A VARIANT OF THE LEVEL SET METHOD AND APPLICATIONS TO IMAGE SEGMENTATION

JOHAN LIE*, MARIUS LYSAKER†, AND XUE-CHENG TAI‡

Abstract. In this paper we propose a variant of the level set formulation for identifying curves separating regions into different phases. In classical level set approaches, the sign of n level set functions are utilized to identify up to 2^n phases. The novelty in our approach is to introduce a piecewise constant level set function and use each constant value to represent an unique phase. If 2^n phases should be identified, the level set function must approach 2^n predetermined constants. We just need one level set function to represent 2^n unique phases and this gain the storage capacity. Further, the reinitializing procedure requested in classical level set methods is superfluous using our approach. The minimization functional for our approach is convex and differentiable and thus avoid some of the problems with the non-differentiability of the Delta and Heaviside functions. Numerical examples will be given and we shall also compare our method with related approaches.

Key words. Level set, energy minimizing, partial differential equations, segmentation.

1. Introduction to Related Level Set Methods. A function u defined on an open and bounded domain $\Omega \in \mathbb{R}^m$ may have different properties in distinct regions of Ω. In many applications, one needs to separate Ω into an union of subdomains, i.e. $\Omega = \cup_{i=1}^{n} \Omega_i$, so that distinctive properties of u can be linked to different subdomains and incorporated into a modeling process. Several approaches have been used to find a propitious partition of Ω, and some of them are shortly described below. For the purpose of illustration, only the two-dimensional case is considered here.

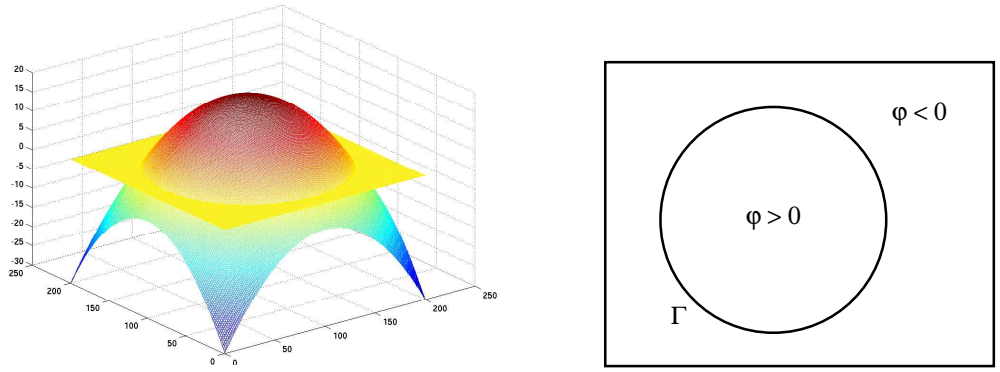
One approach to find a partition of Ω is to use an active contour. The basic idea is to let a closed curve C separate Ω into two disjunctive subdomains. Define the region inside C for Ω_1, and the region outside C for Ω_2. The curve evolves subject to constraints from u together with some smoothness requirements of C. Traditionally, the curve C is parameterized as $\mathbf{x} = \mathbf{x}(s)$, $s \in [0, 1]$. A convenient way to approximate an explicit representation of C, is to divide s into a finite set of points $0 = s_0 < \cdots < s_{i-1} < s_i < s_{i+1} \cdots < s_n = 1$, where the subintervals $[s_i, s_{i+1}]$ can vary in size and then update the particle $\mathbf{x}$ in an iterative way. For some applications, it might be difficult to handle geometrical changes of the curve using this kind of explicit approach. Numerical instability could also be a problem, since the points on the curve may get clustered.

A recent and successful way to represent an evolving curve is the level set method invented by Osher and Setian in [11]. With the level set representation a change of topology, such as merging and breaking of the curve, can be handled in an efficient way. The curve that divide Ω into two subdomains is represented implicitly as the zero level set curve $\Gamma = \{\mathbf{x} \in \Omega,\ \phi(\mathbf{x}) = 0\}$ of a higher dimension level set function ϕ. Rather than evolving the curve itself, the level set approach builds Γ into a function ϕ. Normally, ϕ is initialized as the signed distance function:

$$\phi(\mathbf{x}) = \begin{cases} \text{distance}(\mathbf{x}, \Gamma), & \mathbf{x} \in \text{interior of } \Gamma \\ -\text{distance}(\mathbf{x}, \Gamma), & \mathbf{x} \in \text{exterior of } \Gamma. \end{cases} \tag{1.1}$$

In Fig. 1.1(a) a simple level set function ϕ is illustrated, and its associated interface Γ is given in Fig. 1.1(b). Γ divide Ω into two separate sub-domains. By moving ϕ

*Department of Mathematics, University of Bergen, Norway (johanl@mi.uib.no).
†Department of Mathematics, University of Bergen, Norway (mariusl@mi.uib.no).
‡Department of Mathematics, University of Bergen, Norway (tai@mi.uib.no).



(a) The level set function ϕ. (b) The interface Γ.

FIG. 1.1. *Plot of the higher dimension function* $\phi(\mathbf{x})$ *and the interface* $\Gamma = \{\mathbf{x} \in \Omega, \ \ \phi(\mathbf{x}) = 0\}$.

up or down at different locations, the interface evolves. To divide Ω into more than two domains, one needs to use multiple level set functions [17, 12, 5, 16]. One of the concerns of multiple level set methods is to avoid vacuum and overlap between the different phases [17]. In [16], the following approach was used to create multiple phases without vacuum and overlaps. Assume we introduce two closed curves: Γ_1 and Γ_2, where each curve Γ_i are associated with a level set function ϕ_i, $i = 1, 2$. The domain Ω can now be divided into 4 parts:

$$\begin{aligned} \Omega_1 =&\{\mathbf{x} \in \Omega, \ \ \phi_1 > 0, \ \ \phi_2 > 0\}, \quad \Omega_2 = \{\mathbf{x} \in \Omega, \ \ \phi_1 > 0, \ \ \phi_2 < 0\}, \\ \Omega_3 =&\{\mathbf{x} \in \Omega, \ \ \phi_1 < 0, \ \ \phi_2 > 0\}, \quad \Omega_4 = \{\mathbf{x} \in \Omega, \ \ \phi_1 < 0, \ \ \phi_2 < 0\}. \end{aligned} \tag{1.2}$$

By generalizing further, n level set functions give the possibility of 2^n regions [17, 5]. These regions will be disjunct (no overlap) and their union is Ω (no vacuum).

In this work, we propose an alternative approach. Assume that we need to find n regions $\{\Omega_i\}_{i=1}^n$ which form a portion of Ω. Some of the regions can be empty. This means that we can over-estimate the total number of phases for practical applications and superfluous phases will disappear at convergence. In order to find the regions, we just need to find a piecewise constant function which takes values:

$$\phi = i \text{ in } \Omega_i, \quad i = 1, 2, \cdots, n. \tag{1.3}$$

The discontinuities of ϕ give us the curves that separate the regions. For some applications, we could formulate the identification of the curves as a minimization problem with respect to ϕ under some constraints. We believe that this level set formulation can be applied in different applications where the classical level set methods are used today. In this work, we restrict ourself to one special case: image segmentation. For such an application, we shall formulate the segmentation problem as a minimization problem with respect to ϕ.

The idea of using one level set function to represent several phases is however not completely new. In island dynamics models for epitaxial growth of thin films a large number of individual interfaces have to be identified. In [9, 6] they use one level set function to track islands on different monolayers, i.e., they let $\phi = 0$ represent the

island boundaries of the first monolayer, $\phi = 1$ represent the island boundaries of the second monolayer and so on. By using just one level set function rather than one per layer, their algorithm is kept simple and memory costs are kept low. We will use a similar idea to track different segments in an image. However, to get this terrace level set formulation to fit into a segmentation framework some innovative work has been done in this paper. These issues are addressed in the next section.

Further, we would also like to mention that piecewise constant level set methods have been used in image segmentation before. In [14] they defined an energy functional dependent on $H(\phi_i)$ where $\phi_i = \pm 1$, and they simply checked if the energy decrease or not when one of the level set functions changed sign at a given point. This method is extremely fast, but the authors give some examples where this method will fail.

Let us conclude this first section with the outline for this paper. In section 2 we introduce a functional motivated from an optimal approximation problem. For the approximation problem we will use the partition described in (1.3) to express a general function as a piecewise constant function. In section 3 the augmented Lagrangian method is introduced as a tool to solve the optimal approximation problem. In this section we also describe our algorithm, together with some other implementation details. The novelty in our approach, compared with classical level set methods, is reported in section 4. Finally, section 5 is devoted to numerical examples.

2. Representing Piecewise Constant Functions. We use image segmentation as an example to show that our method can be utilized to identify curves separating regions. Let u_0 be a function defined in a domain Ω, where u_0 reflects the intensity values for a given gray scale image, possibly imposed with noise. It is assumed that the intensity value of the true image is slowly varying inside each phase, but has large jumps between the phases. In order to find the discontinuities, we try to find a piecewise constant function to approximate the function u_0 and the discontinuities of the piecewise constant function will give us the boundary between the different phases.

In order to find a piecewise constant function, we need to find the value for the constants and the location of the discontinuities. To find the locations of the discontinuities, the curve that separates the regions needs to be determined. Mumford and Shah [10] proposed to solve the "minimal" partition problem to approximate a function u_0 by a piecewise constant function u:

$$\inf_{u,\Gamma} \Big\{ F^{MS}(u,\Gamma) = \sum_i \int_{\Omega_i} |c_i - u_0|^2 dx + \nu|\Gamma| \Big\}, \tag{2.1}$$

where $\Omega = \cup_i \Omega_i \cup \Gamma$. They tried to find a decomposition Ω_i of Ω, where $u{=}c_i$ (constant) inside each connected component Ω_i. The positive parameter ν controls the length of the curve Γ. For a fixed Γ, we see that (2.1) is minimized when $c_i = mean(u_0)$ in Ω_i. A challenge when solving (2.1) is to find a unique representation of the parameterized curve Γ. To accommodate this problem, Chan and Vese [2] reformulated and solved (2.1) using a level set approach:

$$\inf_{\mathbf{c},\phi} \Big\{ F(c_1,c_2,\phi) = \int_\Omega |c_1{-}u_0|^2 H(\phi)\, dx + \int_\Omega |c_2{-}u_0|^2 (1{-}H(\phi))\, dx + \nu \int_\Omega |\nabla H(\phi)|\, dx \Big\}, \tag{2.2}$$

where the Heaviside function $H(\phi)$ equals 1 if $\phi \geq 0$ and equals 0 if $\phi < 0$. The regularization term $\int_\Omega |\nabla H(\phi)|\, dx$ measures the length of the interface [1] and favor

smooth curves. Minimizing $F(c_1, c_2, \phi)$ with respect to c_1 and c_2, we obtain: $c_1 = mean(u_0)$ if $\phi > 0$ and $c_2 = mean(u_0)$ if $\phi < 0$.

The active contour model without edges (2.2) can only divide Ω into two distinct regions. Later, Chan and Vese generalized this 2-phase active contour model to a multiphase level set formulation [5, 16]. By using a multiphase level set representation, it is possible to solve the "minimal" partition problem (2.1) for an arbitrary number of partitions. Below we show the energy functional for a 4-phase partition, involving 2 level set functions:

$$\begin{aligned} F(\mathbf{c}, \phi_1, \phi_2) = &\int_\Omega |c_1 - u_0|^2 H(\phi_1)H(\phi_2)\, dx + \int_\Omega |c_2 - u_0|^2 H(\phi_1)(1 - H(\phi_2))\, dx \\ &+ \int_\Omega |c_3 - u_0|^2 (1 - H(\phi_1))H(\phi_2)\, dx + \int_\Omega |c_4 - u_0|^2 (1 - H(\phi_1))(1 - H(\phi_2))\, dx \\ &+ \nu \int_\Omega |\nabla H(\phi_1)|\, dx + \nu \int_\Omega |\nabla H(\phi_2)|\, dx. \end{aligned} \tag{2.3}$$

Once the constant values $\{c_i\}_{i=1}^4$ and the level set functions ϕ_1, ϕ_2 are identified, the piecewise constant function u can be written as:

$$\begin{aligned} u = &c_1 H(\phi_1)H(\phi_2) + c_2 H(\phi_1)(1 - H(\phi_2)) + c_3(1 - H(\phi_1))H(\phi_2) \\ &+ c_4(1 - H(\phi_1))(1 - H(\phi_2)). \end{aligned} \tag{2.4}$$

Given u on this form, an alternative formulation of (2.3) was suggested in [4, 7]. Here the authors tried to minimize the following energy functional directly with u given as in (2.4):

$$F(\mathbf{c}, \phi_1, \phi_2) = \int_\Omega |u - u_0|^2 dx + \nu \int_\Omega |\nabla \phi_1| dx + \nu \int_\Omega |\nabla \phi_2| dx. \tag{2.5}$$

The above functional is essentially an application of the functional given in [4] for the image segmentation problem. Omitting the regularization term, functional (2.3) equals functional (2.5) except from one scalar. However, the gradients are not equal, producing different numerical schemes. Tests in [7] indicate somewhat better performance using the gradient associated with functional (2.5). Using the approaches above, two level set functions must be utilized to find four disjunct regions covering Ω. Correspondingly, we need to identify n functions and 2^n constants in order to identify the regions for 2^n phases.

Using our new approach as shown in (1.3), we just need one function to identify all the phases. Associated with ϕ, we define the basis functions ψ_i as:

$$\psi_i = \frac{1}{\alpha_i} \prod_{\substack{j=1 \\ j \neq i}}^{n} (\phi - j) \quad \text{and} \quad \alpha_i = \prod_{\substack{k=1 \\ k \neq i}}^{n} (i - k). \tag{2.6}$$

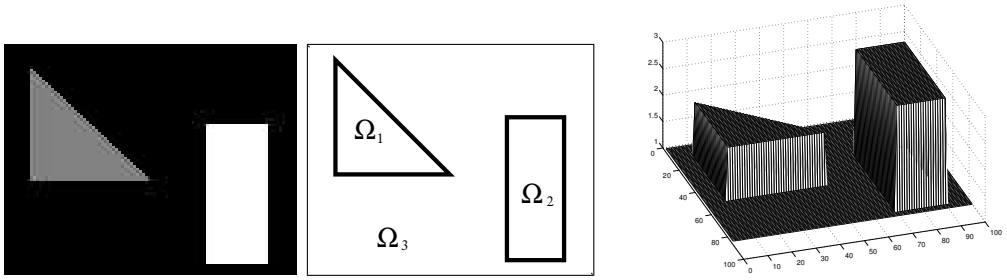
It is clear that a function u given by:

$$u = \sum_{i=1}^{n} c_i \psi_i \tag{2.7}$$

is a piecewise constant function and $u = c_i$ in Ω_i if ϕ is as given in (1.3). The sum in u involves basis functions of polynomial functions of order $n-1$ in ϕ and the unknown coefficients c_i, $i = 1, 2, \cdots n$. Each ψ_i is expressed as a product of linear factors of the form $(\phi - j)$, with the ith factor omitted. Thereupon $\psi_i(\mathbf{x})$=1 for $\mathbf{x} \in \Omega_i$, and $\psi_i(\mathbf{x})$ equals zeros elsewhere as long as (1.3) holds. The basis functions can further be utilized to express arc length of $\partial\Omega_i$ and the area Ω_i of each phase, i.e.

$$|\partial\Omega_i| = \int_\Omega |\nabla\psi_i| \, dx \text{ and } |\Omega_i| = \int_\Omega \psi_i \, dx. \tag{2.8}$$

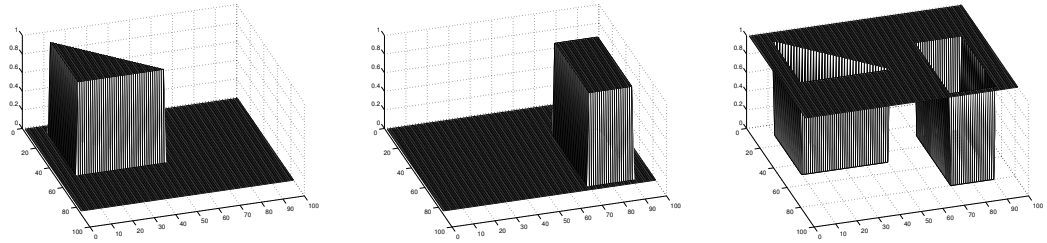
In Fig. 2.1 we demonstrate the relationship between an image function u, the partition of Ω associated with this image, the piecewise constant level set function ϕ and the corresponding basis functions ψ_i, $i = 1, 2, 3$. To ensure that equation (2.7)



(a) Image u with 3 objects.

(b) Partition $\Omega = \cup_{i=1}^{3}\Omega_i$.

(c) Level set function $\phi = 1 \vee 2 \vee 3$.



(d) Basis function ψ_1. $\psi_1 = 0$ or 1.

(e) Basis function ψ_2. $\psi_2 = 0$ or 1.

(f) Basis function ψ_3. $\psi_3 = 0$ or 1.

FIG. 2.1. *The relationship between an image function $u \in BV(\Omega)$, the partition of Ω associated with the image, the piecewise constant level set function ϕ and the corresponding basis functions ψ_i, $i = 1, 2, 3$.*

gives us an unique representation of u, i.e. at convergence different values of ϕ should correspond to different function values $u(\phi)$ in (2.7), we introduce:

$$K(\phi) = (\phi - 1)(\phi - 2)\cdots(\phi - n) = \prod_{i=1}^{n}(\phi - i). \tag{2.9}$$

If a given function $\phi : \Omega \mapsto R$ satisfies

$$K(\phi) = 0, \tag{2.10}$$

there exists a unique $i \in \{1, 2, \cdots, n\}$ for every $x \in \Omega$ such that $\phi(x) = i$. Thus, each point $x \in \Omega$ can belong to one and only one phase if $K(\phi) = 0$. Thus constraint (2.10) is used to guarantee that there is no vacuum and overlap between the different phases. In [17] some other constraints for the classical level set methods was used to avoid vacuum and overlap.

Based on the above observations, we propose to solve the following constrained minimization problem for segmenting an image u_0:

$$\min_{\substack{\mathbf{c},\,\phi \\ K(\phi)=0}} \Big\{ F(\mathbf{c}, \phi) = \int_\Omega |u - u_0|^2 dx + \beta \sum_{i=1}^{n} \int_\Omega |\nabla \psi_i| dx \Big\}. \tag{2.11}$$

We see that large approximation errors will be penalized by the fidelity term $\int_\Omega |u - u_0|^2$. From (2.8), it is clear that the penalization term is just the sum of the length of the subdomain boundaries and the penalization parameter $\beta > 0$ controls the effect of these terms. Another approach could be to regularize ϕ directly in (2.11). This would make the algorithm simpler, but the total variation norm of ϕ is related to both the jumps and the length of the subdomain boundaries. Such a penalization will depend on the jumps which is not preferable for our case, see Fig. 2.1(c). We just want to penalize the length of the curves. If the image u_0 is a piecewise constant function and we take $\beta = 0$, then any minimizer of (2.11) will give a function u such that $u = u_0$ where u is related to the minimizers $\mathbf{c}$ and ϕ as in (2.7).

3. Minimization using augmented Lagrangian Method. We shall use the augmented Lagrangian method to solve the constrained minimization problem (2.11). The augmented Lagrangian functional for this minimization problem is defined as:

$$L(\mathbf{c}, \phi, \lambda) = F(\mathbf{c}, \phi) + \int_\Omega \lambda K(\phi)\, dx + \frac{r}{2} \int_\Omega |K(\phi)|^2 dx, \tag{3.1}$$

where $\lambda \in L^2(\Omega)$ is the multiplier and $r > 0$ is a penalty parameter which needs to be chosen properly. To find a minimizer for (2.11), we need to find the saddle points for L. We use the following Uzawa type algorithm to find a saddle point for $L(\mathbf{c}, \phi, \lambda)$:

Algorithm 1. Choose initial values for ϕ^0 and λ^0. For $k = 1, 2, \cdots$, do:

1. Find $\mathbf{c}^k$ from

$$L(\mathbf{c}^k, \phi^{k-1}, \lambda^{k-1}) = \min_{\mathbf{c}} L(\mathbf{c}, \phi^{k-1}, \lambda^{k-1}). \tag{3.2}$$

2. Use (2.7) to update $u = \sum_{i=1}^n c_i^k \psi_i(\phi^{k-1})$.

3. Find ϕ^k from

$$L(\mathbf{c}^k, \phi^k, \lambda^{k-1}) = \min_{\phi} L(\mathbf{c}^k, \phi, \lambda^{k-1}). \tag{3.3}$$

4. Use (2.7) to update $u = \sum_{i=1}^n c_i^k \psi_i(\phi^k)$.

5. Update the Lagrange-multiplier by

$$\lambda^k = \lambda^{k-1} + rK(\phi^k). \tag{3.4}$$

This algorithm has a linear convergence and its convergence has been analyzed in Kunisch and Tai [8] under a slightly different context. This algorithm has also been used in Chan and Tai [4, 3] for a level set method for elliptic inverse problems.

There are different methods to solve minimization problem (3.3). We have used the gradient method with a fixed step size to solve it. For a given old value of ϕ, we get a new ϕ by:

$$\phi^{new} = \phi^{old} - \Delta t \frac{\partial L}{\partial \phi}(\mathbf{c}^k, \phi^{old}, \lambda^{k-1}). \tag{3.5}$$

The step size Δt is chosen by a trial and error approach and it is fixed during the whole iterative procedure. It is not necessary to solve the minimization problem (3.3) exactly. We terminate the gradient iteration (3.5) when

$$\left\| \frac{\partial L}{\partial \phi}(\mathbf{c}^k, \phi^{new}, \lambda^{k-1}) \right\|_{L^2} \leq \frac{1}{10} \left\| \frac{\partial L}{\partial \phi}(\mathbf{c}^k, \phi^{k-1}, \lambda^{k-1}) \right\|_{L^2} \tag{3.6}$$

is reached or else after each 400 iterations, whereupon we take $\phi^k = \phi^{new}$. To compute $\frac{dL}{d\phi}$ we utilize the chain rule to get:

$$\frac{\partial L}{\partial \phi} = (u - u_0)\frac{\partial u}{\partial \phi} - \beta \sum_{i=1}^{n} \nabla \cdot \Big(\frac{\nabla \psi_i}{|\nabla \psi_i|} \Big) \frac{\partial \psi_i}{\partial \phi} + \lambda \frac{\partial K}{\partial \phi} + rK \frac{\partial K}{\partial \phi}. \tag{3.7}$$

It is easy to get $\partial u/\partial \phi$, $\partial \psi_i/\partial \phi$ and $\partial K/\partial \phi$ from (2.6), (2.7) and (2.9).

As u is linear with respect to the c_i values, we see that L is quadratic with respect to c_i. Thus the minimization problem (3.2) can be solved exactly. Note that

$$\frac{\partial L}{\partial c_i} = \int_\Omega \frac{\partial L}{\partial u}\frac{\partial u}{\partial c_i} = \int_\Omega (u - u_0)\psi_i \, dx, \qquad \text{for } \; i = 1, 2, \cdots n. \tag{3.8}$$

Therefore, the minimizer of (3.2) satisfies a linear system of equations $A\mathbf{c}^k = b$:

$$\sum_{j=1}^{n} \int_\Omega (\psi_j \psi_i) c_i^k \, dx = \int_\Omega u_0 \psi_i \, dx, \qquad \text{for } \; i = 1, 2, \cdots n. \tag{3.9}$$

In the above $\psi_j = \psi_j(\phi^k)$, $\psi_i = \psi_i(\phi^k)$ and thus, $\mathbf{c}^k = \{c_i^k\}_{i=1}^n$ depends on ϕ^k. In our simulations, we form the matrix A and vector b and solve the equation $A\mathbf{c}^k = b$ by an exact solver.

Remark 1. *The updating for the constant values in (3.2) is very ill-posed. A small perturbation of the ϕ function produces a large perturbation for the $\mathbf{c}$ values. Due to this reason, we have tried out a variant of Algorithm 1. In each iteration we alternate between (3.3) and (3.4), while (3.2) is only carried out if $\|K(\phi^{new})\|_{L^2} < \frac{1}{10}\|K(\phi^{old})\|_{L^2}$. Here, ϕ^{old} denotes the value of ϕ when (3.2) was carried out the last time and ϕ^{new} denotes the current value of ϕ. If we use such a strategy, we can do just one or a few iterations for the gradient scheme (3.5) and Algorithm 1 is still convergent. This strategy is particular efficient when the amount of noisy is high.*

Remark 2. *In Algorithm 1, we give initial values for ϕ and λ. We first minimize with the constant values, and then minimize with the level set function. The multiplier is updated in the end of each iteration. In situations where good initial values for $\mathbf{c}$ are available, an alternative variant of Algorithm 1 may be used, i.e. we first minimize with the level set function followed by an minimization for the constant values and then update the multiplier.*

4. Strength and Weakness of our Level Set Formulation. In this section we will point out the main differences between a classical level set formulation and the variant we have proposed here. First, while multiple level set methods [5, 16] utilize n level set functions to represent 2^n distinct phases, our alternative level set formulation let us represent the same number of phases using only one level set function. This gains the storage capacity.

Further, when minimizing (2.3), the associated Euler-Lagrange equations are [5, 16]:

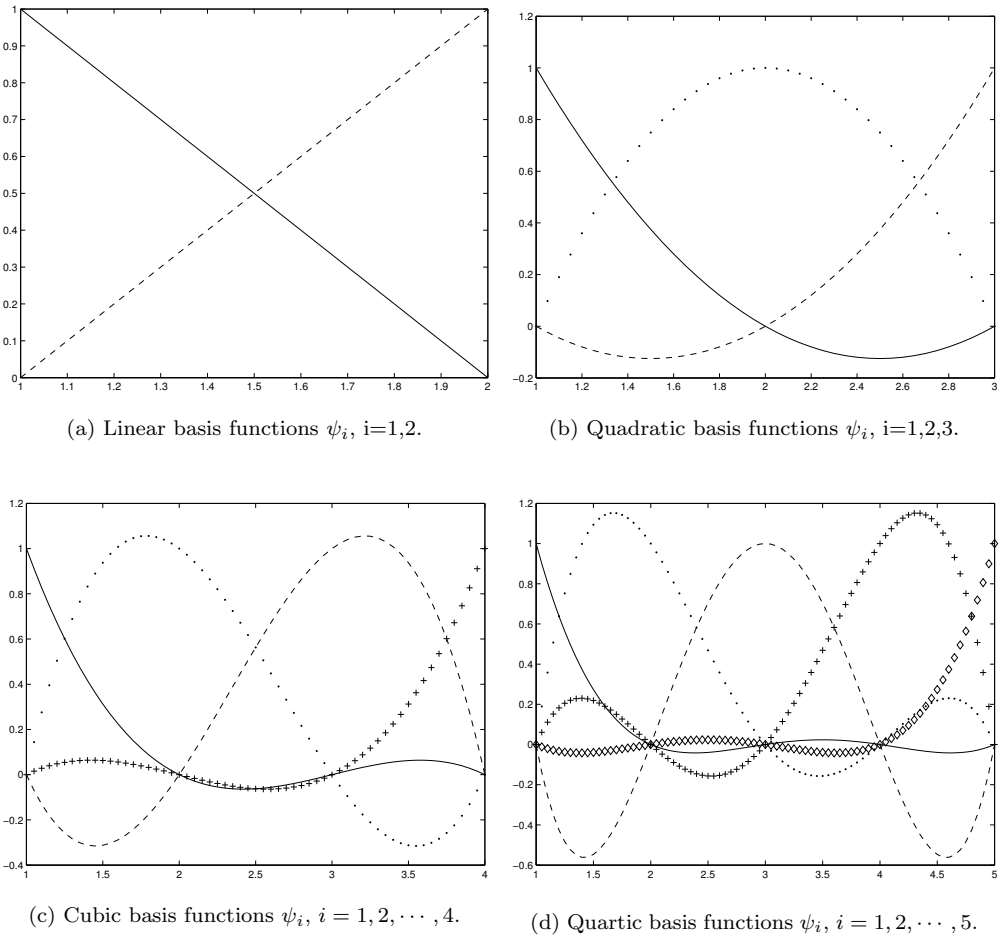
$$\begin{aligned}
\frac{\partial \phi_1}{\partial t} &= \delta(\phi_1)\Big\{\beta \ \mathrm{div}\Big(\frac{\nabla \phi_1}{|\nabla \phi_1|}\Big) \\
&\quad - \Big[\Big((u_0-c_1)^2-(u_0-c_3)^2\Big)H(\phi_2)+\Big((u_0-c_2)^2-(u_0-c_4)^2\Big)(1-H(\phi_2))\Big]\Big\}, \\
\frac{\partial \phi_2}{\partial t} &= \delta(\phi_2)\Big\{\beta \ \mathrm{div}\Big(\frac{\nabla \phi_2}{|\nabla \phi_2|}\Big) \\
&\quad - \Big[\Big((u_0-c_1)^2-(u_0-c_2)^2\Big)H(\phi_1)+\Big((u_0-c_3)^2-(u_0-c_4)^2\Big)(1-H(\phi_1))\Big]\Big\},
\end{aligned} \tag{4.1}$$

where both the Heaviside function $H(\phi)$ and the Delta function $\delta(\phi)$ are involved. Normally, these functions have to be regularized, see for example [2] for different approximations for $H_\epsilon(\phi)$ and $\delta_\epsilon(\phi)$. Without these approximations the level set functions ϕ_i, i=1,2, do not evolve properly, and converge to a local minimum. However, some numerical error do occur when approximations are used for $H_\epsilon(\phi)$ and $\delta_\epsilon(\phi)$. For small ϵ, the Delta function $\delta_\epsilon(\phi)$ is a function with sharp singular layers around the zero level curve of ϕ. We do not use such approximations in our formulation. The energy function for our approach is smooth and convex. The convergence for the Uzawa algorithm can be analyzed and it has a linear convergence rate.

Also, ϕ is normally initialized as the signed distance function like in (1.1). However, as the interface evolves, ϕ will generally drift away from its initialized value. Moreover, ϕ may develop noisy features and steep gradients that are not amenable to finite-difference approximations. For this reason, it is advisable to reinitialize the level set function, i.e. force ϕ to satisfy the partial differential equation: $|\nabla \phi| = 1$. The reinitializing technique is superfluous in our formulation.

Advantages using classical level set methods are the ability to express geometrical quantities and motions. The length of the zero level curve Γ and the area A of the region inside Γ are measured by $|\Gamma| = \int_\Omega |\nabla H(\phi)| \ dx$ and $|A| = \int_\Omega |H(\phi)| \ dx$. Further, the unit normal $\mathbf{N}$ and the mean curvature κ to Γ are defined by $\mathbf{N}=\frac{\nabla \phi}{|\nabla \phi|}$ and $\kappa = \nabla \cdot \mathbf{N}$. Fo our approach, there are some difficulty to find a suitable way to represent the unit normal and the mean curvature for the curves.

Another shortcoming of our method is that higher order polynomials are used as basis functions to represent u in (2.7). Further, the same basis functions are also used to find $\mathbf{c}$ in (3.9). To illustrate this weakness, we depict the linear, quadratic, cubic and quartic basis functions in Fig. 4.1. In Fig. 4.1 we see that one, and only one basis function is different from zero at each root when $K(\phi) = 0$ is fulfilled. However, by inspecting the higher order basis functions ψ_i in Fig. 4.1(c) and (d) we see that even a small perturbation away from each root cause a significant change in the contribution from several of the basis functions. This sensitivity for perturbations makes the coefficient matrix in (3.9) ill-conditioned. To cure this problem we could search for



(a) Linear basis functions ψ_i, i=1,2.

(b) Quadratic basis functions ψ_i, i=1,2,3.

(c) Cubic basis functions ψ_i, $i = 1, 2, \cdots, 4$.

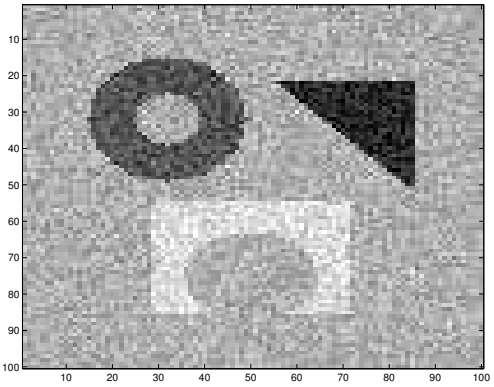
(d) Quartic basis functions ψ_i, $i = 1, 2, \cdots, 5$.

FIG. 4.1. *Different basis functions defined in (2.6).*

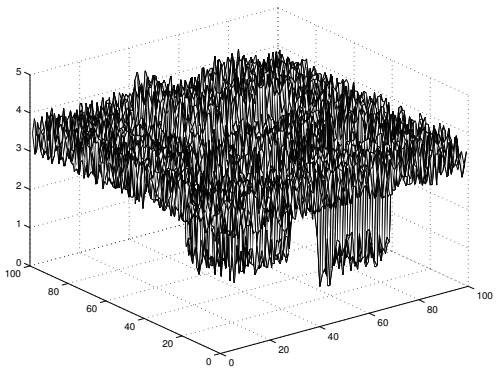
other basis functions with better orthogonality properties, but that is outside the scope of this paper.

To the end, we emphasize that numerical tests have shown that proper initial guess are needed using our approach. Fortunately we can use a simple strategy to approximate the constant coefficients c_i. Basically these constants are given by (3.9) if $\psi_i(\phi)$ is approximately known. Assume now that an observed signal u_0 should be segmented into n phases. From the constraint $K(\phi) = 0$ it follows that ϕ must take values in the set $\{1, 2, \cdots, n\}$. As an approximation for ϕ we let $\phi = u_0$ and thereafter rescale ϕ such that $1 \leq \phi(x, y) \leq n$ for $(x, y) \in \Omega$. Using this approach we have a way to initialize ϕ and find $\psi_i(\phi)$ to get good initial guess for the coefficients c_i, $i = 1, 2, \cdots, n$.

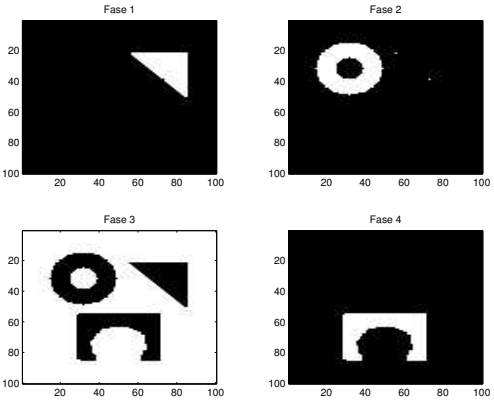
5. Numerical Results. In this section we validate the piecewise constant level set method (PCLSM) with numerical examples. We consider only two-dimensional cases and restrict ourself to gray-scale images, but the model can handle any dimension



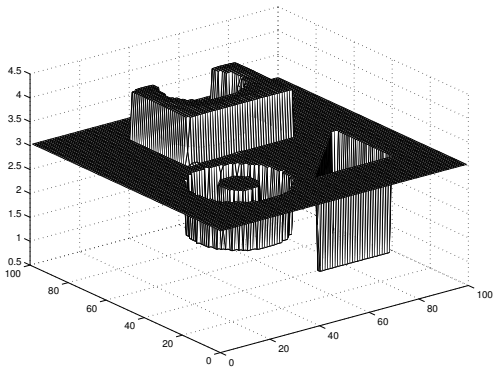
(a) Observed image u_0 (SNR$\approx$ 5.2).



(b) Initial level set function ϕ.



(c) Different phases using PCLSM.

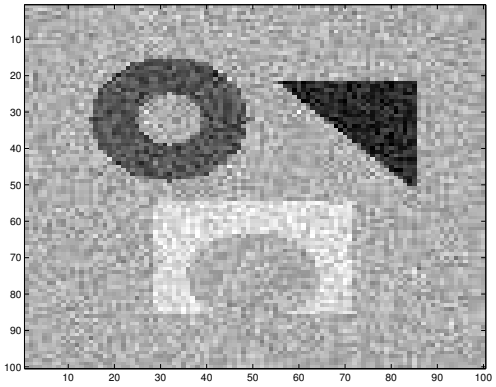


(d) At convergence ϕ approaches 4 constant values.

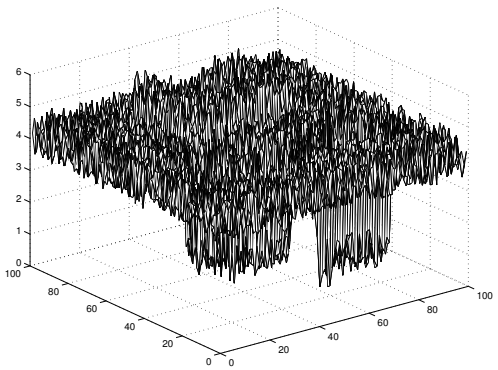
FIG. 5.1. *Each separate phase* $\phi = 1 \vee 2 \vee 3 \vee 4$ *are depicted as a bright region in Fig. 5.1(c).*

and can be extended to vector-valued images as well. Both synthesized images, natural scenes and a MR image are evaluated in this section. Our results will be compared with other related works [5, 16, 15]. The original image is known for some cases we evaluate here. Thereupon it is trivial to find the perfect segmentation result. To complicate such a segmentation process we typically expose the original image with Gaussian distributed noise and use the polluted image as the observation data u_0. To indicate the amount of noise that appears in the observation data, we report the signal-to-noise-ratio: SNR=$\frac{\text{variance of data}}{\text{variance of noise.}}$

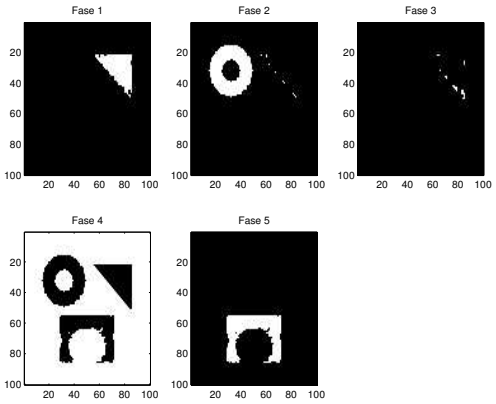
To demonstrate a 4-phase segmentation we begin with a noisy synthetic image containing 3 objects (and a background) as shown in Fig. 5.1(a). This is the same image as Chan and Vese used to examine their multiphase algorithm [5, 16]. A careful evaluation of our algorithm is reported below. The observation data u_0 is given in Fig. 5.1(a) and the only assumption we make is that a 4-phase model should be utilized to find the segments. In Fig. 5.1(d) the ϕ function is depicted at convergence (less



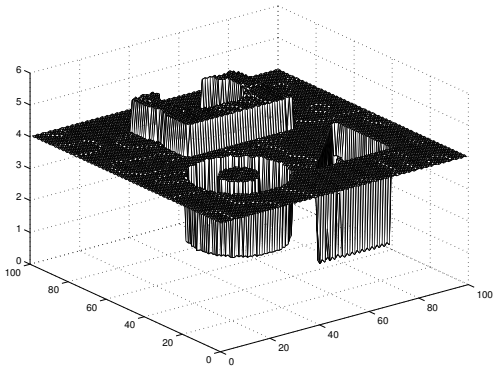
(a) Observed image u_0 (SNR$\approx$ 5.2).



(b) Initial level set function ϕ.



(c) Different phases using PCLSM.

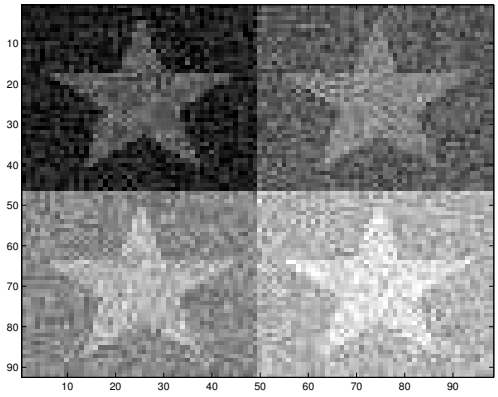


(d) At convergence ϕ approaches 4 constant values.

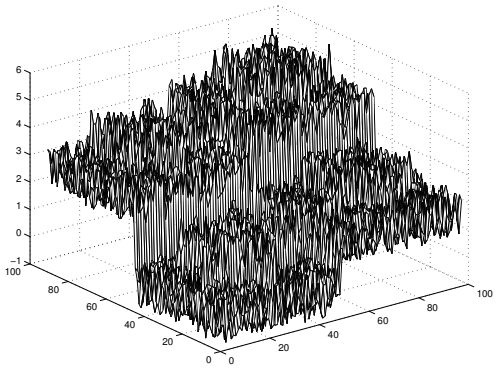
FIG. 5.2. *Each separate phase* $\phi = 1 \vee 2 \vee 3 \vee 4 \vee 5$ *are depicted as a bright region in Fig. 5.2(c).*

than 1000 iteration). ϕ approaches the predetermined constants $\phi = 1 \vee 2 \vee 3 \vee 4$. Each of these constants represents one unique phase as seen in Fig. 5.1(c). Our result is in accordance with what Chan and Vese report in [5, 16].

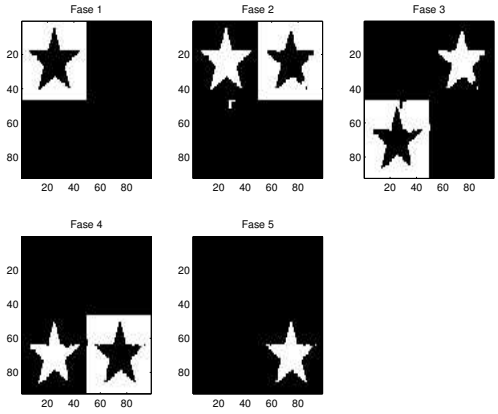
In many applications the number of objects to detect are not known a priori. A robust and reliable algorithm should find the correct segmentation even when the exact number of phases are not known. By introducing a model with more phases than one actually needs, we can find the correct segmentation if all superfluous phases are empty when the algorithm has converged. To see if our algorithm can handle such a case we use Fig. 5.2(a) as the observation image and utilize a 5-phase model. Our results are reported below. One of the 5 phases must be empty if a 5-phase model is used to find a 4-phase segmentation. Due to the high noise level some pixels can easily be misclassified and contribute to the phase that should be empty. The level set function shown in Fig. 5.2(d) approaches the constants $\phi = 1 \vee 2 \vee 4 \vee 5$, except from the few misclassified pixels where $\phi = 3$ as seen in Fig. 5.2(c). By comparing



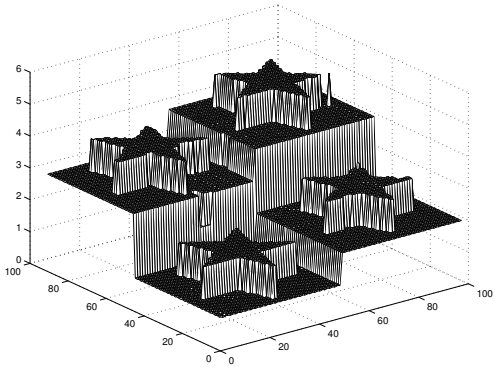
(a) Observed image u_0 (SNR$\approx$ 10.6).



(b) Initial level set function ϕ.



(c) Different phases using PCLSM.

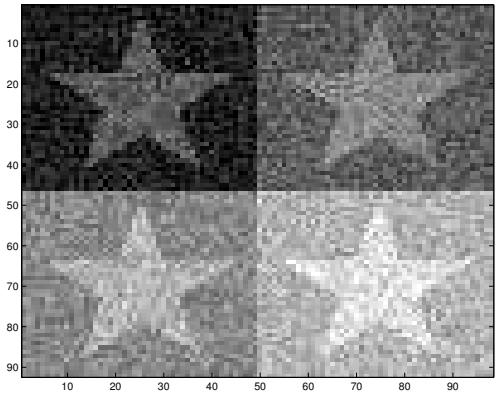


(d) At convergence ϕ approaches 5 constant values.

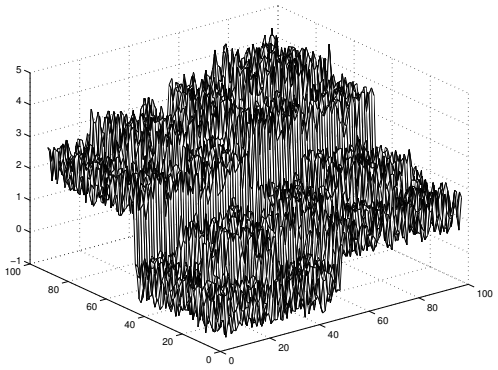
FIG. 5.3. *Each separate phase* $\phi = 1 \vee 2 \vee 3 \vee 4 \vee 5$ *are depicted as a bright region in Fig. 5.3(c).*

Fig. 5.1(c) (where a 4-phase model is used) and Fig. 5.2(c) (where a 5-phase model is used), we observe only small changes in the segmented phases, except from the extra nonempty phase $\phi = 3$ in Fig. 5.2(c). For a test like this, we can not control which phase that ends up empty.

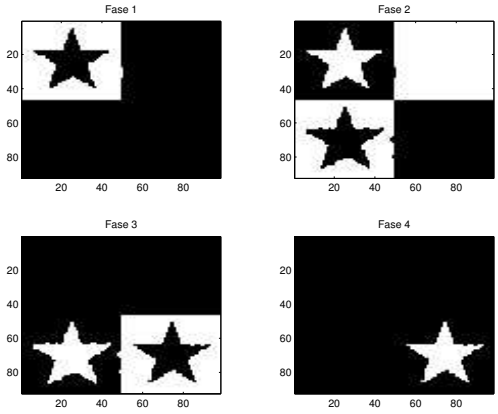
In the example below we use PCLSM to find a 5-phase segmentation. We use a noisy synthetic image containing 4 stars on 4 different backgrounds, as shown in Fig. 5.3(a). There are only 5 different phases in this image. The star in the upper left corner belongs to same phase as the background in the upper right corner. Further, the star in the upper right corner belongs to the same phase as the background in the lower left corner, and the star in this corner belongs to the same phase as the background in the lower right corner. In total that gives us 5 different phases. Almost no misclassification was observed in this tests. By using one level set function, we manage to find all 5 objects, as seen from Fig. 5.3(c). This is in contrast to classi-



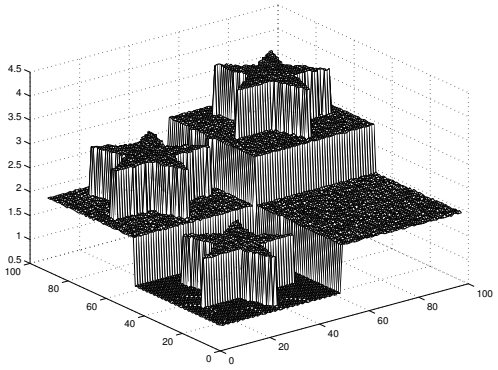
(a) Observed image u_0 (SNR$\approx$ 10.6).



(b) Initial level set function ϕ.



(c) Different phases using PCLSM.



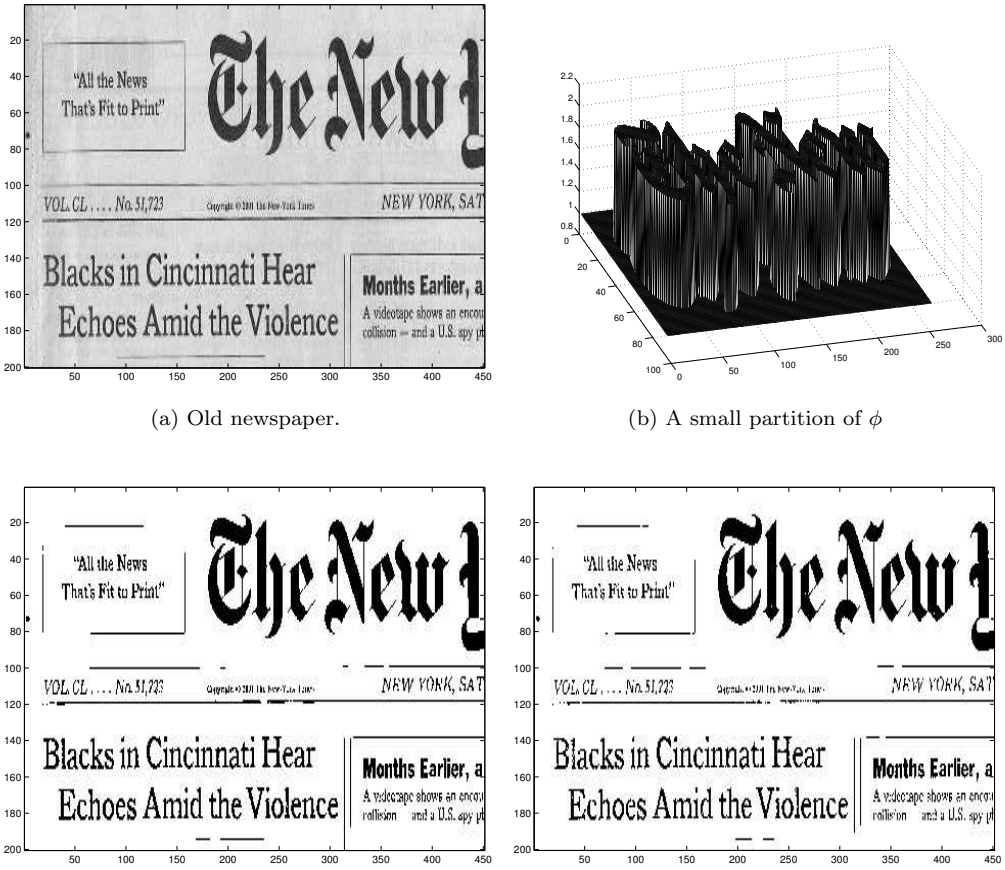
(d) At convergence ϕ approaches 4 constant values.

FIG. 5.4. *Each separate phase* $\phi = 1 \vee 2 \vee 3 \vee 4$ *are depicted as a bright region in Fig. 5.4(c).*

cal level set methods where 3 level set functions must be utilized to get the same result.

If an image actually consists of 5 objects, but a 4-phase model is utilized in the segmentation procedure, one may hope that 2 of the 5 objects merge into 1 new object without affecting the segmentation result for the other objects. Below we depict our result using Fig. 5.4(a) as the observation data in a 4-phase segmentation. By comparing Fig. 5.3(c) and Fig. 5.4(c) we see that phase $\phi = 2$ and phase $\phi = 3$ in Fig. 5.3(c) have merged into one phase $\phi = 2$ in Fig. 5.4(c). All other phases seem to be unaffected of this.

All the examples above indicate that our algorithm is an interesting alternative to the multiphase algorithm [5, 16] where standard level set formulation is utilized. We have shown that our algorithm is robust with respect to noise. As in [5, 16] we have shown that the exact number of phases do not have to be known a priori. Redundant phases can disappear in the sense that they will end up empty. Below we proceed with



(a) Old newspaper.

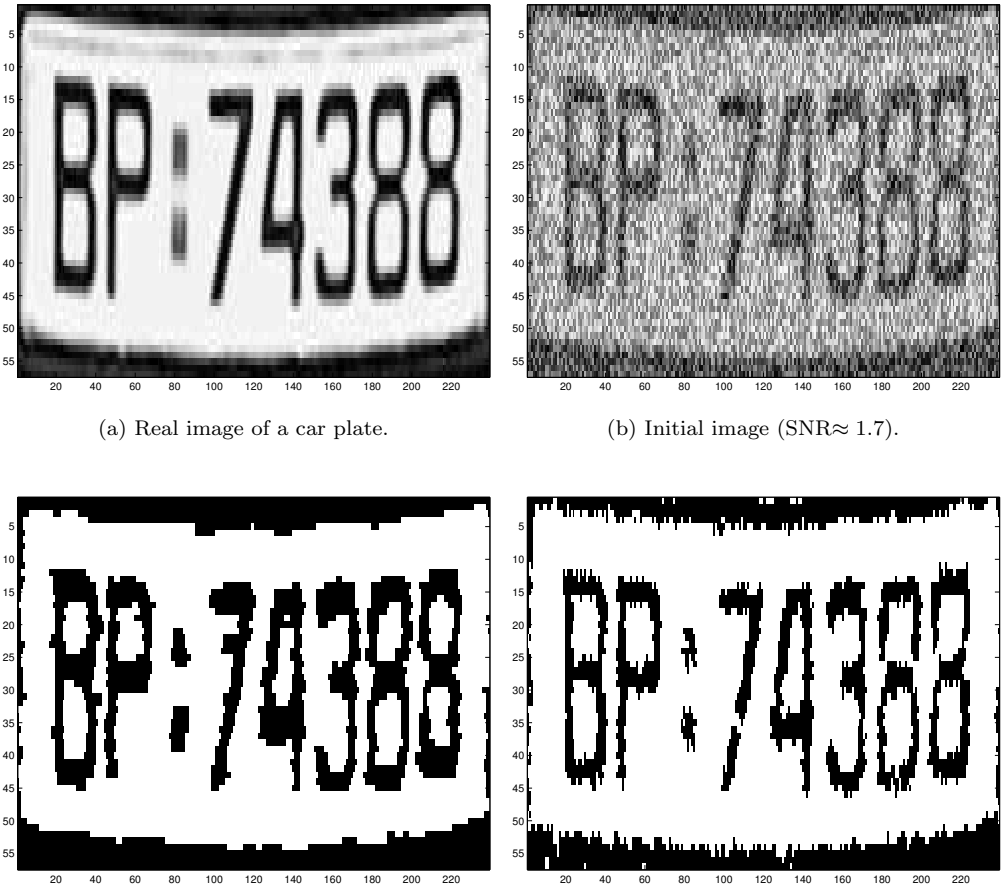
(b) A small partition of ϕ

(c) Result using PCLSM.

(d) Result using CVM.

FIG. 5.5. *For the purpose of illustration only a small partition of the level set function ϕ is depicted in the upper right corner. We see that ϕ takes two constant values, as it should.*

some examples using real pictures (some natural scene and one MR image). First we demonstrate that PCLSM can be use to extract characters or numbers form images. We begin with an image of an old newspaper where only two phases are needed; one phase to represent the characters and one phase to represent the remaining. To evaluate the segmentation process, the Chan/Vese method [5, 16], for short (CVM), is examined using the same input image. The result achieved with the segmentation techniques PCLSM and CVM are shown in Fig. 5.5(c) and Fig. 5.5(d) respectively. Few differences between the segmented images were observed, but a careful inspection of all characters in Fig. 5.5(c) and Fig. 5.5(d) reveal that PCLSM preserve small details somewhat better than CVM. In this paper we will mainly focus on the segmentation process, but let us briefly describe a way to identify the characters separated by phase $\phi = 2$ given in Fig. 5.5(b). First one can isolate each character in this phase by a *Peak-and-valley method* [13]. Thereafter, to identify the isolated characters, the center of mass for each character is aligned to the center of mass for all character templates in a database. Thereafter each isolated character is registered to all tem-



(a) Real image of a car plate.

(b) Initial image (SNR$\approx$ 1.7).

(c) Segmented with PCLSM.

(d) Segmented with CVM.

FIG. 5.6. *Character and number segmentation from a car plate.*

plates. Finally the difference between each registered character and all templates are measured. The template corresponding to the smallest difference in L_2-norm is chosen to be the solution.

In the next example we illustrate a 2-phase segmentation on a real car plate image. Locating and reading car plates is a well known problem, and there are a number of commercial software available. Below we demonstrate that our level set method can be used in such an application. Since character and number recognitions are the main target in this example, we assume the location of the car plate to be known. A perfect segmentation was found with both PCLSM and CVM if Fig. 5.6(a) was used as the observation data. We challenge these two segmentation techniques by adding Gaussian distributed noise to the real image and use the polluted image in Fig. 5.6(b) as the observation data. With this amount of noise both PCLSM and CVM miss some details along the edges for the characters and numbers. By increasing the regularization term for CVM we obtain smoother edges but then each character or number

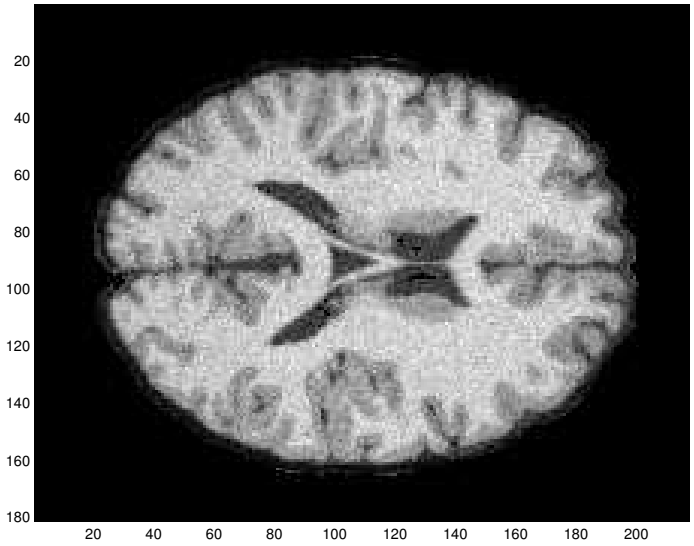
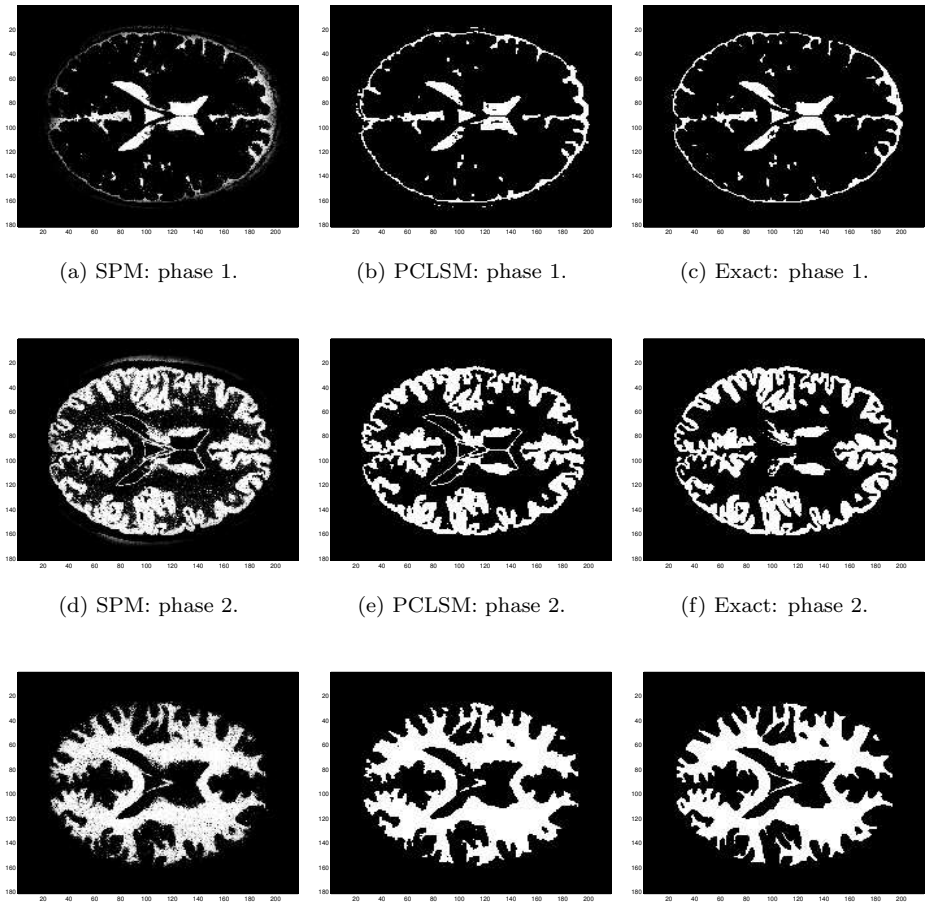


FIG. 5.7. *MRI image with a change in the intensity values going from left to right caused by the non-uniformity RF-puls.*

may be broken into several pieces. From Fig. 5.6(c) we see that each character and number appear as unbroken.

In our next example segmentation of a MR image is demonstrated. The image in Fig. 5.7 is available to the public at *http://www.bic.mni.mcgill.ca/brainweb/*. These realistic MRI data are used by the neuroimaging community to evaluate the performance of various image analysis methods in a setting where the truth is known. For the image used in this test the noise level is 7% and the non-uniformity intensity level of the RF-puls is 20%, see *http://www.bic.mni.mcgill.ca/brainweb/* for details concerning the noise level percent and the intensity level of the RF-puls. Both PCLSM and SPM [15] are used to segment the above MRI phantom and the results are depicted in Fig. 5.8. SPM is a statistical segmentation software program to process MRI data for research and clinical use. In Fig. 5.8 there are three tissue classes that should be identified; phase 1: cerebrospinal fluid, phase 2: gray matter, phase 3: white matter. Because of this, four phases were used, but we do not depict the background phase here. The main difference between PCLSM and SPM is their ability to handle noisy data. SPM is adjusted for MR data, but each phase obtained with SPM is perforated with misclassified pixels. The same problem do not occur with PCLSM.

As a final example we consider a scattered data problem. Here the goal is to connect single observation points and form continuous objects. The lights in a satellite image of Europe by night act as scattered data points. The intensity from each light source are the same, but the density of data points fluctuate within the satellite image. Generally it is hard to find features by connecting single observation points. Models based on edge stopping functions will fail completely for applications like this. We do not reckon our model as well suited for scattered data problems, but by following the proposition from remark 1 in section 3 we managed to get the result in Fig. 5.9(c). This result is comparable with what Chan and Vese report in [2] for the same satellite image. In Fig. 5.9(b)-(d) we demonstrate the effect of the total variation norm for ψ_i



(a) SPM: phase 1. (b) PCLSM: phase 1. (c) Exact: phase 1.

(d) SPM: phase 2. (e) PCLSM: phase 2. (f) Exact: phase 2.

(g) SPM: phase 3. (h) PCLSM: phase 3. (i) Exact: phase 3.

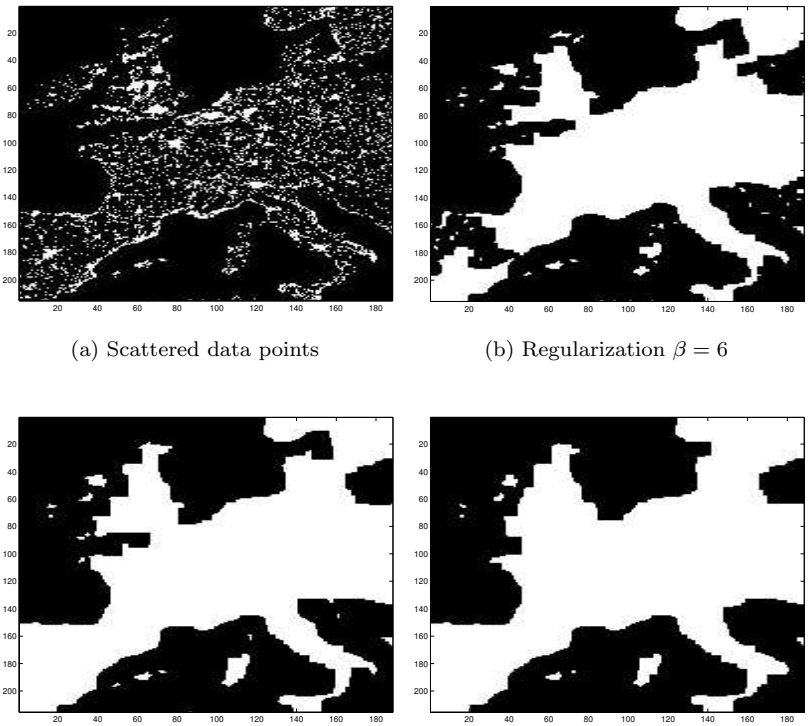
FIG. 5.8. *SPM is adjusted for MR data, but it seems hard for SPM to handle to noise properly. All phases found by PCLSM match the exact phases in a better way than SPM do.*

in (2.11). Weighting these terms too softly causes an image with perforated regions as seen in Fig. 5.9(b). On the other hand, if these terms are weighted too hard all sparse regions disappear, as seen in Fig. 5.9(d).

Acknowledgment. The authors thank Stanley Osher for valuable comments on a draft of this paper and for bring our attention to references [9, 6] before the submission of this work.

REFERENCES

[1] E. L. C. AND G. R. F., *Measure therory and fine properties of functions*, 1992.
[2] T. CHAN AND L. A. VESE, *Active contours without edges*, IEEE Image Proc., 10 (2001), pp. 266–277.
[3] T. F. CHAN AND X.-C. TAI, *Identification of discontinuous coefficients in elliptic problems using total variation regularization*, SIAM J. Sci. Comput., (Accepted and to appear). Available online athttp://www.mi.uib.no/~tai/.



(a) Scattered data points (b) Regularization $\beta = 6$

(c) Regularization $\beta = 7$ (d) Regularization $\beta = 9$

FIG. 5.9. *Scattered data are connected to form an object.*

[4] T. F. CHAN AND X.-C. TAI, *Level set and total variation regularization for elliptic inverse problems with discontinuous coefficients*, J. Comput. Physics, (to appear).

[5] T. F. CHAN AND L. A. VESE, *Image segmentation using level sets and the piecewise constant mumford-shah model*, tech. rep., CAM Report 00-14, UCLA, Math. Depart., April 2000. revised December 2000.

[6] S. CHEN, B. MERRIMAN, M. KANG, R. E. CAFILSCH, C. RATSCH, L.-T. CHENG, M. GYURE, R. P. FEDKIW, AND S. OSHER, *Level set method for thin film epitaxial growth*, tech. rep., CAM 00-03, UCLA Math. Depart., 2000.

[7] E. HODNELAND, *Segmentation of digital images*, cand. scient thesis, Department of Mathematics, University of Bergen, 2003. Available online at http://www.mi.uib.no/%7Etai.

[8] K. KUNISCH AND X.-C. TAI, *Sequential and parallel splitting methods for bilinear control problems in Hilbert spaces*, SIAM J. Numer. Anal., 34 (1997), pp. 91–118.

[9] B. MERRIMAN, R. CAFLISCH, S. OSHER, C. RATSCH, S. CHEN, M. KANG, AND M. GYURE, *Island dynamics and level set methods for continuum modeling of epitaxial growth*, in Applied and industrial mathematics, Venice–2, 1998, Kluwer Acad. Publ., Dordrecht, 2000, pp. 145–171.

[10] D. MUMFORD AND J. SHAH, *Optimal approximation by piecewise smooth functions and assosiated variational problems*, Comm. Pure Appl. Math, 42 (1989), pp. 577–685.

[11] S. OSHER AND J. A. SETHIAN, *Fronts propagating with curvature dependent speed: Algorithms based on hamilton-jacobi formulations*, J. Comput. Phys., 79 (1988), pp. 12–49.

[12] A. G. SAMSON C., BLANC-FÉRAUD L. AND Z. J., *A level set model for image classification*, IJCV, 40 (2000), pp. 187–197.

[13] C. J. SETCHELL, *Applications of computer vision to road-traffic monitoring*, phd thesis, Department of Computer Science, University of Bristol, 1997. Available online at http://www.cs.bris.ac.uk/Tools/Reports/Abstracts/1997-setchell.html.

[14] B. SONG AND T. CHAN, *A fast algorithm for level set based optimization*, tech. rep., CAM 0268, UCLA Math. Depart., 2002.

[15] SPM, *http://afni.nimh.nih.gov/afni/*.
[16] L. A. Vese and T. F. Chan, *A new multiphase level set framework for image segmentation via the Mumford and Shah model*, International Journal of Computer Vision, 50 (2002.), pp. 271–293.
[17] H.-K. Zhao, T. Chan, B. Merriman, and S. Osher, *A variational level set approach to multiphase motion*, J. Comput. Phys., 127 (1996), pp. 179–195.