INTERNATIONAL JOURNAL OF
NUMERICAL ANALYSIS AND MODELING
Volume 2, Number 4, Pages 422–438

A PIECEWISE CONSTANT LEVEL SET FRAMEWORK

JOHAN LIE, MARIUS LYSAKER, AND XUE-CHENG TAI

Abstract. In this work we discuss variants of a PDE based level set method. Traditionally interfaces are represented by the zero level set of continuous level set functions. We instead use piecewise constant level set functions, and let interfaces be represented by discontinuities. Some of the properties of the standard level set function are preserved in the proposed method. Using the methods for interface problems, we need to minimize a smooth convex functional under a constraint. The level set functions are discontinuous at convergence, but the minimization functional is smooth and locally convex. We show numerical results using the methods for segmentation of digital images.

Key Words. image segmentation, image processing, PDE, variational, level set, piecewise constant level set

1. Introduction

The level set method was proposed by Osher and Sethian in [1] as a versatile tool for tracing interfaces separating a domain Ω into subdomains. Interfaces are treated as the zero level set of higher dimensional functions. Moving the interfaces can implicitly be done by evolving level set functions instead of explicitly moving the interfaces. We give a brief introduction to the level set method in §2. For a recent survey on the level set methods see [2, 3, 4, 5]. Applications of the level set method include image analysis, reservoir simulation, inverse problems, computer vision and optimal shape design [6, 7, 8, 9]. In this work, we present variants of the level set method. The primary concern for our approach is to remove the connection between the level set functions and the signed distance function and thus remove some of the computational difficulties associated with the calculation of the Eikonal equation, see §2. Another motivation is to avoid the non-differentiability associated with the Heaviside and Delta functions used in some of the level set formulations [6, 10]. This will also turn the minimization functional into a locally convex and smooth functional. The third concern of this approach is to develop fast algorithms for level set methods. Due to the fact that the functional and the constraints for this approach are rather smooth, it is possible to apply Newton types of iterations to construct fast algorithms for the proposed model. One of the variants extends the level set models proposed in [11, 12] and it is also closely related to the phase-field methods [13, 14, 15, 16]. Our framework can be used for different applications where a domain should be divided into subdomains. In this work, we concentrate on image segmentation problems.

For a given digital image $u_0 : \Omega \to \mathbb{R}$, the aim is to separate Ω into a set of subdomains Ω_i such that $\Omega = \cup_{i=1}^{n}\Omega_i$ and u_0 is nearly a constant in each Ω_i. Having determined the partition of Ω into a set of subdomains Ω_i, one can do

2000 *Mathematics Subject Classification.* 49Q10, 35R30.
We acknowledge support from the Norwegian Research Council.

further modelling on each domain independently and automatically. One general image segmentation model was proposed by Mumford and Shah in [17]. Numerical approximations are thoroughly treated in [18]. Using this model, the image u_0 is decomposed into $\Omega = \cup_i \Omega_i \cup \Gamma$, where Γ is a curve separating the different domains. Inside each Ω_i, u_0 is approximated by a smooth function. The optimal partition of Ω is found by minimizing the Mumford-Shah functional (6). This is explained in §2. Following the Mumford-Shah formulation for image segmentation, Chan and Vese [6, 10] solved the minimization problem by using level set functions. The interface Γ is traced by the level set functions. Motivated by the Chan-Vese approach, we will in this article solve the segmentation problem in a different way, i.e. by introducing a piecewise constant level set function ϕ. Instead of using the zero level of a function to represent the interface between subdomains, we let the interface be represented implicitly by the discontinuities of a set of basis functions $\psi_i(\phi)$. In order to divide Ω into subdomains Ω_i, such that $\Omega = \cup_i \Omega_i$, we use a set of functions ψ_i satisfying $\psi_i = 1$ in Ω_i and $\psi_j = 0$ in Ω_i when $j \neq i$, see Figure 1.

The rest of this article is structured as follows. In §2 we give a brief review of the traditional level set method. Our general framework and the minimization functional used for image segmentation is formulated in §3. The segmentation problem is formulated as a minimization problem with a smooth cost functional under a constraint. We are essentially minimizing the Mumford-Shah functional associated with the new level set model. In §4 and §5 we explain our two variants of the level set method for image segmentation in more detail. Both sections include algorithms and numerical results. We conclude with a brief discussion. For a more detailed treatment of the two methods, including more numerical results we refer the reader to [19, 20].

2. Standard Level Set Methods

The main idea behind the level set formulation is to represent an interface $\Gamma(t)$ bounding a possibly multiply connected region in $\mathbb{R}^n$ by a Lipschitz continuous function ϕ, having the following properties

$$\begin{cases} \phi(x,t) > 0, & \text{if x is inside } \Gamma, \\ \phi(x,t) = 0, & \text{if x is at } \Gamma, \\ \phi(x,t) < 0, & \text{if x is outside } \Gamma. \end{cases} \tag{1}$$

Some regularity must be imposed on ϕ to prevent the level set function of being too steep or too flat near the interface. This is normally done by requiring ϕ to be a signed distance function to the interface

$$\begin{cases} \phi(x,t) = d(\Gamma,x), & \text{if x is inside } \Gamma, \\ \phi(x,t) = 0, & \text{if x is at } \Gamma, \\ \phi(x,t) = -d(\Gamma,x), & \text{if x is outside } \Gamma, \end{cases} \tag{2}$$

where $d(\Gamma, x)$ denotes Euclidean distance between x and Γ. Having defined the level set function ϕ as in (2), there is a one to one correspondence between the curve Γ and the function ϕ. The distance function ϕ obeys the Eikonal equation

$$|\nabla\phi| = 1. \tag{3}$$

The solution of (3) is not unique in the distributional sense. Finding the unique vanishing viscosity solution of (3) is usually done by solving the following initial value problem to steady state

$$\phi_t + sgn(\tilde{\phi})(|\nabla\phi| - 1) = 0 \tag{4}$$

$$\phi(x,0) = \tilde{\phi}(x). \tag{5}$$

In the above, $\tilde{\phi}$ may not be a distance function. When the steady state of equation (4) is reached, it will be a distance function having the same zero curve as $\tilde{\phi}$. This is commonly known as the reinitialization procedure. For numerical computations this procedure is crucial. Some finite difference schemes exists, see [2, 3, 9] for some details.

2.1. Level Set Methods and Image Segmentation. The active contour (snake) model evolves a curve $\Gamma(t)$ in order to detect objects in an image u_0 [21]. The curve is moved from an initial position $\Gamma(0)$ in the direction normal to the curve, subject to constraints in the image. An edge detector $g(\nabla u_0)$ determines when $\Gamma(t)$ is situated at the boundary of an object. One limitation of the snake model is that the curve is represented explicitly, thus topological changes like merging and breaking of the curve may be hard to handle. To address this problem, a level set formulation of the active contour model was introduced in [22]. Later, Chan-Vese introduced a level set model for active contour segmentation, with the very important property that the stopping criteria is independent of ∇u_0 [6]. This means that boundaries not defined by gradients can be detected. Instead, the evolvement of the curve is based on the general Mumford-Shah formulation of image segmentation, by minimizing the functional

$$F^{MS}(u,\Gamma) = \sum_i \int_{\Omega_i} |c_i - u_0|^2 dx + \mu|\Gamma| + \nu \int_{\Omega\backslash\Gamma} |\nabla u|^2 dx. \tag{6}$$

In the above, $|\Gamma|$ is the length of Γ. A minimizer of this functional is smooth in $\Omega\backslash\Gamma$. The piecewise constant Mumford-Shah formulation of image segmentation is to find a partition of Ω such that u in Ω_i equals a constant c_i, and $\Omega = \cup_i^n \Omega_i \cup \Gamma$. The two last terms in (6) are regularizers measuring curve-length of the curves bounding the phases, and smoothness of u in $\Omega\backslash\Gamma$. Based on (6), Chan and Vese [6] proposed the following minimization problem for a two-phase segmentation

$$\min_{c_1,\, c_2,\, \phi} \Big\{ \int_\Omega |u_0 - c_1|^2 H(\phi) dx + \int_\Omega |u_0 - c_2|^2 (1 - H(\phi)) dx + \nu \int_\Omega H(\phi) dx + \mu \int_\Omega \delta(\phi)|\nabla\phi| dx \Big\}. \tag{7}$$

Here ϕ is the level set function satisfying (1) and $H(\phi)$ is the Heaviside function

$$H(\phi) = \begin{cases} 1, & \phi \geq 0, \\ 0, & \phi < 0. \end{cases} \tag{8}$$

Finding a minimum of (7) is done by introducing an artificial time variable, and moving ϕ in the steepest descent direction to steady state

$$\phi_t = \delta_\epsilon(\phi) \left(-(u_0 - c_1)^2 + (u_0 - c_2)^2 - \nu + \mu \nabla \cdot \left(\frac{\nabla\phi}{|\nabla\phi|} \right) \right), \tag{9}$$

$$\phi(\boldsymbol{x}, 0) = \phi_0(\boldsymbol{x}). \tag{10}$$

Here δ_ϵ is a globally positive approximation to the δ function, see [6]. The recovered image is a piecewise constant approximation to u_0.

If we do not impose any other conditions on the level set functions ϕ, then the minimizer of F^{MS} with respect to ϕ may not be unique. Thus, we require the level set function ϕ to be a distance function. This means that the level set function is a steady state of both (4) and (9). In practice, this means we need to reinitialize the level set function.

This level set framework was later generalized to multiple phase segmentation using multiple level set functions [10]. A four phase segmentation can be accomplished by minimizing the functional

$$
\begin{aligned}
F(\boldsymbol{c},\phi_1,\phi_2)=&\int_\Omega |c_1-u_0|^2 H(\phi_1)H(\phi_2)dx+\int_\Omega |c_2-u_0|^2 H(\phi_1)(1-H(\phi_2))dx\\
&+\int_\Omega |c_3-u_0|^2(1-H(\phi_1))H(\phi_2)dx+\int_\Omega |c_4-u_0|^2(1-H(\phi_1))(1-H(\phi_2))dx\\
&+\nu\int_\Omega H(\phi_1)dx+\nu\int_\Omega H(\phi_2)dx+\mu\int_\Omega \delta(\phi_1)|\nabla\phi_1|dx+\mu\int_\Omega \delta(\phi_2)|\nabla\phi_2|dx.
\end{aligned}
\tag{11}
$$

Having determined $\boldsymbol{c}=\{c_i\}_{i=1}^4$, ϕ_1 and ϕ_2 by the minimization of (11), four different regions can be identified by the sign of the two level set functions such that

$$
u(\boldsymbol{x})=\begin{cases} c_1, & \text{if } \phi_1(\boldsymbol{x})>0\ ,\ \phi_2(\boldsymbol{x})>0,\\ c_2, & \text{if } \phi_1(\boldsymbol{x})>0\ ,\ \phi_2(\boldsymbol{x})<0,\\ c_3, & \text{if } \phi_1(\boldsymbol{x})<0\ ,\ \phi_2(\boldsymbol{x})>0,\\ c_4, & \text{if } \phi_1(\boldsymbol{x})<0\ ,\ \phi_2(\boldsymbol{x})<0.\end{cases}
\tag{12}
$$

By utilizing (8), the recovered cartoon image u consisting of four phases can be written as

$$
\begin{aligned}
u\ =&\ c_1H(\phi_1)H(\phi_2)+c_2H(\phi_1)(1-H(\phi_2))\\
+&\ c_3(1-H(\phi_1))H(\phi_2)+c_4(1-H(\phi_1))(1-H(\phi_2)).
\end{aligned}
\tag{13}
$$

Increasing the number of phases is done by increasing the number of level set functions. With the use of N level set functions it is possible to represent up to 2^N phases.

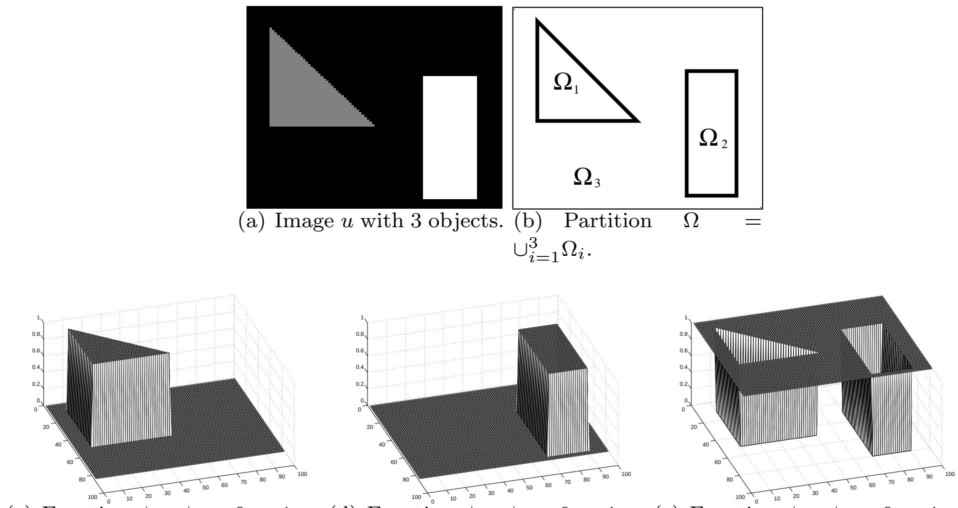
In this work we solve the piecewise constant Mumford-Shah segmentation using a slightly different level set approach. We separate the connection between the level set function and the distance function. This means that we get rid of the reinitialization procedure. Our approach is truly variational, i.e. the equations we need to solve are coming from the Euler-Lagrange equations for some smooth convex functions. The problem of point-wise non-differentiability of the Heaviside and Delta functions is avoided and the cost functional is also convex.

3. A General Framework

In this section a general framework for representing subdomains of Ω is developed. To each subdomain Ω_i corresponds a basis function ψ_i, such that $\psi_i=1$ in Ω_i and zero elsewhere. Two different realizations of the basis functions ψ_i are developed in §4 and §5. The basis functions are constructed using one or several level set functions $\{\phi_j\}_{j=1}^l$. As mentioned in the introduction, each ψ_i is compactly supported in Ω_i. Using this property, we can construct a piecewise constant function u by a weighted sum of the basis functions. If we let $\boldsymbol{c}=\{c_i\}_{i=1}^n$ be a set of real scalars, we can represent a piecewise constant function u taking these n distinct constant values by

$$
u=\sum_{i=1}^n c_i\psi_i.
\tag{14}
$$

In Figure 1 we demonstrate the relationship between an image function u, the partition of Ω associated with this image, and the corresponding basis functions ψ_i, $i=1,2,3$.



(a) Image u with 3 objects. (b) Partition $\Omega = \cup_{i=1}^{3}\Omega_i$.

(c) Function ψ_1. $\psi_1 = 0$ or 1. (d) Function ψ_2. $\psi_2 = 0$ or 1. (e) Function ψ_3. $\psi_3 = 0$ or 1.

Figure 1: The relationship between an image function $u \in BV(\Omega)$, the partition of Ω associated with the image, and the corresponding basis functions ψ_i, $i = 1, 2, 3$.

A function u_0 being almost equal to a constant in n subdomains can thus be approximated by (14) provided we optimally choose $\boldsymbol{c}$ and $\{\psi_i\}_{i=1}^n$. This is done by solving a minimization problem subject to a constraint corresponding to the choice of basis functions. The constraint controls the structure of possible solutions. We will not go into details concerning $K(\phi)$ here, but instead return to that issue in §4 and §5.

The simple structure of the basis functions gives us the opportunity to measure the lengths of curves surrounding Ω_i and the area of each region Ω_i by

$$|\partial\Omega_i| = \int_\Omega |\nabla\psi_i| dx, \qquad \text{and} \qquad |\Omega_i| = \int_\Omega \psi_i dx. \tag{15}$$

Here we note that $|\partial\Omega_i|$ is the Total Variation (TV)-norm of ψ_i [23].

The above framework can be used as a tool for image segmentation. Let u_0 be an image to be segmented, where u_0 might contain noise. We want to construct a piecewise constant function u which approximates u_0 in a proper sense. The segmentation can be formulated as a minimization of the following functional

$$F(\phi, \boldsymbol{c}) = \frac{1}{2}\int_\Omega |\, u - u_0\,|^{\,2} dx + \beta \sum_{i=1}^{n} \int_\Omega |\nabla\psi_i| dx, \tag{16}$$

where u is related to $\mathbf{c}$ and u as in (14). The first term of F is a least squares fidelity term, measuring the closeness of u to u_0. The second term measures the length of all the curves. To be able to pick out different subdomains in the image in an automatic way, we also impose a constraint, i.e. we try to solve the constrained minimization problem

$$\min_{\boldsymbol{c},\phi} \; F(\boldsymbol{c}, \phi) \quad \text{subject to} \quad K(\phi) = 0. \tag{17}$$

This problem can be solved by using the augmented Lagrangian method of optimization [24, 25]. A minimizer of F corresponds to a saddle-point of the augmented Lagrangian functional

$$L(\mathbf{c},\phi,\lambda) = F(\mathbf{c},\phi) + \int_\Omega \lambda K(\phi)\, dx + \frac{r}{2}\int_\Omega |K(\phi)|^2 dx, \tag{18}$$

where $r > 0$ is a penalty parameter which needs to be chosen properly, and λ is a function defined on the same domain as ϕ called the Lagrangian multiplier. At a saddle-point of (18) we must have

$$\frac{\partial L}{\partial \phi} = 0, \quad \frac{\partial L}{\partial c_i} = 0 \text{ and } \frac{\partial L}{\partial \lambda} = 0. \tag{19}$$

Essentially we minimize L w.r.t $\boldsymbol{c}$ and ϕ, and maximize L w.r.t λ. In §4 and §5 we introduce iterative algorithms to find the saddle-points in (19) coming from two different level set formulations. However, below we go through some calculations that both these two formulations have in common.

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 w.r.t $\boldsymbol{c}$ can be solved exactly. Note that

$$\frac{\partial L}{\partial c_i} = \int_\Omega \frac{\partial L}{\partial u}\frac{\partial u}{\partial c_i} = \int_\Omega (u - u_0)\psi_i\, dx, \qquad \text{for } i = 1, 2, \dots n. \tag{20}$$

Therefore, the minimizer satisfies a linear system of equations $A\mathbf{c}^k = b$ in the following form:

$$\sum_{j=1}^n \int_\Omega (\psi_j \psi_i) c_i^k\, dx = \int_\Omega u_0 \psi_i\, dx, \qquad \text{for } i = 1, 2, \dots n. \tag{21}$$

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. We form the matrix A and vector b and solve the equation $A\mathbf{c}^k = b$ using an exact solver. The minimization with respect to ϕ will be solved by the following gradient method:

$$\phi^{new} = \phi^{old} - \Delta t \frac{\partial L}{\partial \phi}(\mathbf{c}, \phi^{old}, \lambda), \tag{22}$$

where Δt is a small positive number called the time-step. For a given $\boldsymbol{c}$ and λ, we need to iterate many times in order to find the minimizer with respect to ϕ. In our simulations, we mostly just do a fixed number of iterations or stop the iteration after the norm of gradient $\partial L/\partial \phi$ has been reduced by a given factor. This is the most time-consuming part of the algorithms. Therefore we are currently working on the issue of accelerating the convergence by using a Newton-type of iteration.

4. Piecewise Constant Level Set Method with the Polynomial Approach

We shall first present the piecewise constant level set method (PCLSM). Assume that we need to find n regions $\{\Omega_i\}_{i=1}^n$ which form a portion of Ω. In order to find the regions, we want to find a piecewise constant function which takes values

$$\phi = i \text{ in } \Omega_i, \quad i = 1, 2, \dots, n. \tag{23}$$

With this approach we just need one function to identify all the phases in Ω. The basis functions ψ_i associated with ϕ are defined in the following form:

$$\psi_i = \frac{1}{\alpha_i} \prod_{\substack{j=1 \\ j\neq i}}^{n} (\phi - j) \quad \text{and} \quad \alpha_i = \prod_{\substack{k=1 \\ k\neq i}}^{n} (i-k). \tag{24}$$

It is clear that the function u given by (14) is a piecewise constant function and $u = c_i$ in Ω_i if ϕ is as given in (23). The function u is a polynomial of order $n-1$ in ϕ. 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 (23) holds.

To ensure that equation (14) gives us a unique representation of u, i.e. at convergence different values of ϕ should correspond to different function values $u(\phi)$ in (14), we introduce

$$K(\phi) = (\phi-1)(\phi-2)\cdots(\phi-n) = \prod_{i=1}^{n}(\phi - i). \tag{25}$$

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

$$K(\phi) = 0, \tag{26}$$

there exists a unique $i \in \{1, 2, \ldots, n\}$ 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 $K(\phi) = 0$. The constraint (26) is used to guarantee that there is no vacuum and overlap between the different phases. In [26] some other constraints for the classical level set methods were used to avoid vacuum and overlap.

Following the framework in §3, we will use the basis functions (24), the constraint (25) and the representation (14) of u. To find a minimizer for (18), we need to find the saddle point which satisfies $\frac{\partial L}{\partial \phi} = 0$, $\frac{\partial L}{\partial \mathbf{c}} = 0$ and $\frac{\partial L}{\partial \lambda} = 0$. Remember that $\frac{\partial L}{\partial c_i}$ is zero if $\{c_i\}_{i=1}^n$ are computed from (21). We use the Uzawa-type Algorithm 1 to find a saddle point for $L(\mathbf{c}, \phi, \lambda)$. This algorithm has a linear convergence rate and its convergence has been analyzed by Kunisch and Tai [27] under a slightly different context. The algorithm has also been used by Chan and Tai [7, 28] for a level set method for elliptic inverse problems.

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

- Find $\mathbf{c}^k$ from

$$L(\mathbf{c}^k, \phi^{k-1}, \lambda^{k-1}) = \min_{\mathbf{c}} L(\mathbf{c}, \phi^{k-1}, \lambda^{k-1}). \tag{27}$$

- Use (14) to update $u = \sum_{i=1}^n c_i^k \psi_i(\phi^{k-1})$.
- Find ϕ^k from

$$L(\mathbf{c}^k, \phi^k, \lambda^{k-1}) = \min_{\phi} L(\mathbf{c}^k, \phi, \lambda^{k-1}). \tag{28}$$

- Use (14) to update $u = \sum_{i=1}^n c_i^k \psi_i(\phi^k)$.
- Update the Lagrange-multiplier by

$$\lambda^k = \lambda^{k-1} + rK(\phi^k). \tag{29}$$

- If not converged: Set k=k+1 and go to step 1.

To compute $\frac{dL}{d\phi}$ we utilize the chain rule to get

$$\frac{\partial L}{\partial \phi} = (u - u_0)\frac{\partial u}{\partial \phi} - \beta \sum_{i=1}^{n} \nabla\cdot\Big(\frac{\nabla \psi_i}{|\nabla \psi_i|}\Big)\frac{\partial \psi_i}{\partial \phi} + \lambda \frac{\partial K}{\partial \phi} + rK\frac{\partial K}{\partial \phi}. \tag{30}$$

It is easy to get $\partial u/\partial\phi$, $\partial\psi_i/\partial\phi$ and $\partial K/\partial\phi$ from (14), (24) and (25). We use the gradient method (22) to solve (28). We do a fixed number of iterations, for example 400 iterations or stop the iteration after the L_2 norm of gradient has been reduced by 10%.

Remark 1. *The updating for the constant values in (27) is very ill-posed. A small perturbation of the ϕ function produces a large perturbation for the c_i values. Due to this reason, we have tried out a variant of Algorithm 1. In each iteration we alternate between (28) and (29), while (27) 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 (27) 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 (22) and Algorithm 1 is still convergent. This strategy is particular efficient when the amount of noise 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* **c** *are available, an alternative variant of Algorithm 1 may be used, i.e. we first minimize with the level set function followed by a minimization for the constant values and then update the multiplier.*

4.1. Numerical Experiments with the Polynomial Approach. In this section we validate the piecewise constant level set method with numerical examples. We consider only two-dimensional cases and restrict ourself to gray-scale images, but the model can handle any dimension and can be extended to vector-valued images as well. Our results will be compared with the related works [6, 10]. The original image is known for the 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 the background) as shown in Figure 2(a). This is the same image as Chan and Vese used to examine their multiphase algorithm [6, 10].

A careful evaluation of our algorithm is reported below. The observation data u_0 is given in Figure 2(a) and the only assumption we make is that a 4-phase model should be utilized to find the segmentation. In Figure 2(d) the ϕ function is depicted at convergence. The function ϕ approaches the predetermined constants $\phi = 1 \vee 2 \vee 3 \vee 4$. Each of these constants represents one unique phase as seen in Figure 2(c). Our result is in accordance with what Chan and Vese reported in [6, 10].

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 is 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 again use Figure 2(a) as the observation image and utilize a 5-phase model. Our results are reported Figure 3. One of the 5 phases must

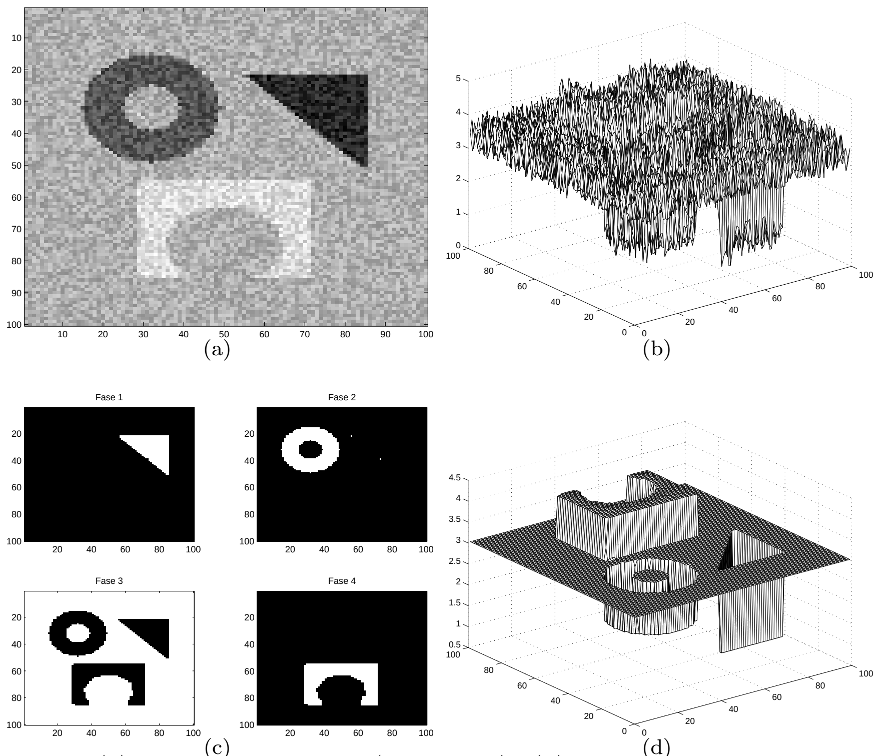


Figure 2: (a) Observed image u_0 (SNR$\approx$ 5.2). (b) Initial level set function ϕ. (c) Each separate phase $\phi = 1 \vee 2 \vee 3 \vee 4$ are depicted as a bright region. (d) At convergence ϕ approaches 4 constant values.

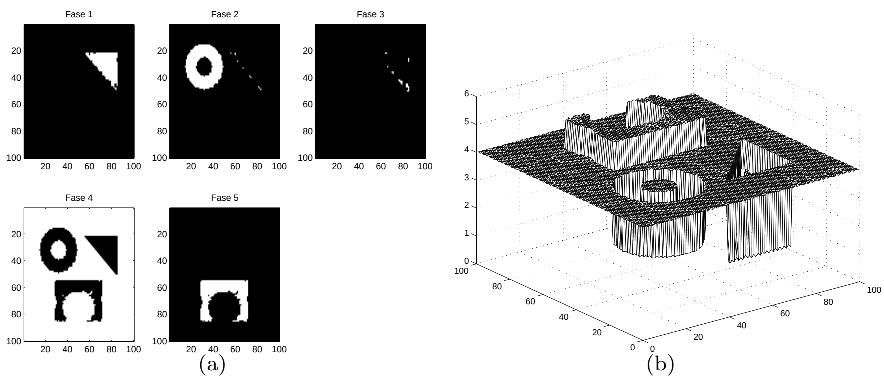
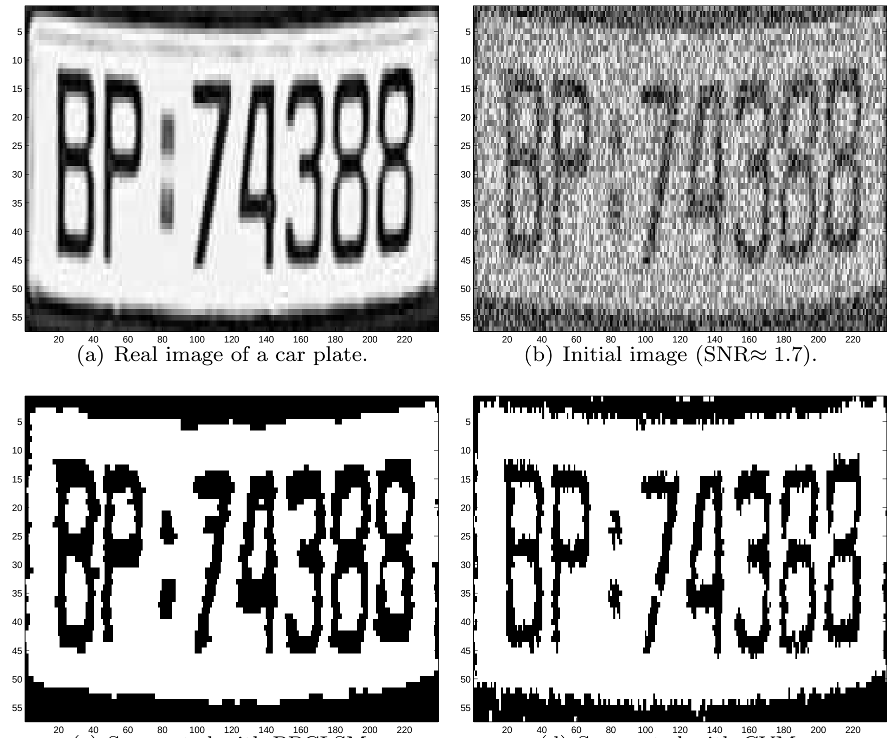


Figure 3: (a) Each separate phase $\phi = 1 \vee 2 \vee 3 \vee 4 \vee 5$ are depicted as a bright region. (b) At convergence ϕ approaches 4 constant values.

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 Figure 3(b) approaches the



(a) Real image of a car plate. (b) Initial image (SNR≈ 1.7).

(c) Segmented with PPCLSM. (d) Segmented with CVM.

Figure 4: Character and number segmentation from a car plate.

constants $\phi = 1 \vee 2 \vee 4 \vee 5$, except from the few misclassified pixels where $\phi = 3$ as seen in Figure 3(a). By comparing Figure 2(c) (where a 4-phase model is used) and Figure 3(a) (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 Figure 3(a). For a test like this, we can not control which phase that ends up empty.

Both the examples above indicate that our algorithm is an interesting alternative to the multiphase algorithm [6, 10] where standard level set formulation is utilized. We have shown that our algorithm is robust with respect to noise.

Below we proceed with one example using a real picture. We want to demonstrate that PPCLSM (polynomial PCLSM) can be use to extract characters or numbers from images. We use an image of a car plate 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 [6, 10], for short (CVM), is examined using the same input image. A perfect segmentation was found with both PPCLSM and CVM if no noise is added. We challenge these two segmentation techniques by adding Gaussian distributed noise to the real image and use the polluted image in Figure 4(b) as the observation data. With this amount of noise both PPCLSM 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 may be broken into several pieces. From Figure 4(c) we see that each character and number appear as unbroken.

5. The Binary Approach for PCLSM

We will now introduce an alternative realization of the basis functions in (14). Using the following approach, we can represent a maximum of 2^N subdomains, using N level set functions $\{\phi_i\}_{i=1}^N$. To simplify notation, we form the vector $\boldsymbol{\phi} = \{\phi_1, \phi_2, \ldots, \phi_N\}$. To introduce the binary level set idea, let us first assume that the interface Γ is enclosing $\Omega_1 \subset \Omega$. By standard level set methods the interior of Ω_1 is represented by points $\boldsymbol{x} : \phi(\boldsymbol{x}) > 0$, and the exterior of Ω_1 is represented by points $\boldsymbol{x} : \phi(\boldsymbol{x}) < 0$, as in (1). We instead let

$$\phi(\boldsymbol{x}) = 1 \text{ if } \boldsymbol{x} \in \text{interior of} \Omega_1, \qquad \phi(\boldsymbol{x}) = -1 \text{ if } \boldsymbol{x} \in \text{ exterior of } \Omega_1$$

. As proposed, Γ is implicitly defined as the discontinuity of ϕ. Representing four subdomains is done in an analogous way as (12), by

$$u(\boldsymbol{x}) = \begin{cases} c_1, & \text{if } \phi_1(\boldsymbol{x}) = 1, \quad \phi_2(\boldsymbol{x}) = 1, \\ c_2, & \text{if } \phi_1(\boldsymbol{x}) = 1, \quad \phi_2(\boldsymbol{x}) = -1, \\ c_3, & \text{if } \phi_1(\boldsymbol{x}) = -1, \quad \phi_2(\boldsymbol{x}) = 1, \\ c_4, & \text{if } \phi_1(\boldsymbol{x}) = -1, \quad \phi_2(\boldsymbol{x}) = -1. \end{cases}$$

Thus, a piecewise constant function taking four different constant values can be written

$$\begin{aligned} u = {} & \frac{c_1}{4}(\phi_1+1)(\phi_2+1) - \frac{c_2}{4}(\phi_1+1)(\phi_2-1) \\ & -\frac{c_3}{4}(\phi_1-1)(\phi_2+1) + \frac{c_4}{4}(\phi_1-1)(\phi_2-1) \end{aligned} \tag{31}$$

Using (31), we can form the set of basis functions ψ_i as in the following

$$u = c_1 \underbrace{\frac{1}{4}(\phi_1+1)(\phi_2+1)}_{\psi_1} + c_2 \underbrace{(-1)\frac{1}{4}(\phi_1+1)(\phi_2-1)}_{\psi_2} + \ldots, \tag{32}$$

and we can write: $u = \sum_{i=1}^4 c_i \psi_i$. This mechanism can also be easily generated to the case using more level set functions. For $i = 1, 2, \ldots, 2^N$, let $(b_1^{i-1}, b_2^{i-1}, \ldots, b_N^{i-1})$ be the binary representation of $i-1$, where $b_j^{i-1} = 0 \vee 1$. Let $s(i) = \sum_{j=1}^N b_j^{i-1}$, and write ψ_i and u as

$$\psi_i = \tfrac{(-1)^{s(i)}}{2^N} \prod_{j=1}^N (\phi_j + 1 - 2b_j^{i-1}) \quad \text{and} \qquad u = \sum_{i=1}^{2^N} c_i \psi_i. \tag{33}$$

It is now easy to see that these basis functions have the properties needed for the framework in §3. Using this representation for the basis functions, we need N constraints, one constraint K_i to each of the level set functions ϕ_i. Let use define $K_i(\phi_i) = \phi_i^2 - 1 \; \forall\, i$. Setting $K_i(\phi_i) = 0$ implies ϕ_i can only take the values ± 1 at convergence.

Having determined the choice of basis functions $\{\psi_i\}_{i=1}^n$ and the representation of u by (33), we need to find the saddle point of L by the augmented Lagrangian method. This means that we must minimize L w.r.t $\boldsymbol{\phi}$ and $\boldsymbol{c}$, and maximize L w.r.t $\boldsymbol{\lambda}$, which is of the same dimension as $\boldsymbol{\phi}$.

We minimize L w.r.t $\boldsymbol{\phi}$ by using the gradient method (22) for all the N level set functions. The gradients for the level set functions are given as:

$$\frac{\partial L}{\partial \phi_i} = (u - u_0) \sum_{j=1}^{2^N} c_j \frac{\partial \psi_j}{\partial \phi_i} - \beta \sum_{j=1}^{2^N} \nabla \cdot \Big(\frac{\nabla \psi_j}{|\nabla \psi_j|} \Big) \frac{\partial \psi_j}{\partial \phi_i} + 2\lambda_i \phi_i + 2r(\phi_i^2 - 1)\phi_i. \tag{34}$$

The constraints K_i are independent of the constant values c_i and thus the same formula (21) can be used to update the c_i values.

Similar to the algorithm used for the polynomial approach for the PCLSM, we could use the following algorithm to find a saddle point for the binary approach for the PCLSM. Note that we may need more than one level set function in this approach and the constraint $\boldsymbol{K}$ also has a different meaning now.

Algorithm 2. Choose initial values for $\boldsymbol{\phi}^0$ and $\boldsymbol{\lambda}^0$. For $k = 1, 2, \ldots$, do:

- Update $\boldsymbol{\phi}^k$ by (22), to approximately solve

$$L(\boldsymbol{c}^{k-1}, \boldsymbol{\phi}^k, \boldsymbol{\lambda}^{k-1}) = \min_{\phi} L(\boldsymbol{c}^{k-1}, \boldsymbol{\phi}, \boldsymbol{\lambda}^{k-1}) \tag{35}$$

- Construct $u(\boldsymbol{c}^{k-1}, \boldsymbol{\phi}^k)$ by $u = \sum_{i=1}^{2^N} c_i^{k-1} \psi_i^k$.
- Update $\boldsymbol{c}^k$ by (21), to solve

$$L(\boldsymbol{c}^k, \boldsymbol{\phi}^k, \boldsymbol{\lambda}^{k-1}) = \min_{\boldsymbol{c}} L(\boldsymbol{c}, \boldsymbol{\phi}^k, \boldsymbol{\lambda}^{k-1}). \tag{36}$$

- Update the multiplier by

$$\boldsymbol{\lambda}^k = \boldsymbol{\lambda}^{k-1} + r\boldsymbol{K}(\boldsymbol{\phi}^k). \tag{37}$$

- If not converged: Set k=k+1 and go to step 1.

Remark 3. *In most of our simulations we have set r to be constant during the processing. This is done to make the simulations as "safe" as possible. Better convergence behavior can be expected if r is increased during the iterations, but be aware of ill-conditioning if r is increased too quickly. This is a common approach when using the augmented Lagrangian method. See* [24, 25] *for details concerning the general algorithm.*

Remark 4. *The minimization w.r.t* $\mathbf{c}$ *done in step* 3 *should not be done too early in the process, e.g. not before $|\phi_i| \approx 1$.*

If $|\phi_i|$ is far from 1, then ψ_i is far from orthogonal to ψ_j, and the inner product (ψ_i, ψ_j) in (21) will give contributions at points where it should not. This means that the matrix inversion in (21) *does not give a good approximation to* $\mathbf{c}$ *unless $|\phi_i| \approx 1$ $\forall i$. See Remark 1 of Algorithm 1.*

Remark 5. *In this algorithm, Δt in the gradient iteration depends on both β and r. Larger r or β requires a small Δt. A bigger β will suppress oscillations, while a bigger r makes the level set functions ϕ_i converge to ± 1 quicker. Choosing r too big will reduce the influence of the fitting term $F(\boldsymbol{\phi}, \mathbf{c})$ and thus may increase the iteration number needed to converge to the true solution. For practical problems, it is normally not too difficult to find a approximate range for these two parameters.*

5.1. Numerical Experiments with the Binary Approach. Now we will present some of the numerical results achieved using the binary piecewise constant level set formulation (BPCLSM). First of all, the image depicted in Figure 2 (a) has been processed using the BPCLSM and two level set functions, giving similar results as the PPCLSM. We here note that the processing of images without knowing the number of phases present is possible also using the BPCLSM, however not as elegantly as by PPCLSM.

We now give an example where one level set function is used to segment a grayscale image of a galaxy into two phases. The initial image u_0 is not easily approximated by a constant function taking only two different constant values. By

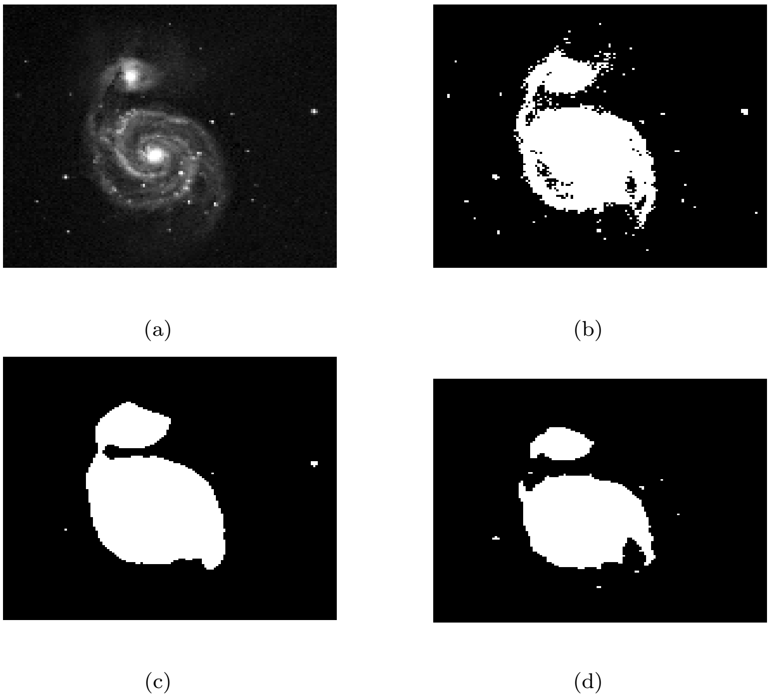


Figure 5: An image of a galaxy is processed using the modified (quicker) version of the augmented Lagrangian algorithm. (a) The original image u_0. (b) A piecewise constant approximation to u_0 after 20 iterations with $\beta = 1 \cdot 10^{-6}$. (c) Another piecewise constant approximation with $\beta = 3 \cdot 10^{-4}$. We compare with the result using CV in (d), using 300 iterations.

varying the regularization term β we control the connectivity of the segmented image. In Figure 5 we show u_0 and results of two choices for the regularization parameter β.

By inspecting the results in figure 5 (b) and (c) we see that we can control the segmented image according to what is important for the observer. The portion of the galaxy where the density of stars is high is best expressed when β is chosen to be big. Not so dense regions of the galaxy are taken into account when the regularization parameter is relaxed. As expected, the resulting image is more oscillatory when a smaller β is used.

We also segment an MR-image of a brain by using two level set functions. The goal is to partition the image into three different tissue classes in addition to the background. The numerical result is shown in Figure 6. Note that the connectivity of each phase can be controlled by β.

6. Extension of PPCLSM to 3D-images

Before we conclude this paper, we return to PPCLSM and show a preliminary numerical example where a $3D$ volume is segmented into four different subdomains of R^3 (three objects and the background). The only difference from the $2D$ case is

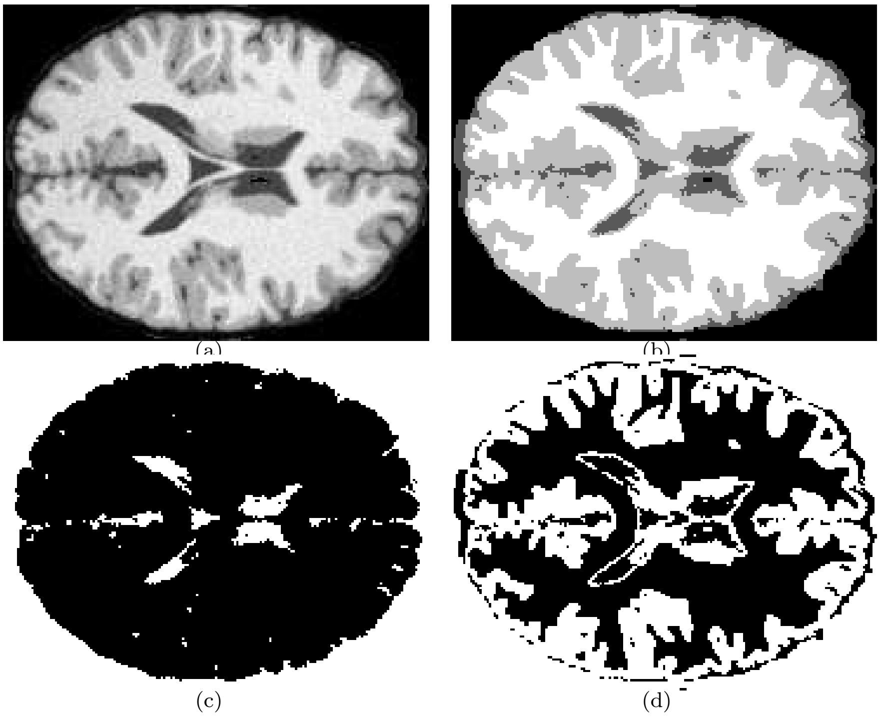


Figure 6: Two level set functions are used to find four regions in a MR-image u_0 of a brain. (a) This is a synthetically produced image, downloaded from www.bic.mni.mcgill.ca/brainweb. The value for $\beta = 1e^{-3}$. (b) The piecewise constant approximation to u_0. (c)–(d) The piecewise constant level set functions ϕ_1 and ϕ_2. In (c)–(d), white represents 1 and black represents -1.

that now ϕ is defined on a subset of R^3 instead of R^2. We observe that the method is able to extract the different objects in a very precise sense.

7. Conclusion

In this article we have introduced a framework for subdomain identification. We have also pointed out two methods for image segmentation using this framework. The method is easy to be extended to high dimensional problems. We are also working on a Newton-type of iteration for improving the convergence properties of the method. Numerical experiments indicate that these methods are able to trace interfaces with complicated geometries and sharp corners. The level set functions are discontinuous at convergence, but the minimization functionals are smooth and at least locally convex.The PPCLSM is favorable in terms of computational complexity and memory requirements, and in terms of handling cases where a priori information of the number of subdomains is lacking. Molding BPCLSM into existing software for level set methods is possibly easier than PPCLSM because of similarities in the machinery of standard level set methods. Our methods are not moving the interfaces during the iterative procedure, and thus have some advantages in treating geometries, for example in situations where inside "holes" need to be identified. Using our approach, we have removed the reinitialization procedure sometimes needed in traditional level set methods. We have proposed and

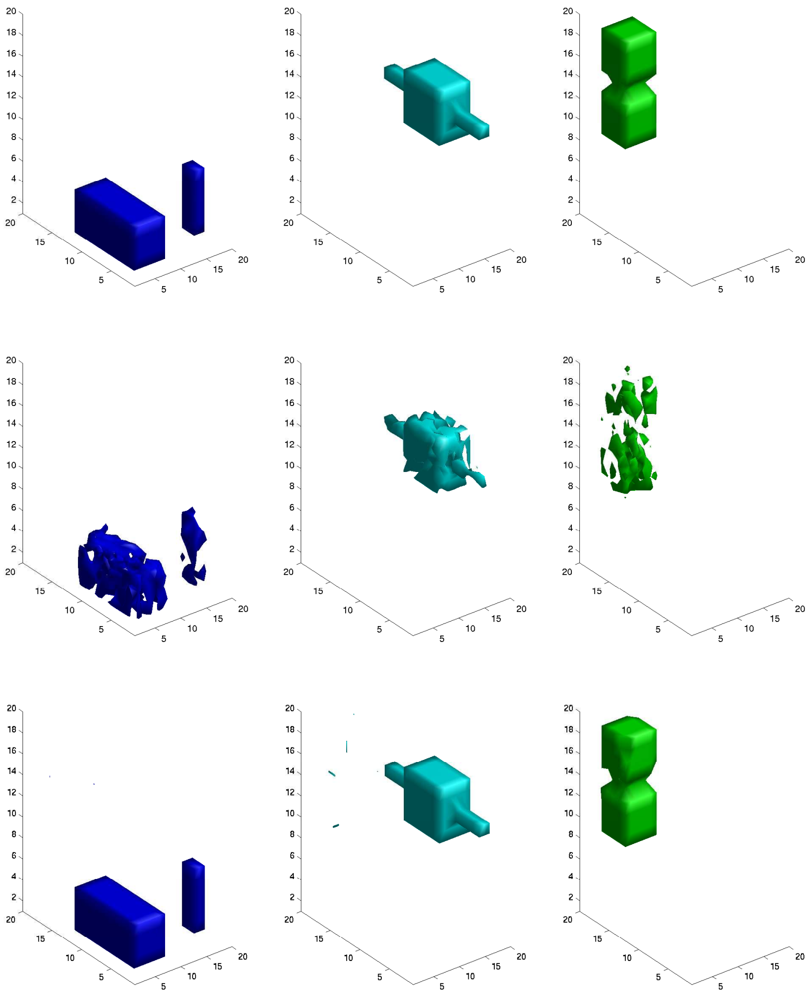


Figure 7: Example of 3D segmentation where a box holds several objects that should be identified. First row: True objects inside the box. Second row: Observed objects with noise inside the box. Last row: Objects found by PPCLSM.

demonstrated the validity of an alternative approach for interface identification, in particular for image segmentation.

Acknowledgments

We acknowledge support from the Norwegian Research Council.

References

[1] S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations," *J. Comput. Phys.*, vol. 79, no. 1, pp. 12–49, 1988.

[2] S. Osher and R. Fedkiw, "Level set methods: An overview and some recent results," *J: Comput. Phys.*, vol. 169, no. 2, pp. 463–502, 2001.

[3] ——, *Level Set Methods and Dynamic Implicit Surfaces*, ser. Applied Mathematical Sciences. Springer, 2003, vol. 153.

[4] J. A. Sethian, *Level set methods and fast marching methods*, 2nd ed., ser. Cambridge Monographs on Applied and Computational Mathematics. Cambridge: Cambridge University Press, 1999, vol. 3, evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science.

[5] X.-C. Tai and T. F. Chan, "A survey on multiple level set methods with applications for identifying piecewise constant functions," *International J. Numerical Analysis and Modelling*, vol. 1, no. 1, pp. 25–48, 2004.

[6] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Trans. Image Processing*, vol. 10, no. 2, pp. 266–277, 2001.

[7] T. F. Chan and X.-C. Tai, "Level set and total variation regularization for elliptic inverse problems with discontinuous coefficients," *J. Comput. Phys.*, vol. 193, pp. 40–66, 2003.

[8] R. P. Fedkiw, G. Sapiro, and C.-W. Shu, "Shock capturing, level sets, and PDE based methods in computer vision and image processing: a review of Osher's contributions," *J. Comput. Phys.*, vol. 185, no. 2, pp. 309–341, 2003.

[9] M. Sussman, P. Smereka, and S. Osher, "A level set approach for computing solutions to incompressible two phase flow," *J. Comput. Phys.*, vol. 114, pp. 146–159, 1994.

[10] L. A. Vese and T. F. Chan, "A multiphase level set framevork for image segmentation using the mumford and shah model," *International Journal of Computer Vision*, vol. 50, no. 3, pp. 271–293, 2002.

[11] F. Gibou and R. Fedkiw, "A fast hybrid k-means level set algorithm for segmentation," *Stanford Technical Report*, 2002.

[12] B. Song and T. Chan, "A fast algorithm for level set based optimization," *CAM-UCLA*, no. 68, 2002, under revision for publication in SIAM J. Sci. Comput.

[13] S. Baldo, "Minimal interface criterion for phase transitions in mixtures of Cahn-Hilliard fluids," *Ann. Inst. H. Poincaré Anal. Non Linéaire*, vol. 7, no. 2, pp. 67–90, 1990.

[14] B. Bourdin and A. Chambolle, "Design-dependent loads in topology optimization," *ESAIM Control Optim. Calc. Var.*, vol. 9, pp. 19–48 (electronic), 2003.

[15] J. Müller and M. Grant, "Model of surface instabilities induced by stress," *PHYSICAL REVIEW LETTERS*, vol. 82, no. 8, 1999.

[16] C. Samson, L. Blanc-Féraud, G. Aubert, and J. Zerubia, "A variational model for image classification and restoration," *TPAMI*, vol. 22, no. 5, pp. 460–472, may 2000.

[17] D. Mumford and J. Shah, "Optimal approximation by piecewise smooth functions and assosiated variational problems," *Comm. Pure Appl. Math*, vol. 42, p. 577685, 1989.

[18] A. Chambolle, "Image segmentation by variational methods: Mumford and Shah functional and the discrete approximations," *SIAM J. Appl. Math.*, vol. 55, no. 3, pp. 827–863, 1995.

[19] J. Lie, M. Lysaker, and X.-C. Tai, "A variant of the levelset method and applications to image segmentation," *CAM-UCLA*, 2003.

[20] ——, "A binary level set model and some applications to image processing," *UCLA, CAM 04-31*, 2004.

[21] M. Kass, A.Witkin, and D. Terzopulos, "Snakes: Active contour models," *Int. J. Comput. Vision*, vol. 1, pp. 321–331, 1988.

[22] V. Caselles, F. Catte, T. Coll, and F. Dibos, "A geometric model for active contours in image processing," *Numerische Mathematik*, vol. 66, pp. 1–31, 1993.

[23] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithm," *Physica D.*, vol. 60, pp. 259–268, 1992.

[24] D. P. Bertsekas, *Constrained optimization and Lagrange multiplier methods*, ser. Computer Science and Applied Mathematics. New York: Academic Press Inc., 1982.

[25] J. Nocedal and S. J. Wright, *Numerical optimization*, ser. Springer Series in Operations Research. New York: Springer-Verlag, 1999.

[26] H.-K. Zhao, T. Chan, B. Merriman, and S. Osher, "A variational level set approach to multiphase motion," *J. Comput. Phys.*, vol. 127, no. 1, pp. 179–195, 1996.

[27] K. Kunisch and X.-C. Tai, "Sequential and parallel splitting methods for bilinear control problems in Hilbert spaces," *SIAM J. Numer. Anal.*, vol. 34, pp. 91–118, 1997.

[28] T. F. Chan and X.-C. Tai, "Identification of discontinuous coefficients in elliptic problems using total variation regularization," *SIAM J. Sci. Comput.*, vol. 25, pp. 881–904, 2003.

Institute of Mathematics, University of Bergen, Joh. Brunsgate 12, N-5008 Bergen, Norway
E-mail: johanl@mi.uib.no
URL: http://www.mi.uib.no/BBG

Simula Research Laboratory, M. Linges v 17, Fornebu P.O.Box 134, N-1325 Lysaker, Norway
E-mail: mariul@simula.no
URL: http://www.simula.no/people

Institute of Mathematics, University of Bergen, Joh. Brunsgate 12, N-5008 Bergen, Norway
E-mail: tai@mi.uib.no
URL: http://www.mi.uib.no/~tai