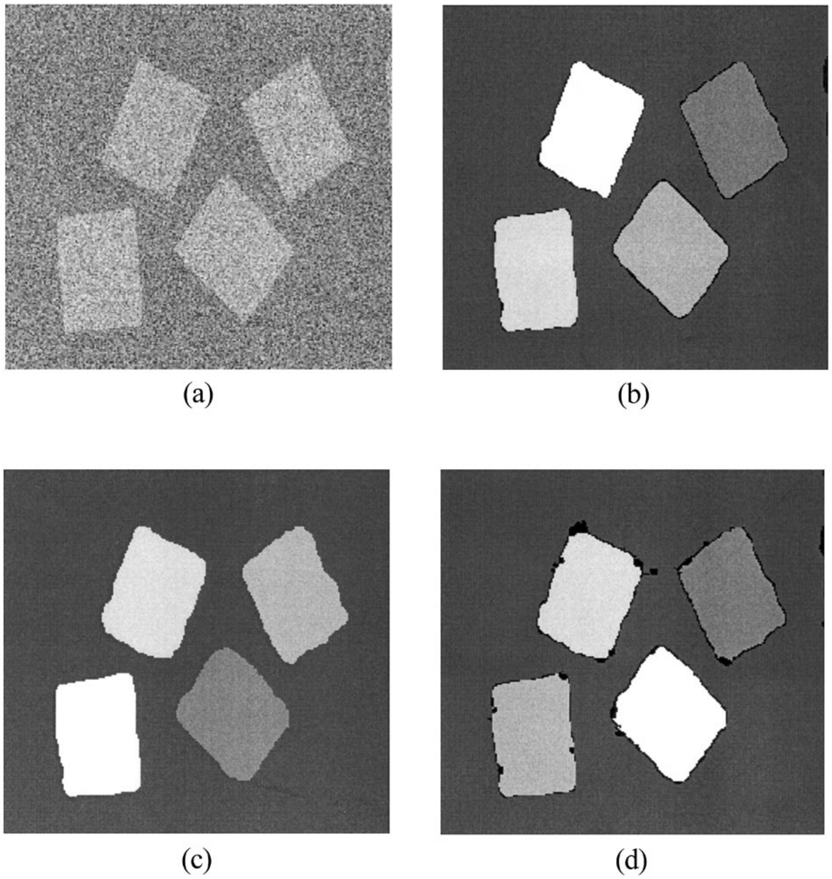


Fig. 4. Segmentation results for a synthetic image with $\kappa = 4.0$. (a) A noisy synthetic image consisting of 230 × 240 pixels. (b) Segmentation result using the logarithmic operation with $T_0 = 40$. (c) Segmentation result using the logarithmic operation with $T_0 = 1000$. (d) Segmentation result using the maximization operation with $T_0 = 40$ and $W_z = 65.0$.

A. Synthetic Images

The use of synthetic images allows us to have a thorough evaluation of effectiveness of our extended LEGION algorithm.

The first synthetic image used to test our algorithm is shown in Fig. 4(a), which was used by Sarkar and Boyer [29]. As stated before, the original LEGION algorithm fails to segment the image into four parallelograms and an image background. To simplify the display of segmentation results, we use the so-called *gray map* convention [39], where each gray level indicates a distinct segment. In the gray-map convention, all the excitable oscillators constitute the *background* corresponding to those parts with high intensity variations. Fig. 4(b) is a resulting gray map of this simulation, where the background is indicated by the black areas. Our algorithm segments the image into five segments corresponding to four parallelograms and the image background. To demonstrate immunity to the termination problem, we use a different termination condition

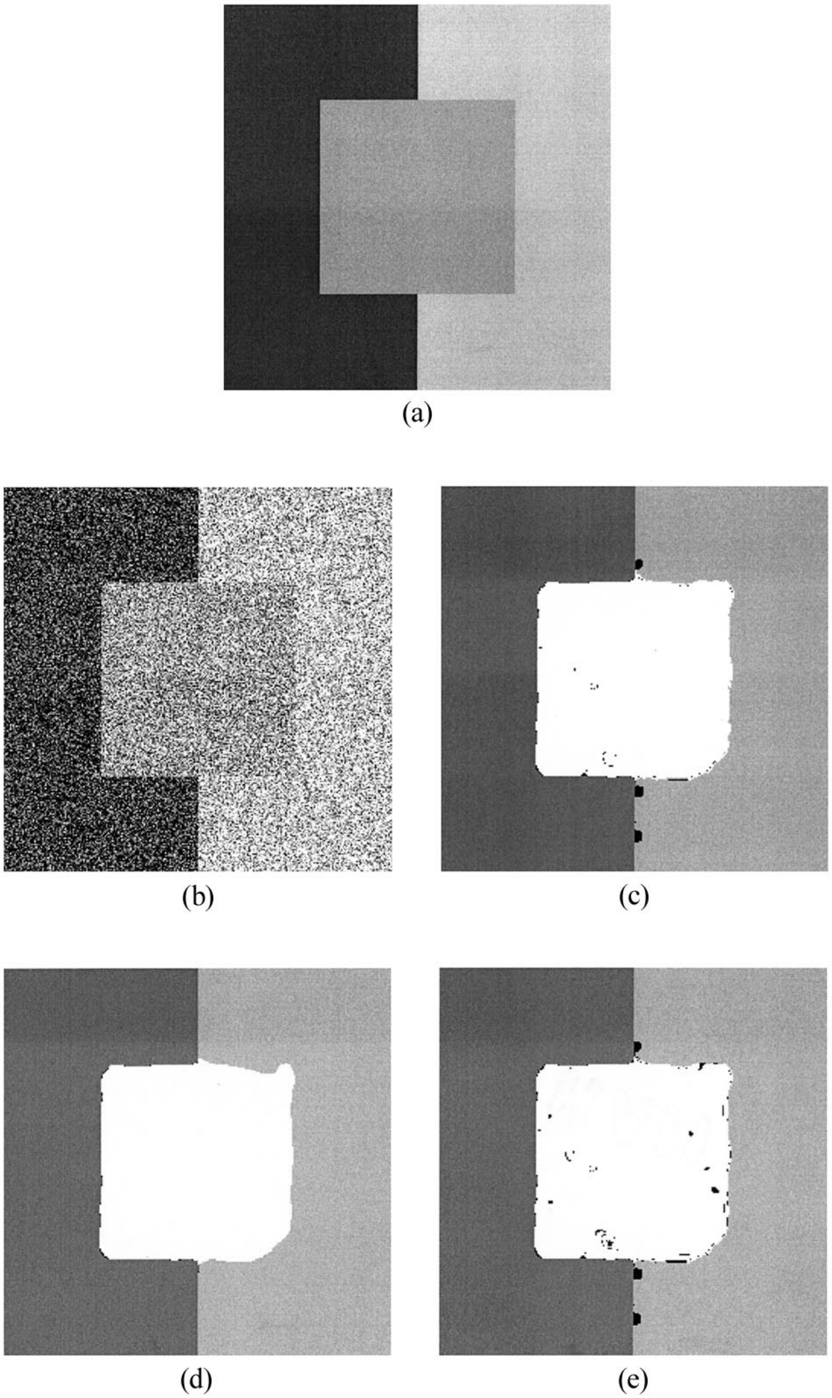


Fig. 5. Segmentation results for a noisy synthetic image with $\kappa = 10.0$. (a) A noise-free synthetic image consisting of 256 × 256 pixels, where the intensities corresponding to three regions are 42, 160, and 198, respectively. (b) A noisy version of the image in (a). (c) Segmentation result using the logarithmic operation with $T_0 = 650$. (d) Segmentation result using the logarithmic operation with $T_0 = 2000$. (e) Segmentation result using the maximization operation with $T_0 = 650$ and $W_z = 65.0$.

in weight adaptation and Fig. 4(c) shows the corresponding segmentation result. A very similar segmentation result is achieved. To evaluate the logarithmic grouping rule, we also conduct a simulation by using the previous maximization rule [39] in our algorithm. As illustrated in Fig. 4(d), the resulting segmentation contains ragged boundaries and small holes in the image background though four parallelograms are separated correctly with the help of weight adaptation. This result is not as good in comparison with those shown in Fig. 4(b) and (c).

Fig. 5(a) shows a noise-free synthetic image, where a square is in the middle of two adjacent rectangles and the boundary sharpness along the square is different. Fig. 5(b) shows a noisy version of Fig. 5(a) by adding Gaussian noise with zero mean and $\sigma = 64$. Note different sharpness of boundaries, e.g., for four vertical lines, sharpness of three left boundaries is better than the right one. The noisy image shown in Fig. 5(b) is challenging to an edge-based segmentation algorithm since weak edges may either be detected along with many noisy details in a small scale or hardly be detected at all in a large scale. For the same purpose as described in the last paragraph, Fig. 5(c) and (d) illustrate two segmentation results using the logarithmic grouping rule with two different termination conditions. Our algorithm yields three segments corresponding to the three rectangular regions and a background indicated by the small black areas. Once again, our simulations show that the segmentation results are insensitive to the termination condition. For comparison, Fig. 5(e) shows a segmentation result using the maximization grouping rule. Again, the quality of segmentation produced by the logarithmic grouping rule is superior to the one produced by the maximization grouping rule.

These simulation results show that our algorithm can segregate objects that are highly corrupted by noise due to weight adaptation. Our weight adaptation method does not suffer from the termination problem; that is, very similar segmentation results are achieved in a wide range of update iterations. Moreover, the logarithmic grouping rule performs better than the maximization rule. In the sequel, we shall report only the results produced by the logarithmic grouping rule.

B. Aerial Images

The segmentation of aerial images is a difficult task because the pixels within a semantically meaningful entity are generally not homogeneous while pixels belonging to different entities may have similar attributes. In addition, aerial images taken from satellites are often corrupted by noise from different sources. Therefore, most of traditional approaches such as various thresholding methods and classification techniques have limited success. We have applied our extended LEGION algorithm to satellite images. The images used in our simulations are provided by the U.S. Geological Survey. These high-resolution satellite images are processed using a nonlinear transformation to compensate variations in actual pixel sizes on the ground due to perspective projection. In our simulations, two types of experiments are conducted: to extract hydrographic objects and to segment images entirely.

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. It is critical for creating a geographic information system. Hydrographic objects tend to be more homogeneous in comparison with other kinds of objects. When LEGION is applied, we can utilize this property to generate leaders so that only oscillators belonging to hydrographic objects are identified as leaders and other objects are naturally put into a background. In simulations, we use a set of special parameters, R_p, T_μ, and T_σ, to generate leaders and fix most of other parameters, i.e., R, s, κ, and T_0. With this set of parameters leaders can be only those oscillators within very homogeneous regions so as to reflect hydrographic objects. Other parameters as listed in Table II are determined based on one image and then used in all the simulations on extracting hydrographic objects including those not reported here.

Fig. 6(a) shows a satellite image that contains a river. The docks cause the river boundaries to be irregular. Fig. 6(b) shows the extraction result. To facilitate comparisons, we display the river by marking it as white and superimposing it on the original image. To demonstrate the effectiveness of our algorithm, Fig. 6(c) provides the corresponding part of the USGS 1:24 000 topographic map. It is evident that our algorithm extracts the river precisely even along the noisy lower bank. Moreover, important details are preserved, such as the small island near the uppermost river branch. Fig. 6(d) shows a more difficult image in which there are several hydrographic objects. Almost all the boundaries are noisy. Fig. 6(e) shows the extraction result, and Fig. 6(f) gives the corresponding part of the topographic map. Again, our algorithm extracts those major hydrographic objects well, even along some narrow river branches. A careful comparison between the extracted regions and the maps indicate that the former portray the images even a little better because stationary maps do not reflect well the changing nature of geography.

Next, we show the entire segmentation results on two other aerial images. Fig. 7(a) shows an aerial image severely corrupted by noise. Again, we use a gray map to display the segmentation results. As illustrated in Fig. 7(b), the image is separated into seven regions and a background indicated by the black scattered areas. Our segmentation results show that the salient regions, e.g., the water body and the island, are segmented with high accuracy. Fig. 7(c) shows another satellite image, where a river is separated by a bridge. The segmentation result is shown in Fig. 7(d) and the image is segmented to 69 regions and a background indicated by the black areas. Fig. 7(d) shows that most of homogeneous regions are segmented and the two parts of the river are both segmented with good accuracy (see the three islands). Moreover, many regions with high variations are put into the background since no leader can be generated from them.

C. MRI Images

With the wide use of MRI techniques in medicine, automatic MRI image processing is increasingly demanded in practice. We use MRI images of human heads to further test our extended LEGION algorithm for segmentation. Fig. 8 shows six MRI images with different views and the corresponding segmentation results.

Fig. 8(a) shows a midsagittal MRI image. The segmentation result is shown in Fig. 8(b) by a gray map. The image is segmented to 43 regions plus a background indicated by the black regions. Many salient regions are separated such as the cerebral cortex, the cerebellum, the corpus callosum and fornix area, parts of the extracranial tissue and the bone marrow, and several other anatomical structures. The next image is another sagittal section, as illustrated in Fig. 8(c). This image contains the cortex, the cerebellum, the lateral ventricle (the black hole embedded in the cortex), the eye, the sinus (the black hole below the eye), the extracranial soft tissue, and the bone marrow. Fig. 8(d) shows the segmentation result with 46 segments plus a background. It is evident that most of aforementioned salient regions are separated correctly. In particular, the cerebellum is segmented with high accuracy despite low contrast with its adjacent cortex. However, a small part of extracranial tissue is put into the background. It is worth pointing out that these two MRI images are used to test the original LEGION algorithm [39], and our algorithm yields considerably better results.

Fig. 8(e) is an image of a coronal section, and Fig. 8(f) shows its segmentation. The image is segmented to 25 regions plus a background. Salient regions segmented are the cortex, the cere-

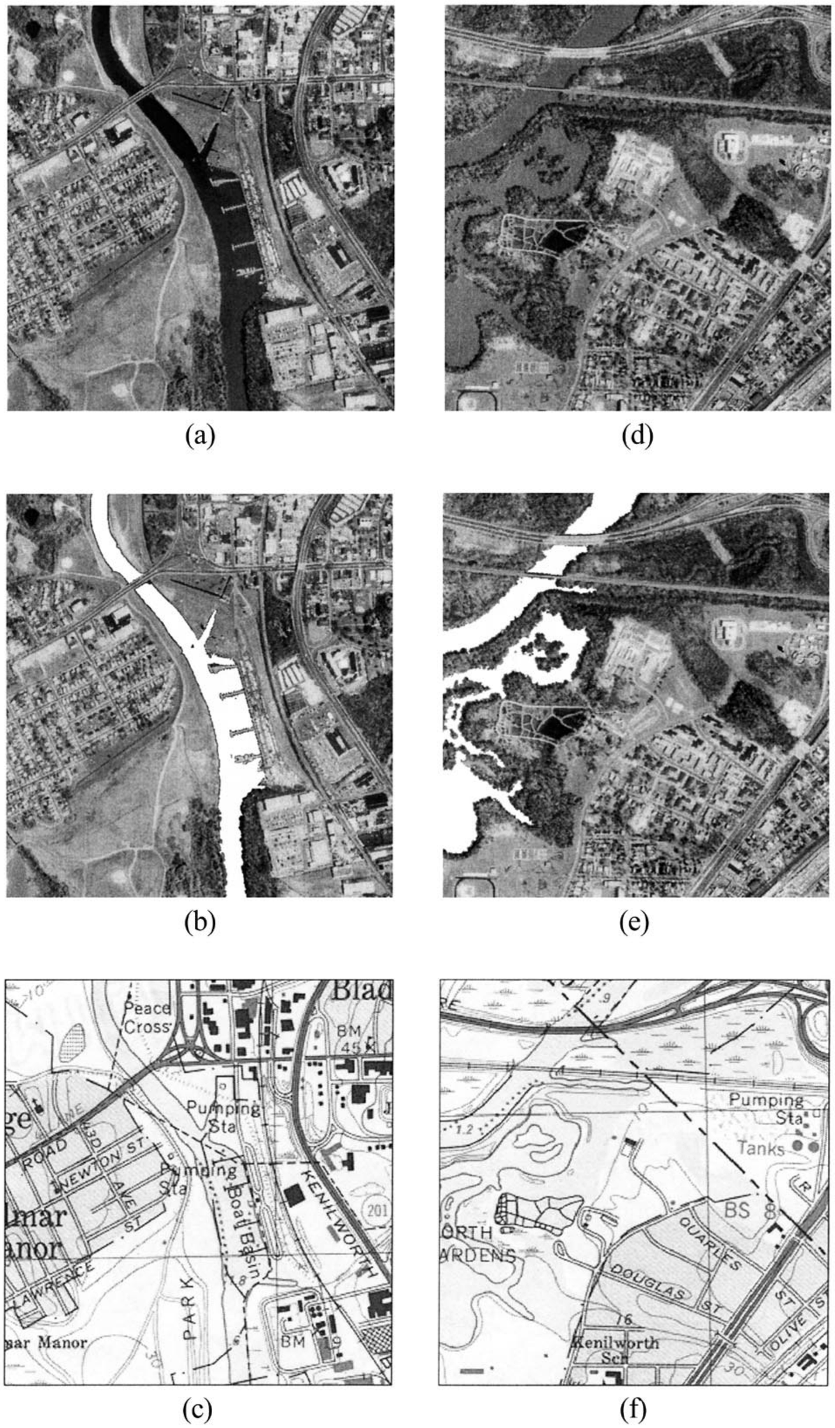


Fig. 6. Results of extracting hydrographic objects from two satellite images. (a) Original image containing a river (670 × 606 pixels). (b) Extraction result for the image in Fig. 6(a) with $\theta_\sigma = 0.01$ and $W_z = 70.0$. (c) The topographic map corresponding to the image in Fig. 6(a). (d) Original image containing several hydrographic objects (640 × 606 pixels). (e) Extraction result for the image in Fig. 6(d) with $\theta_\sigma = 0.03$ and $W_z = 80.0$. (f) The topographic map corresponding to the image in Fig. 6(d).

bellum, and the lateral ventricles. Fig. 8(g) is another coronal section, and the image is segmented into 30 regions plus a background as shown in Fig. 8(h). Similarly, those significant regions are also segregated. However, much extracranial tissue is suppressed into the background since no leader can be generated. Next, we use a horizontal section shown in Fig. 8(i). Fig. 8(j) shows the result, and the image is segmented into 55 regions plus a background. Salient regions segmented include two hemispheres, two eyeballs, and the third ventricle at the center. Fig. 8(k) is the same image of Fig. 8(i) with the resolution reduced by half. As shown in Fig. 8(l), the image is separated into 41 regions plus a background. Although two eyeballs and the third ventricle are well segmented, the brain is grouped together.

We have applied our extended LEGION algorithm to many other images and our algorithm yields segmentation results of similar quality as demonstrated above. In particular, our simu-

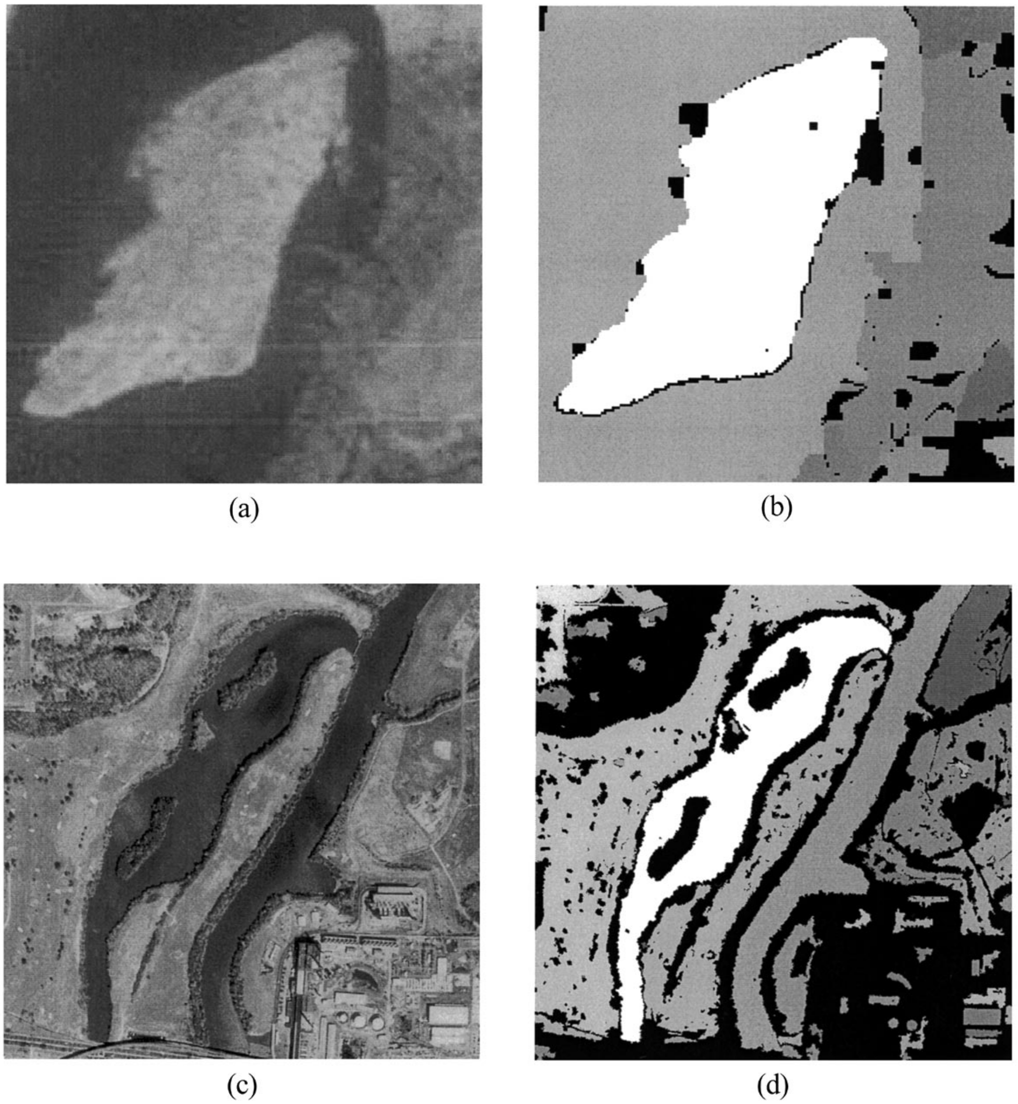


Fig. 7. Segmentation results for two satellite images. (a) Original image consisting of 181 × 181 pixels. (b) Segmentation result of the image in Fig. 7(a) with $\kappa = 60.0$ and $W_z = 95.0$. (c) Original image consisting of 640×640 pixels. (d) Segmentation result of the image in Fig. 7(c) with $\kappa = 40.0$ and $W_z = 70.0$.

lations show that weight adaptation is relatively immune to the termination problem. In our simulations, the same termination condition is always used for a type of images. Finally, computing times taken by our algorithm are similar to those taken by the original LEGION algorithm [39] for the same images, which indicates that weight adaptation does not involve expensive computation.

D. Parameter Choice

Based on simulation results, we discuss parameters in our extended LEGION algorithm. In original LEGION [33], [39], there are several parameters that determine its dynamics. Due to approximation, most of them, i.e., α, β, ρ, ϵ, and θ_z disappear. However, there are still parameters that need to be adjusted in our extended LEGION algorithm. Most of parameters may be fixed for a natural image category, e.g., satellite imagery. As shown in Table II, these include the parameter R to determine the size of the lateral neighborhood and those for generating leaders R_p, T_μ, and T_σ. Since images belonging to a natural category are of similar statistical properties, we can determine these parameters by analyzing only one image for a class of images in our simulations. T_0 is a termination parameter used in weight adaptation. As shown in simulations reported in this paper including the next section, the similar effects in both smoothing and segmentation are achieved for a large interval of T_0. As for W_z for segmentation, its role and effect were discussed at length by Wang and Terman [39].

In the sequel, we focus on the other three parameters used in weight adaptation. The parameter s determines the magnitude of the edges to be preserved during the adaptation process in terms of local discontinuities. The parameter can be viewed equivalently as the scale used in a diffusion scheme if the termination condition or the number of iterations is fixed [28]. In a diffusion scheme, a large scale causes all the discontinuities to disappear, while a small scale causes all the discontinuities to be preserved. The parameter s plays a similar role in our method though it is not the only way to determine whether discontinuities will be preserved due to two types of discontinuity used in our algorithm. Our simulations show that a proper choice of

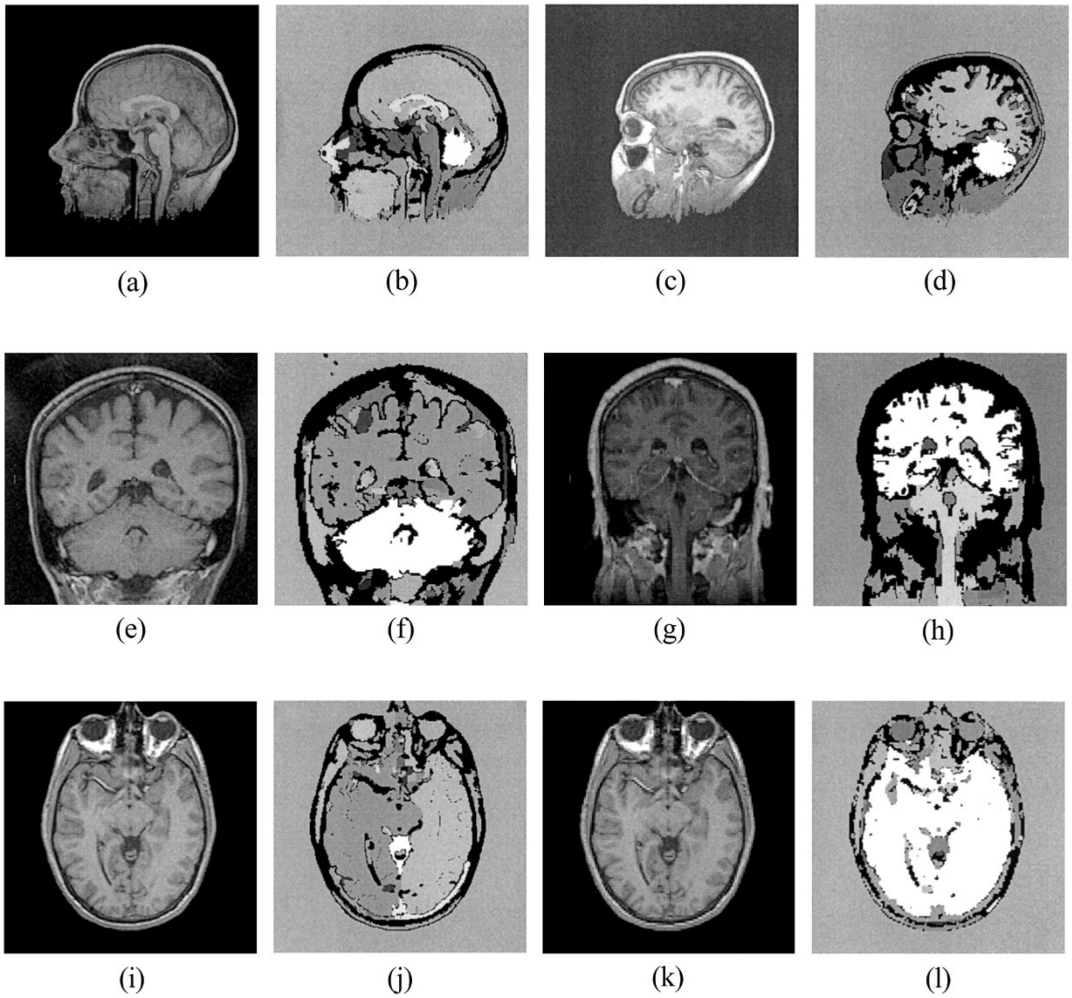


Fig. 8. Segmentation results for MRI images. (a) Original image consisting of 257 × 257 pixels. (b) Segmentation result of the image in (a) with $\kappa = 80.0$, $\theta_\sigma = 0.01$, $s = 6.0$, and $W_z = 45.0$. (c) Original image consisting of 257 × 257 pixels. (d) Segmentation result of the image in Fig. 8(c) with the same parameters as in (b). (e) Original image consisting of 175 ×175 pixels. (f) Segmentation result of the image in Fig. 8(e) with $\kappa = 100.0$, $\theta_\sigma = 0.009$, $s = 5.0$, and $W_z = 65.0$. (g) Original image consisting of 256 × 256 pixels. (h) Segmentation result of the image in Fig. 8(g) with the same parameters as in (f). (i) Original image consisting of 256 × 256 pixels. (j) Segmentation result of the image in Fig. 8(i) with $\kappa = 80.0$, $\theta_\sigma = 0.02$, $s = 7.0$, and $W_z = 42.0$. (k) Original image consisting of 256 × 256 pixels. (j) Segmentation result of the image in Fig. 8(i) with the same parameters as in (k).

s can often be used in a category of images, e.g., satellite imagery, as shown in Table II. For preserving tiny but nontrivial structures, e.g., MRI imagery, a slight adjustment of s may be needed within a small interval, as also shown in Table II. The parameter κ plays two roles; one is to determine the speed of weight adaptation, and the other is to determine what discontinuities should be preserved. In general, a small value of κ results in fast adaptation and reduces discontinuities, while a large value results in slow adaptation and preserves discontinuities. Its choice depends upon domain tasks; a large value should be chosen if detailed structures should be preserved (e.g., for MRI images), while a small value should be chosen if only major structures are needed (e.g., for noisy synthetic images). The last parameter θ_σ is used for reducing noise effects. Suppose that noise is independent of original image data, the variance calculated in (6) is the summation of the variance of noise-free image data and the variance of noise, regardless of noise type. If we know its type or distribution, the variance of noise may be estimated more accurately. Unfortunately, real images are generally corrupted by noise from various unknown sources, and it is impossible to calculate the exact variance of noise. In our method, we use the parameter θ_σ to reduce the effects of noise instead of directly evaluating the variance of noise. A small θ_σ tends to preserve detailed structure and a large value tends to extract only major regions. These three parameters are tuned based on statistical properties of an image, e.g., histogram of intensities. As shown in Table II, the change of these three parameters for a category of images is within a small interval.

V. COMPARISONS

In this section, we present comparative results in terms of segmentation and smoothing. Multiscale-based techniques have been shown to be effective for image processing. First, we use a recent multiscale adaptive segmentation method [4] for comparison in image segmentation. As mentioned before, the proposed weight adaptation scheme can be connected to a nonlinear

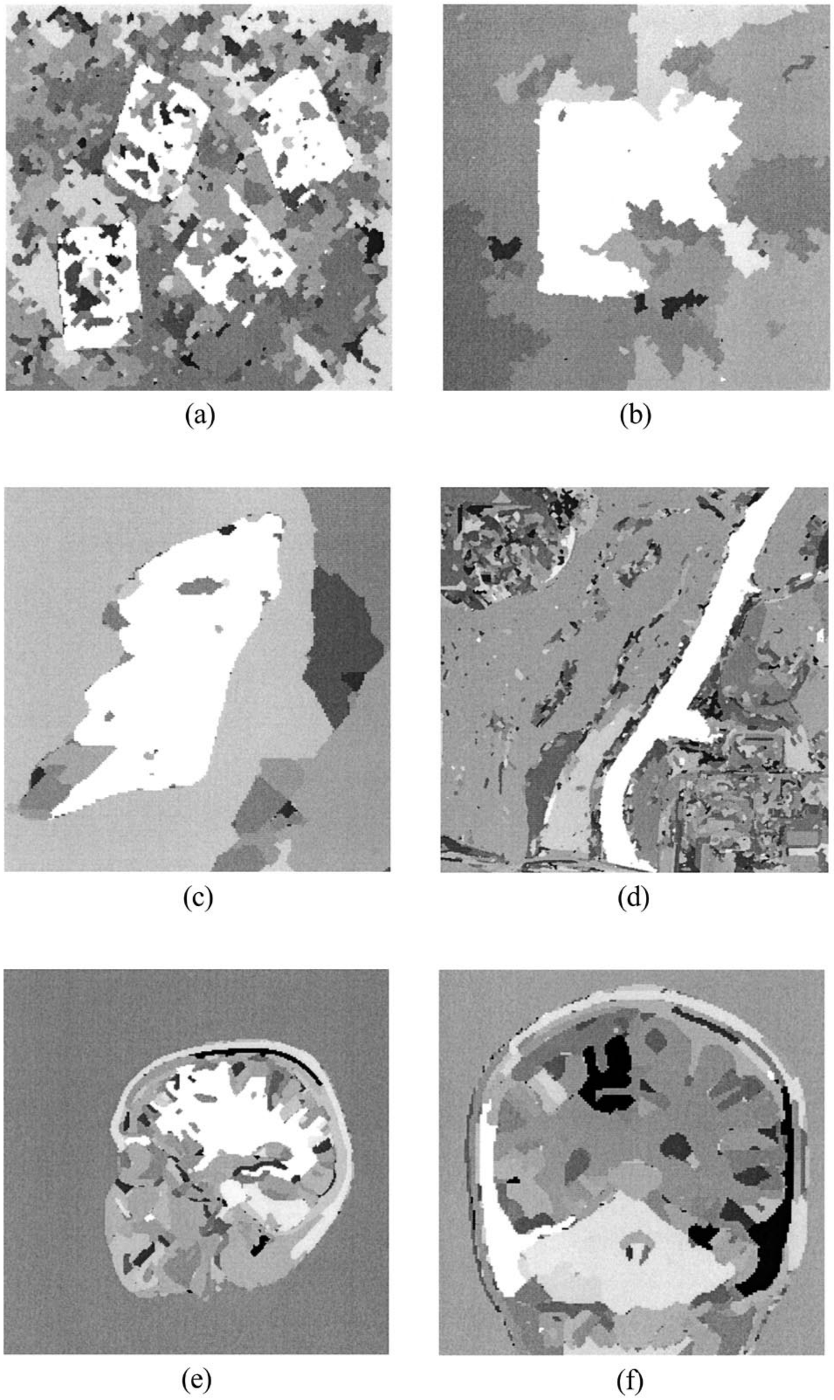


Fig. 9. Segmentation results by the multiscale adaptive segmentation method of Caelli and Bischof. (a) Segmentation result of the image in Fig. 4(a). (b) Segmentation result of the image in Fig. 5(b). (c) Segmentation result of the image in Fig. 7(a). (d) Segmentation result of the image in Fig. 7(c). (e) Segmentation result of the image in Fig. 8(c). (f) Segmentation result of the image in Fig. 8(e).

smoothing method. Thus, we also compare results with several recent nonlinear smoothing methods.

A. Comparison in Segmentation

A multiscale adaptive method was recently proposed by Caelli and Bischof for image segmentation [4]. The basic idea underlying the method is to use region characteristics to decide whether a given region should be segmented at a finer scale. The algorithm uses both edge and region attributes at multiple scales to adaptively and automatically choose the best scale for segmenting various parts. The algorithm results in a recursive procedure of integrating edge detection and region growing from coarse to fine scale to achieve optimal segmentation and has been successfully applied in various kinds of real images [4]. Features of this multiscale algorithm include that no parameter needs to be tuned by users and the algorithm segments an image entirely.

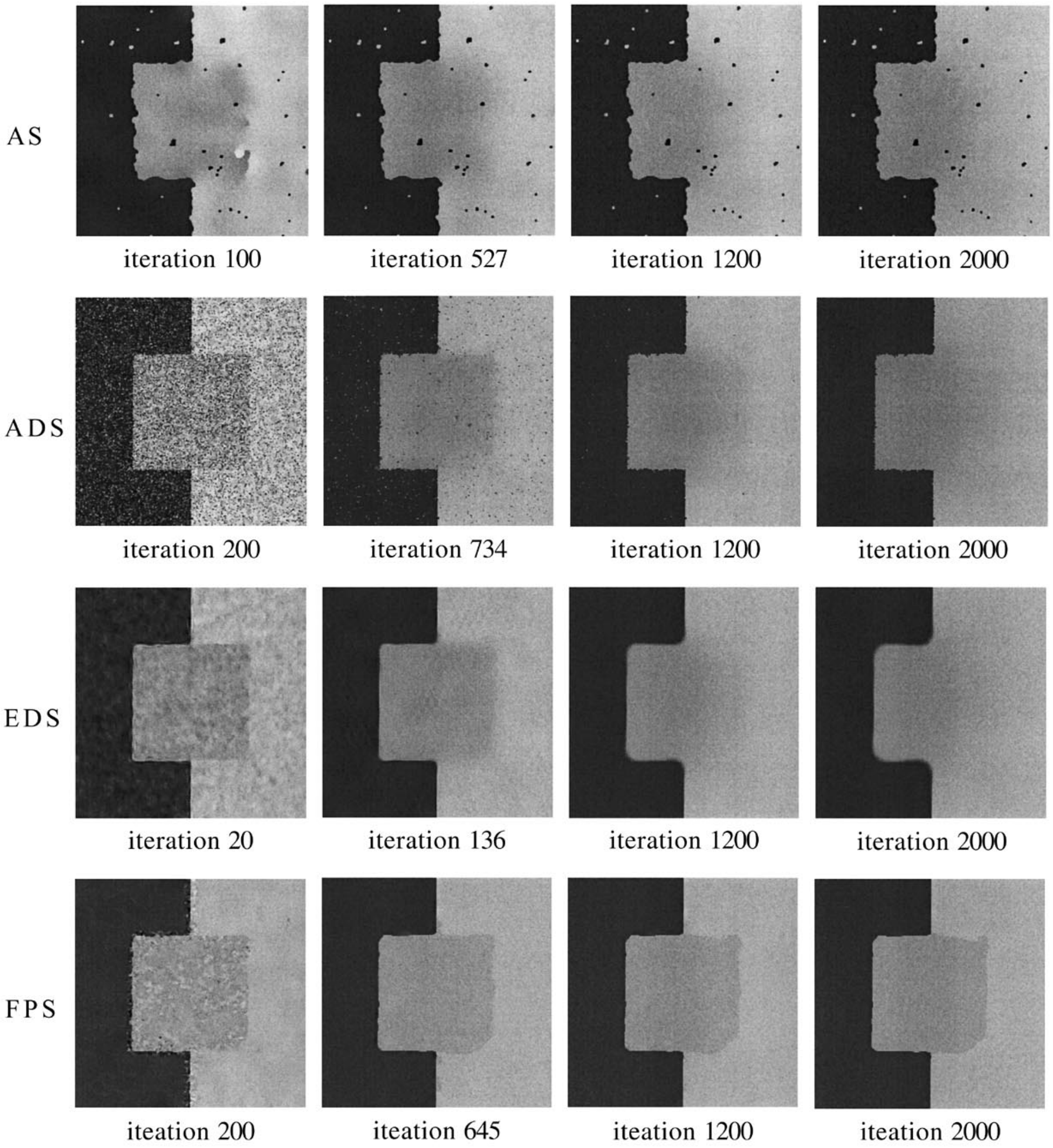


Fig. 10. Smoothing results for the synthetic image in Fig. 5(b) by different smoothing algorithms, where each row shows a set of snapshots for a smoothing algorithm. For the AS algorithm, $k = 13.0$; for the ADS algorithm, $K = 4.0$; for the EDS algorithm, $\lambda = 3.5$ and $\sigma = 3.0$; for the FPS algorithm, $R = 9$, $s = 12.0$, $\kappa = 10.0$, and $\theta_\sigma = 0.7$.

As suggested in [4], we use three scales in their algorithm. The same images for testing our algorithm are used for comparison. These images belong to three different types: two synthetic images for comparing the capability of noise removal, two satellite images with complicated boundaries for comparison in performance in real images and two MRI images with low contrasts for comparison in feature preservation. Fig. 9 shows those gray maps of their segmentation results. Fig. 9(a) and (b) show the results on the synthetic images in Fig. 4(a) and 5(b), respectively. The results are rather poor, and the two synthetic images are segmented to 98 and 51 fragments, respectively. The reason is probably that edges cannot be reliably detected in very noisy images, despite multiple scales used. Fig. 9(c) and (d) show the segmentation results of the aerial images in Fig. 7(a) and (c). In Fig. 9(c), the image is segmented to 48 regions. The river and the island are segmented and the boundary of the island is reasonably accurate. However, the boundary between the river and its bank cannot be correctly segmented due to noise. In addition, the island is segmented to a large region as well as many fragments. In Fig. 9(d), the image is segmented to 198 regions. The right part of the river is accurately segmented, but the left part cannot be segmented. Fig. 9(e) and (f) show the segmentation results of the MRI images in Fig. 8(c) and (e), where the two images are segmented to 97 and 98 regions, respectively. In Fig. 9(e), most of salient regions are partially segmented such as the cerebral cortex, the cerebellum, the lateral ventricle, and the extracranial tissue. In particular, the extracranial tissue is completely extracted. However, the cerebellum is undesirably segmented to several regions. In Fig. 9(f), significant segments include the cerebral cortex, the cerebellum, and parts of the extracranial tissue. Although much of the cerebellum is extracted, it is shrunk, and so is the segmented cortex.

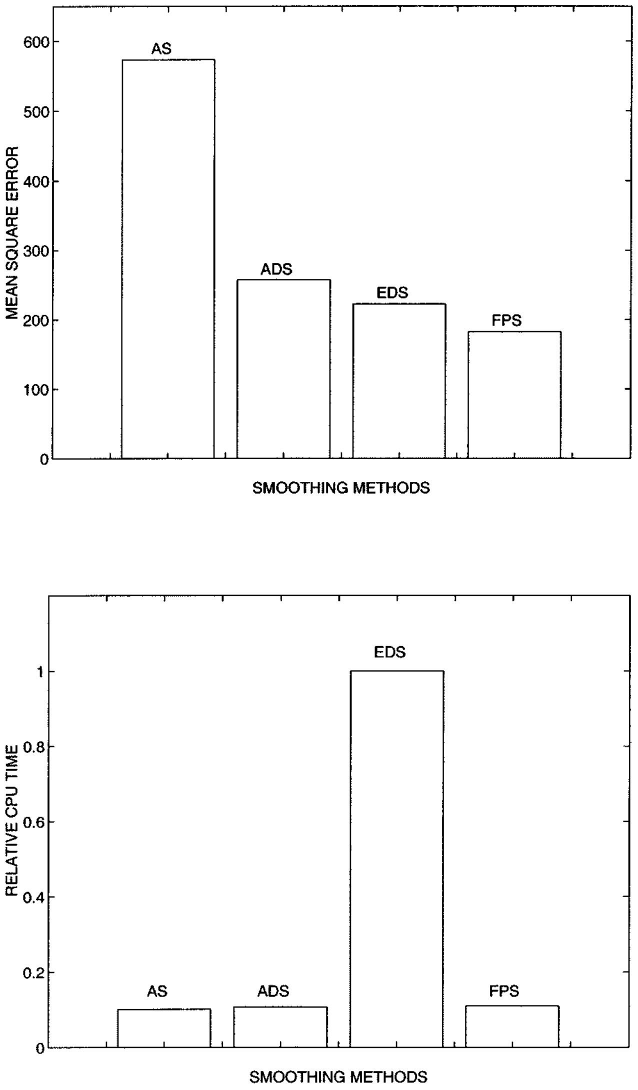


Fig. 11. Comparisons in restoration and computing time. (a) MSE's for the optimal snapshots in different smoothing algorithms. (b) Relative CPU times for 2000 iterations, where the EDS algorithm takes 968 s on an HP/C160 workstation.

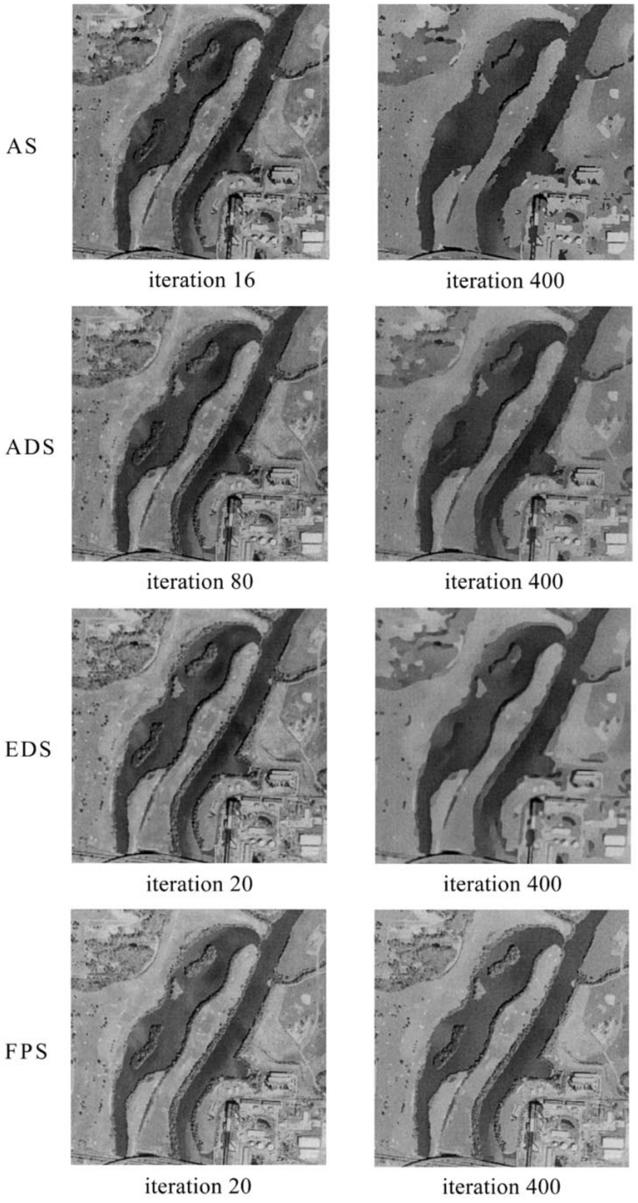
In general, the multiscale adaptive method of Caelli and Bischof is relatively sensitive to noise, which results in either failure to yield meaningful objects in some cases or oversegmentation. In comparison, our algorithm yields considerably better segmentation results.

In addition, another modified LEGION network was recently proposed for specially segmenting medical images [32]. This algorithm produces satisfactory results for MRI images. In comparison, our algorithm yields more homogeneous regions and more precise boundaries between different regions though it produces worse results in extracting extracranial tissue.

B. Comparison in Smoothing

Nonlinear smoothing techniques have been recognized as an effective way of preprocessing in early vision. A nonlinear smoothing technique can remove noise and preserve significant discontinuities. Adaptive smoothing is a classical way to perform nonlinear smoothing, applying a smoothing operator that adapts itself to local image structure [8], [19], [25], [28], [40]. In order to demonstrate its effectiveness, we compare our FPS algorithm to three other adaptive smoothing algorithms: adaptive smoothing (AS) [28], anisotropic diffusion smoothing (ADS) [25], and edge-enhanced diffusion smoothing (EDS) [40], respectively. These algorithms have proved to be useful in many domains of image processing [41]. Like comparison in segmentation, we use three images belonging to different types for comparison. In simulations, we determine parameter values in the aforementioned smoothing algorithms by searching for the parameter space as suggested in their original work, and we report only the best results here.

Fig. 12 shows the smoothing results for the aerial image in Fig. 7(c) produced by the four algorithms. Due to unavailability



iteration 16 iteration 400

iteration 80 iteration 400

iteration 20 iteration 400

iteration 20 iteration 400

Fig. 12. Smoothing results for the aerial image in Fig. 7(c) by different smoothing algorithms, where each row shows a set of snapshots for a smoothing algorithm. For the AS algorithm, $k = 6.0$; for the ADS algorithm, $K = 1.5$; for the EDS method with $\lambda = 3.0$ and $\sigma = 1.5$; for the FPS algorithm, $R = 2$, $s = 8.0$, $\kappa = 40.0$, and $\theta_\sigma = 0.02$.

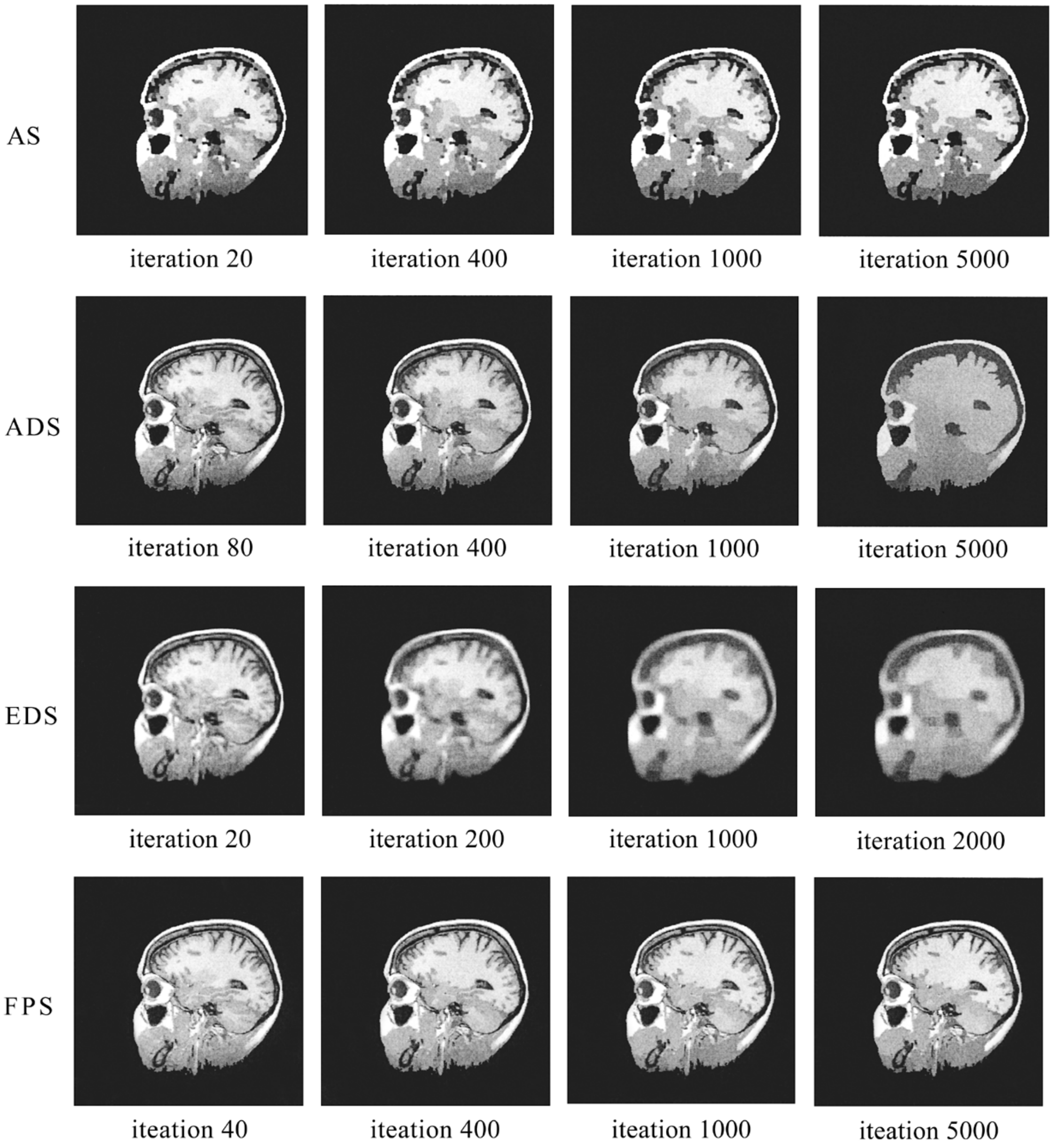


Fig. 13. Smoothing results of the MRI image in Fig. 8(c) by different smoothing algorithms, where each row shows a set of snapshots for a smoothing algorithm. For the AS algorithm, $k = 4.5$; for the ADS algorithm, $K = 1.5$; for the EDS algorithm, $\lambda = 3.0$ and $\sigma = 1.0$; for the FPS algorithm, $R = 1$, $s = 6.0$, $\kappa = 80.0$, and $\theta_\sigma = 0.01$.

of its ground truth for a real image, we manually select the best possible results based on edge maps For each algorithm, two snapshots are shown and arranged in the same way as in Fig. 10. The first snapshot corresponding to the best possible result and the second one is the smoothed image when the algorithms are terminated after 400 iterations. As shown in the first snapshot, nonlinear smoothing effects are reasonably achieved by the four algorithms. Except for the AS algorithm, the results produced by the other three preserve most of significant features; e.g, the bridge and boundaries between the river and its surroundings (the bank and the islands). As shown in the second snapshot, however, further smoothing in 400 iterations causes some significant features, e.g. the boundaries between the river and the islands, to vanish in all those except our algorithm. This demonstrates that our algorithm is the only one that does not suffer from the termination problem.

Fig. 13 shows smoothing results of the MRI image in Fig. 8(c) produced by the four algorithms. The same method as described above is used to select the best possible results. For each algorithm, four snapshots are shown and arranged in the same way as in Fig. 10. All the algorithms can preserve most of the significant discontinuities in the image after a few iterations. For instance, the cortex and the cerebellum become more homogeneous and their boundary gets clearer. Except for our algorithm, however, further smoothing causes most of the important features to disappear.

In general, the above results as well as others not reported due to space demonstrate that our smoothing algorithm in comparison yields good results though two more parameters are used in our algorithm. In particular, our algorithm does not suffer from the termination problem and manages to preserve significant features for a long iteration period, which distinguishes our algorithm from other smoothing algorithms [41].

VI. DISCUSSIONS

The fundamental goal in adaptive smoothing is to preserve discontinuities due to significant features while eliminating variations due to noise and irrelevant features. For most existing adaptive smoothing algorithms such as those in [25], [28], [40], only a local discontinuity measure, e.g., spatial gradient, is used to determine significant features, which tends to be sensitive to noise. In order to obtain good results, one needs to specify a nonlinear mapping from a local discontinuity measure to a diffusion speed and then choose a termination time to achieve a compromise between noise removal and feature preservation. Unlike those algorithms, our method adopts two complementary discontinuity measures. Lateral discontinuities provide a useful context to distinguish discontinuities due to significant features from those due to noise. Moreover, they are used to control the diffusion speed, which results in more robust performance. This also greatly alleviates the termination problem; as shown by our numerical simulations, our algorithm can reach a near steady state in a few iterations for most tasks.

LEGION is a neurocomputational mechanism for oscillatory correlation [33], [38].The notions of lateral potential and dynamic normalization were proposed [39] and these mechanisms were supposed to operate in the initialization stage, where weights rapidly reach a steady state. As a result, the initialization process does not change the dynamics of LEGION and the rigorous results on LEGION [33] still hold. Our proposed weight adaptation method provides an alternative way for fast weight dynamics to remove noise and to extract features. It still operates in the initialization stage. Simulation results show that only a few iterations are required for most tasks, which indicates that our weight adaptation can reach near steady state rapidly. Like [39], our proposed weight adaptation method does not alter LEGION dynamics. However, our method leads to substantial performance improvements in noise removal and feature preservation.

As mentioned earlier, iterative smoothing has been widely used to deal with noisy images, and some form of smoothing seems inescapable for early visual processing. How can this be achieved in the visual system? Our notion of smoothing through weight dynamics points to a possible way by which smoothing effects can be achieved in the brain. Fast-changing synapses were suggested by von der Malsburg [35], [36] who forcefully argued for its biological plausibility (see also [6]). On the same conceptual ground, dynamic weight normalization was used to enhance the quality of synchronization in a network of neural oscillators [33], [37], [38]. Our suggestion for weight adaptation follows the same line of argument. What is important is that such weight dynamics is temporary in nature and adapts to new stimuli (images) rapidly without keeping any permanent trace. Given that many powerful learning (weight update) methods have been proposed in neural networks, it is reasonable to expect that future research along the same path will reveal more effective techniques for image processing and computer vision.

ACKNOWLEDGMENT

The authors would like to thank T. Caelli for allowing them to use their multiscale adaptive segmentation software, J. Weickert for helping them understand his edge-enhancing anisotropic diffusion method, N. Shareef for providing MRI images, and The Ohio State University Center for Mapping for providing their satellite image database. They would also like to thank anonymous referees for their suggestions.

REFERENCES

[1] R. Adams and L. Bischof, "Seeded region growing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, pp. 641–647, 1994.

[2] N. Ahuja, "A transform for multiscale image segmentation by integrated edge and region detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, pp. 1211–1235, 1996.

[3] C. M. Bishop, *Neural Networks for Pattern Recognition*. Oxford, U.K.: Oxford Univ. Press, 1995.

[4] T. Caelli and W. Bischof, *Machine Learning and Image Interpretation*. New York: Plenum, 1997.

[5] M. Cheriet, J. N. Said, and C. Y. Suen, "A recursive thresholding technique for image segmentation," *IEEE Trans. Image Processing*, vol. 7, pp. 918–921, 1998.

[6] F. Crick, "Function of the thalamic reticular complex: The searchlight hypothesis," *Proc. Nat. Acad. Sci. USA*, vol. 81, pp. 4586–4590, 1984.

[7] G. Foresti, V. Murino, C. S. Regazzoni, and G. Vernazza, "Group of rectilinear segments by labeled Hough transform," *CVGIP: Image Understand.*, vol. 58, pp. 22–42, 1994.

[8] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-6, pp. 721–741, 1984.

[9] D. Geman, S. Geman, C. Graffigne, and P. Dong, "Boundary detection by constrained optimization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, pp. 609–628, 1990.

[10] E. M. Gurari and H. Wechsler, "On the difficulties involved in the segmentation of pictures," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-4, pp. 304–306, 1982.

[11] J. F. Haddon and J. F. Boyce, "Image segmentation by unifying region and boundary information," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, pp. 929–948, 1990.

[12] R. M. Haralick, "Statistical and structural approaches to texture," *Proc. IEEE*, vol. 67, pp. 786–804, 1979.

[13] R. M. Haralick and L. G. Shapiro, "Image segmentation techniques," *Comput. Graph. Image Process.*, vol. 29, pp. 100–132, 1985.

[14] R. M. Horowitz and T. Pavlidis, "Picture segmentation by a tree traversal algorithm," *J. ACM*, vol. 23, pp. 368–388, 1976.

[15] J. Koenderink, "The structure of images," *Biol. Cybern.*, vol. 50, pp. 363–370, 1984.

[16] J. Koh, M. Suk, and S. M. Bhandarkar, "A multilayer self-organizing feature map for range image segmentation," *Neural Networks*, vol. 8, pp. 67–86, 1995.

[17] R. Kohler, "A segmentation system based on thresholding," *Comput. Graphics Image Process.*, vol. 15, pp. 319–338, 1981.

[18] T. Kohonen, *Self-Organizing Maps*. New York: Springer-Verlag, 1995.

[19] A. Lev, S. Zucker, and A. Rosenfeld, "Iterative enhancement of noisy images," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-7, pp. 435–442, 1977.

[20] S. P. Liou, A. H. Chiu, and R. C. Jain, "A parallel technique for signal-level perceptual organization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, pp. 317–325, 1991.

[21] B. S. Manjunath and R. Chellappa, "A unified approach to boundary perception: Edges, textures, and illusory contours," *IEEE Trans. Neural Networks*, vol. 4, pp. 96–108, Jan. 1993.

[22] R. Mohan and R. Nevatia, "Perceptual organization for scene segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, pp. 616–635, 1992.

[23] N. R. Pal and S. K. Pal, "A review on image segmentation techniques," *Pattern Recognit.*, vol. 26, pp. 1277–1294, 1993.

[24] T. Pavlidis and Y. T. Liow, "Integrating region growing and edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, pp. 225–233, 1990.

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

[26] C. Revol and M. Jourlin, "A new minimum variance region growing algorithm for image segmentation," *Pattern Recognit. Lett.*, vol. 18, pp. 249–258, 1997.

[27] B. D. Ripley, *Pattern Recognition and Neural Networks*. Cambridge, MA: Cambridge Univ. Press, 1996.

[28] P. Saint-Marc, J. S. Chen, and G. Medioni, "Adaptive smoothing: A general tool for early vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, pp. 514–529, 1991.

[29] S. Sarkar and K. L. Boyer, "On optimal infinite impulse response edge detection filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, pp. 1154–1171, 1991.

[30] ——, "Integration, inference, and management of spatial information using Bayesian networks: Perceptual organization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 15, pp. 256–274, 1993.

[31] R. J. Schalkoff, *Digital Image Processing and Computer Vision*. New York: Wiley, 1989.

[32] N. Shareef, D. L. Wang, and R. Yagel, "Segmentation of medical images using LEGION," *IEEE Trans. Med. Imag.*, vol. 18, pp. 74–91, 1999.

[33] D. Terman and D. L. Wang, "Global competition and local cooperation in a network of neural oscillators," *Physica D*, vol. 81, pp. 148–176, 1995.

[34] B. van der Pol, "On 'relaxation oscillations'," *Philos. Mag.*, vol. 2, pp. 978–992, 1926.

[35] C. von der Malsburg, "The Correlation theory of brain function," Max-Planck-Institut for Biophys. Chem., Göttingen, Germany, Internal Rep. 81-2, 1981.

[36] C. von der Malsburg and W. Schneider, "A neural cocktail-part processor," *Biol. Cybern.*, vol. 54, pp. 29–40, 1986.

[37] D. L. Wang, "Emergent synchrony in locally coupled neural oscillators," *IEEE Trans. Neural Networks*, vol. 6, pp. 941–948, 1995.

[38] D. L. Wang and D. Terman, "Locally excitatory globally inhibitory oscillator networks," *IEEE Trans. Neural Networks*, vol. 6, pp. 283–286, 1995.

[39] ——, "Image segmentation based on oscillatory correlation," *Neural Computat.*, vol. 9, pp. 805–836, 1997. (for errata see *Neural Computat.*, vol. 9, pp. 1623-1626, 1997).

[40] J. Weickert, "Theoretical foundations of anisotropic diffusion in image processing," *Computing*, vol. 11, pp. 221–236, 1996.

[41] ——, "A review of nonlinear diffusion filtering," in *Scale-Space Theory in Computer Vision*, B. Romeny, L. Florack, J. Koenderink, and M. Viergever, Eds. New York: Springer-Verlag, 1997, pp. 3–28.

[42] R. Wilson and M. Spann, *Image Segmentation and Uncertainty*. New York: Wiley, 1996.

[43] S. C. Zhu and Y. Yuille, "Region competition: Unifying snakes, region growing, and Bayes/MDL for multiband image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, pp. 884–900, 1996.

[44] S. W. Zucker, "Region growing: Childhood and adolescence," *Comput. Graph. Image Process.*, vol. 5, pp. 382–399, 1976.

Ke Chen (M'97) was born in Xuzhou, China, in 1963. He received the B.S. and the M.S. degrees from Nanjing University, China, in 1984 and 1987, respectively, and the Ph.D. degree from Harbin Institute of Technology, China, in 1990, all in computer science.

Since 1994, he has been on the Faculty of Peking University, China, where he is now a Full Professor of information science. From 1990 to 1992 he was a Postdoctoral Researcher at Tsinghua University, China. From 1992 to 1993 he was a Postdoctoral Fellow of the Japan Society for Promotion of Sciences and worked at Kyushu Institute of Technology. From January 1997 to December 1998 he was a Visiting Scientist at The Ohio State University, Columbus. Currently, he is a Visiting Scientist in Microsoft Research China. He has published over 50 technical papers in refereed journals and international conferences. His current research interests include neural computation, machine learning, pattern recognition, and their applications in machine perception.

Dr. Chen was a recipient of the Top Award for Progress of Science and Technology from the National Education Ministry in China and a recipient of the Yang-Wang Academician Award for Distinguished Professors in Information Science and Technology from Peking University in 1999. He is a member of the IEEE Computer Society and a senior member of CIE.

DeLiang Wang (M'94) received the B.S. and the M.S. degrees from Peking (Beijing) University, Beijing, China, in 1983 and 1986, respectively, and the Ph.D. degree in 1991 from the University of Southern California, Los Angeles, all in computer science.

From July 1986 to December 1987 he was with the Institute of Computing Technology, Academia Sinica, Beijing. Since 1991 he has been with the Department of Computer and Information Science and the Center for Cognitive Science, The Ohio State University, Columbus, where he is currently an Associate Professor. From October 1998 to September 1999 he was a Visiting Scholar in the Vision Sciences Laboratory at Harvard University, Cambridge, MA. His present research interests include neural networks for perception, neurodynamics, neuroengineering, and computational neuroscience.

Dr. Wang is a member of IEEE Computer Society and the International Neural Network Society. He is a recipient of the 1996 U.S. Office of Naval Research Young Investigator Award.

Xiuwen Liu received the B.E. degree in computer science from Tsinghua University, Beijing, China, in 1989, and the M.S. degrees in geodetic science and surveying and in computer and information science in 1995 and 1996, respectively, and the Ph.D. degree in computer and information science, in 1999, all from The Ohio State University, Columbus.

From 1989 to 1993, he was with the Department of Computer Science and Technology, Tsinghua University, China. He is currently a Research Associate in the Department of Computer and Information Science, The Ohio State University. His current research interests include image segmentation, statistical texture modeling, motion analysis, machine learning, image-based computer graphics, neural networks, computational perception, and pattern recognition.