A Direct Approach Toward Global Minimization for Multiphase Labeling and Segmentation Problems

Ying Gu, Li-Lian Wang, and Xue-Cheng Tai

***Abstract*—This paper intends to extend the minimization algorithm developed by Bae, Yuan and Tai [IJCV, 2011] in several directions. First, we propose a new primal-dual approach for global minimization of the continuous Potts model with applications to the piecewise constant Mumford–Shah model for multiphase image segmentation. Different from the existing methods, we work directly with the binary setting without using convex relaxation, which is thereby termed as a direct approach. Second, we provide the sufficient and necessary conditions to guarantee a global optimum. Moreover, we provide efficient algorithms based on a reduction in the intermediate unknowns from the augmented Lagrangian formulation. As a result, the underlying algorithms involve significantly fewer parameters and unknowns than the naive use of augmented Lagrangian-based methods; hence, they are fast and easy to implement. Furthermore, they can produce global optimums under mild conditions.**

***Index Terms*—Augmented Lagrangian method (ALM), Chambolle's algorithm, continuous Potts model, global optimum, multiclass labeling, multiphase segmentation, Mumford–Shah model, primal-dual formulation.**

I. Introduction

THE MULTICLASS labeling and multiphase segmentation problems share some similarity in nature, as typically, both of them aim to find a partition of an image into m disjoint regions (phases or classes) according to some optimization rule. As a matter of fact, the former term is commonly used in computer vision, whereas the latter one often appears in the variational and partial differential equation community. A typical model for piecewise constant multiphase image segmentation that minimizes the total interface of edge sets is the piecewise constant Mumford–Shah model (PCMSM) [25]. Thus,

$$\min_{\{c_i,\Omega_i\}_{i=1}^m}\left\{E_{\rm PCMS}\left(\{c_i\}_{i=1}^m,\Gamma\right):=\lambda\sum_{i=1}^m\int_{\Omega_i}|c_i-I|^2d\boldsymbol{x}+|\partial\Omega_i|\right\},\quad\lambda>0\quad(1)$$

where $I:\Omega\to\mathbb{R}$ is the input image, $\{c_i\}$ are the optimal mean values, and the subregions $\{\Omega_i\}$ form a nonoverlapping partition of Ω (with $\Gamma=\cup_{i=1}^m\partial\Omega_i$). With an appropriate numerical implementation, this model has many applications in segmenting images with nearly piecewise constant intensities or in finding a simplified "cartoon" approximation of a given image (see, e.g., [5], [12], and [21] and the references therein). The development of fast and robust methods for multiphase segmentation has attracted many recent attentions and been yet challenging. Indeed, even the intensity values $\{c_i\}$ are known a prior, minimizing the PCMSM is a hard task. In this situation, it turns out to be a special case of the (continuous) Potts model [32] for multiclass labeling without favoring ordering

$$\min_{\{\Omega_i\}_{i=1}^m}\left\{E_{\rm PTS}(\Gamma):=\lambda\sum_{i=1}^m\int_{\Omega_i}f_i(\boldsymbol{x})d\boldsymbol{x}+\sum_{i=1}^m|\partial\Omega_i|\right\}\quad\text{subject to}\quad\cup_{i=1}^n\Omega_i=\Omega\ \text{with}\quad\Omega_k\cap\Omega_l=\emptyset,\forall k\neq l\quad(2)$$

where the parameter $\lambda>0$, and $\{f_i\}_{i=1}^m$ can be viewed as intrinsic forces to enforce the criterion of classification and labeling. In general, the multiclass labeling problem is to assign each pixel $\boldsymbol{x}\in\Omega$ a unique label l from a set of m labels (or classes) $\{l_1,\ldots,l_m\}$. It is known that it is NP-hard when it is approached in a "discrete" manner, as the number of unknowns exponentially grows with the size of the problem. Accordingly, many recent attempts have tackled this problem from a continuous point of view by minimizing the continuous Potts model (2) (see, e.g., [2], [6], [16], [29], and [31] and the references

Manuscript received April 24, 2011; revised August 29, 2011 and December 16, 2011; accepted December 25, 2011. Date of publication January 02, 2012; date of current version April 18, 2012. This work was supported by the Singapore Ministry of Education (MOE) under Grant T207B2202 and by the Singapore National Research Foundation under Grant NRF2007IDM-IDM002-010. The Associate Editor coordinating the review of this manuscript and approving it for publication was Prof. Gang Hua.

Y. Gu and L.-L. Wang are with the Division of Mathematical Sciences, School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore 637371.

X.-C. Tai is with the Department of Mathematics, University of Bergen, Bergen, Norway, and also with the Division of Mathematical Sciences, School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore 637371.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2011.2182522

therein). An important issue in solving (2) is to find a convex approximation of the nonconvex problem

$$\min_{\{u_i\}_{i=1}^m} \left\{ E\left(\{u_i\}_{i=1}^m\right) := \lambda \sum_{i=1}^m \int_\Omega f_i(\boldsymbol{x}) u_i(\boldsymbol{x}) d\boldsymbol{x} + \sum_{i=1}^m \mathrm{TV}(u_i) \right\} \quad (3)$$

subject to $u_i \in \{0,1\}$ and $\sum_{i=1}^m u_i = 1$ on Ω, where $\{u_i\}$ are the labeling functions, and the total variation (TV) is defined by

$$\mathrm{TV}(u) = \int_\Omega |Du| = \sup_{\boldsymbol{p}\in S} \int_\Omega u \operatorname{div} \boldsymbol{p}\, d\boldsymbol{x} \quad (4)$$

with

$$S := \left\{ \boldsymbol{p} = (p_1, p_2) \in C_c^1(\Omega; \mathbb{R}^2) : |\boldsymbol{p}| \le 1, \quad \forall \boldsymbol{x} \in \Omega \right\} \quad (5)$$

where $|\boldsymbol{p}| = \sqrt{p_1^2 + p_2^2}$. The optimum u_i is expected to be the indictor function $\mathbf{1}_{\Omega_i}$ of Ω_i, i.e., a binary value.

Indeed, a major class of methods is based on the convex relaxation of the admissible set by allowing for the labeling functions to take "intermediate" values from the unit simplex. Thus,

$$\Delta_+ := \left\{ \vec{u} := (u_1, \ldots, u_m) \in \mathbb{R}^m : u_i \in [0,1] \text{ and } \sum_{i=1}^m u_i = 1, \forall \boldsymbol{x} \in \Omega \right\}. \quad (6)$$

Here, we just mention a few contributions along this line. Zach *et al.* [42] minimized the energy in (3) over (6) by introducing additional variables to decouple the TV term and the data term, whereas Lellmann *et al.* [16] replaced the TV term by nonisotropic TV: $\int_\Omega (\sum_i |\nabla u_i|^2)^{1/2} d\boldsymbol{x}$ and applied a Douglas–Rachford splitting algorithm for solving the modified model (also refer to [17] for an up-to-date review). Pock *et al.* [31] proposed a primal-dual projected gradient algorithm for minimizing an energy functional different from (3) that produced a tighter bound on the energy and often (computationally) led to the global minimizer. Li and Ng [19], [20] applied the same labeling technique to the piecewise constant Mumford–Shah-type model with data fidelity involving fuzzy membership functions and minimized the resulting model by using alternative minimization methods as in [5]. More recently, Bae *et al.* [2] have proposed a dual-type method for the approximate model

$$\min_{\vec{u}\in\Delta_+} E_\varepsilon(\vec{u}) := \sum_{i=1}^m \left\{ \mathrm{TV}(u_i) + \int_\Omega (\lambda f_i u_i + \varepsilon u_i \log u_i) d\boldsymbol{x} \right\} \quad (7)$$

where $\varepsilon > 0$ is sufficiently small. The additional log-sum term can be essentially viewed as a penalization of the constraint $u_i = 0, 1$. Sufficient conditions for the existence of a global optimum solution of the primal-dual model were derived. Moreover, the comprehensive comparison conducted in [2] demonstrated the dual-type approach outperformed most of the aforementioned existing approaches and could achieve global minimizers at least computationally. In [23], the TV term in (3) was replaced by another term that approximates the length term. Afterward, a special linearization technique was used to derive an algorithm that was shown to be fast without solving any equations (since all subproblems admitted explicit solutions, and certain narrowband technique could be applied to accelerate the algorithm).

Another family of methods based on convex formulation via functional lifting and embedding in a higher dimensional space has also attracted much recent attention (see, e.g., [6], [7], [26], [30], and [31]).

This paper is motivated by [2], and the main features and contributions can be summarized as follows.

- Different from most of the previous works, we stick with the binary setting [i.e., without relaxing the admissible set as in (6)] by seeking the minimizer from

$$\mathcal{A} := \left\{ \vec{u} = (u_1, u_2, \ldots, u_m) : u_i^2 = u_i, \sum_{i=1}^m u_i = 1 \right\} \quad (8)$$

  and conduct a direct analysis for the primal-dual formulation of (3), i.e.,

$$\min_{\vec{u}\in\mathcal{A}} \max_{\vec{\boldsymbol{p}}\in\boldsymbol{S}} \left\{ E(\vec{u}, \vec{\boldsymbol{p}}) := \sum_{i=1}^m \int_\Omega u_i (\operatorname{div} \boldsymbol{p}_i + \lambda f_i) d\boldsymbol{x} \right\} \quad (9)$$

  where $\vec{\boldsymbol{p}} = (\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_m)$ and $\boldsymbol{S} = S^m$ [see (5) for the definition of S]. We provide the sufficient and necessary conditions for finding the global optimum of the primal-dual model (9). Although some results have been derived earlier in [2], the arguments and means are quite different.
- Based on the augmented Lagrangian method (ALM), we obtain an explicit expression of the primal variables in terms of the dual variables, where the primal variables take the binary values in most of the situations. As a result, the thresholding is not necessary in this case. Moreover, the iterative algorithm contains almost the minimum number of parameters and is easy to initialize. The essential step is very analogous to the Chambolle's algorithm [8] for image denoising; hence, the method is expected to be fast and robust.
- The approach can be also interpreted as a multiphase level set method based on piecewise constant interpolation of the phases (or classes); thus, it is anticipated to be stabler than the classical level set method [38] and the piecewise constant level set method (PCLSM) [21] (a global polynomial interpolation).

The rest of this paper is organized as follows. We formulate and analyze the algorithm in the forthcoming section. More precisely, we derive the sufficient and necessary conditions for finding the global optimum of the primal-dual model (9) without using convex relaxation. We develop the algorithm that can oftentimes achieve the global optimum. We conduct a comparison study and provide ample numerical results to show the strengths

and performance of the proposed method for multiclass labeling and multiphase image segmentation in Section III.

II. Analysis of the Primal-Dual Model and the Algorithm

In this section, we conduct analysis of the global optimum of the primal-dual model (9) and introduce the fast algorithms. Due to the similar nature of the labeling and segmentation problems, we shall not distinguish the words "class" and "phase" and, likewise, "multiclass" and "multiphase" throughout the paper.

A. Two-Phase Case

To provide some insights of the algorithms for multiphase problems, we first discuss the two-phase case. In this situation, the admissible set (8) can be characterized by a single labeling function u_1 as $u_2 = 1 - u_1$. For convenience, we define $u = 2u_1 - 1$; hence, we have $u_1 = (1+u)/2$ and $u_2 = (1-u)/2$. Accordingly, the constraint on u_1 and u_2 becomes $u^2 = 1$, and by (3)

$$\min_{u^2=1}\left\{\lambda\int_\Omega\left(f_1\frac{1+u}{2}+f_2\frac{1-u}{2}\right)d\boldsymbol{x}+\mathrm{TV}(u)\right\}$$
$$=\min_{u^2=1}\left\{\lambda\int_\Omega gu\,d\boldsymbol{x}+\mathrm{TV}(u)+\lambda\int_\Omega\frac{f_1+f_2}{2}d\boldsymbol{x}\right\} \quad (10)$$

where $g = (f_1 - f_2)/2$. Thus, by dropping the last term, we obtain from the definition (4) that

$$\min_{u^2=1}\max_{\boldsymbol{p}\in S}\{L(u,\boldsymbol{p}) := \langle u, \mathrm{div}\boldsymbol{p}+\lambda g\rangle\} \quad (11)$$

where $\langle u, v\rangle = \int_\Omega uv\,d\boldsymbol{x}$, and S is the same as in (5).

The global optimal u is the binary value ± 1. The following characterization is essential for the development of the algorithm.

Theorem 2.1: Let

$$\boldsymbol{p}^* = \arg\min_{\boldsymbol{p}\in S}\int_\Omega|\mathrm{div}\boldsymbol{p}+\lambda g|d\boldsymbol{x}. \quad (12)$$

Then, the pair $(u^*, \boldsymbol{p}^*)$ is the optimum of the primal-dual problem (11), if and only if $(u^*, \boldsymbol{p}^*)$ satisfies

$$|u^*| = 1, \quad (\mathrm{div}\boldsymbol{p}^*+\lambda g)+|\mathrm{div}\boldsymbol{p}^*+\lambda g|u^* = 0 \text{ a.e. on } \Omega. \quad (13)$$

Proof: Notice that

$$L(u,\boldsymbol{p}) \geq -\|u\|_{L^\infty(\Omega)}\|\mathrm{div}\boldsymbol{p}+\lambda g\|_{L^1(\Omega)}$$

where $\|\cdot\|_{L^p(\Omega)}$ is the usual L^p-norm. Therefore, for any u in the admissible set, we have

$$\max_{\boldsymbol{p}\in S}L(u,\boldsymbol{p}) \geq \max_{\boldsymbol{p}\in S}\left(-\|\mathrm{div}\boldsymbol{p}+\lambda g\|_{L^1(\Omega)}\right)$$
$$= -\min_{\boldsymbol{p}\in S}\|\mathrm{div}\boldsymbol{p}+\lambda g\|_{L^1(\Omega)}.$$

Therefore, the optimum is attained if and only if the equality holds, that is, the second identity of (13) is true. ∎

Remark 2.1: It is worthwhile to point out that the solution of (11) is not unique. Indeed, the primal-dual problem (11) is equivalent to

$$\min_{u^2=1}\max_{\boldsymbol{p}\in\tilde{S}}\{L(u,\boldsymbol{p}) := \langle u, \mathrm{div}\boldsymbol{p}+\lambda g\rangle\} \quad (14)$$

where $\tilde{S}$ is a subset of S in (5) defined by

$$\tilde{S} := \{\boldsymbol{p} = (p_1, p_2) \in C_c^1(\Omega;\mathbb{R}^2) : |\boldsymbol{p}| \leq 1 \text{ and } \mathrm{div}\boldsymbol{p}+\lambda g \neq 0, \quad \forall \boldsymbol{x}\in\Omega\}.$$

Therefore, u^* is expressed by $\boldsymbol{p}^* \in \tilde{S}$ via

$$u^* = -\frac{\mathrm{div}\boldsymbol{p}^*+\lambda g}{|\mathrm{div}\boldsymbol{p}^*+\lambda g|} \quad (15)$$

which indicates that u^* takes the sign of $-(\mathrm{div}\boldsymbol{p}^*+\lambda g)$. ∎

Remark 2.2: The situation is reminiscent to the Chambolle's dual algorithm [8] for the Rudin–Osher–Fatemi (ROF) model [34]. Thus,

$$\min_u\left\{\mathrm{TV}(u)+\frac{\mu}{2}\|u-f\|^2_{L^2(\Omega)}\right\}, \quad \mu>0 \quad (16)$$

where f is a given noisy image. In this case, the primal-dual problem takes the form

$$\min_u\max_{\boldsymbol{p}\in S}\int_\Omega\left(u\mathrm{div}\boldsymbol{p}+\frac{\mu}{2}|u-f|^2\right)d\boldsymbol{x}. \quad (17)$$

The dual algorithm in [8] is essentially based on minimizing the dual problem

$$\min_{\boldsymbol{p}\in S}\int_\Omega|\mathrm{div}\boldsymbol{p}-\mu f|^2 d\boldsymbol{x} \quad (18)$$

by solving the nonlinear equation

$$-\nabla(\mathrm{div}\boldsymbol{p}-\mu f)+|\nabla(\mathrm{div}\boldsymbol{p}-\mu f)|\boldsymbol{p} = 0. \quad (19)$$

For clarity, we sketch the derivation of (19) from (18). Following [8] (also see [11]), we take the Lagrangian $\hat{L}[\boldsymbol{p},\alpha]$ of the dual problem (18) with the Lagrange multiplier $\alpha = \alpha(\boldsymbol{x})$. Thus,

$$\hat{L}[\boldsymbol{p},\alpha] = \int_\Omega\left(\mu f\mathrm{div}\boldsymbol{p}-\frac{1}{2}(\mathrm{div}\boldsymbol{p})^2+\frac{\alpha}{2}\left(1-|\boldsymbol{p}|^2\right)\right)d\boldsymbol{x}. \quad (20)$$

By the optimality condition with respect to $\boldsymbol{p}$

$$-\nabla(\mathrm{div}\boldsymbol{p}-\mu f)+\alpha\boldsymbol{p} = 0. \quad (21)$$

The complementary condition of the Lagrange multiplier implies that if $|\boldsymbol{p}| = 1$ at the optimum, then $\alpha > 0$, whereas if $|\boldsymbol{p}| < 1$, then $\alpha = 0$. In any case, we have $\alpha = |\nabla(\mathrm{div}\boldsymbol{p}-\mu f)|$. This gives (19).

It is important to notice the difference between (12) and (18), that is, L^1-minimization [due to the presence of the constraint $u^2 = 1$ in (11)] versus L^2-minimization.

In [39], a different principle (cf. [5]) was adopted to derive a Chambolle-type algorithm for two-phase segmentation. However, the current approach appears more natural. ∎

Now, we follow the procedure as above to solve the dual problem (12). Similar to (20), we define

$$\tilde{L}[\boldsymbol{p},\gamma]=\int_{\Omega}\left(|\mathrm{div}\boldsymbol{p}+\lambda g|+\frac{\gamma}{2}\left(|\boldsymbol{p}|^2-1\right)\right)d\boldsymbol{x}.$$

This leads to the counterpart of (21), i.e.,

$$-\nabla\left(\frac{\mathrm{div}\boldsymbol{p}+\lambda g}{|\mathrm{div}\boldsymbol{p}+\lambda g|}\right)+\gamma\boldsymbol{p}=0. \tag{22}$$

By the same argument [see the description below (21)], we obtain $\gamma=|\nabla((\mathrm{div}\boldsymbol{p}+\lambda g)/(|\mathrm{div}\boldsymbol{p}+\lambda g|))|$. Inserting it into (22) yields the nonlinear equation analogous to (19), i.e.,

$$-\nabla\left(\frac{\mathrm{div}\boldsymbol{p}+\lambda g}{|\mathrm{div}\boldsymbol{p}+\lambda g|}\right)+\left|\nabla\left(\frac{\mathrm{div}\boldsymbol{p}+\lambda g}{|\mathrm{div}\boldsymbol{p}+\lambda g|}\right)\right|\boldsymbol{p}=0 \tag{23}$$

which can be solved by a gradient descent approach. More precisely, we consider the gradient decent flow of (23). Thus,

$$\frac{\partial\boldsymbol{p}}{\partial t}=-\nabla u-|\nabla u|\boldsymbol{p}\quad\text{with}\quad u=-\frac{\mathrm{div}\boldsymbol{p}+\lambda g}{|\mathrm{div}\boldsymbol{p}+\lambda g|} \tag{24}$$

where the last identity is due to (13), and with a little abuse of notation, we still use $\boldsymbol{p}$ and u to denote the unknowns although they depend on the artificial variable t. Let τ be the time step size and $\boldsymbol{p}^n$ be the approximation of $\boldsymbol{p}$ at $t=n\tau$. To avoid division by zero, we adopt the conventional regularization (cf. [28]) and define

$$u^n:=-\frac{\mathrm{div}\boldsymbol{p}^n+\lambda g}{|\mathrm{div}\boldsymbol{p}^n+\lambda g|_\beta} \tag{25}$$

where $|\mathrm{div}\boldsymbol{p}^n+\lambda g|_\beta=|\mathrm{div}\boldsymbol{p}^n+\lambda g|+\beta$ for some sufficiently small $\beta>0$. Then, we resort to the semi-implicit discretization in time as in [8], to solve (24). Thus,

$$\begin{aligned}\frac{\boldsymbol{p}^{n+1}-\boldsymbol{p}^n}{\tau}&=-\nabla u^n-|\nabla u^n|\boldsymbol{p}^{n+1}\\ \Rightarrow\quad \boldsymbol{p}^{n+1}&=\frac{\boldsymbol{p}^n-\tau\nabla u^n}{1+\tau|\nabla u^n|}.\end{aligned} \tag{26}$$

It is seen that the L^1-minimization (12) induces additional nonlinearity, compared with the L^2-minimization (17).

In view of Remark 2.1, we adopt the Merriman–Brence–Osher-type projection (see, e.g., [24] and [36])

$$\mathcal{P}_B(\kappa):=\begin{cases}1, & \text{if}\quad \kappa\geq 0\\ -1, & \text{if}\quad \kappa<0\end{cases} \tag{27}$$

to obtain the binary value u.

We summarize the algorithm as follows, where the involved differential operators can be discretized as in [8].

Algorithm 1

1. Initialization: set $\boldsymbol{p}^0=\boldsymbol{0}$ and choose β, τ, $\lambda>0$;
2. For $n=0,1,\ldots$
   (i) Compute
   $$u^n:=-\frac{\mathrm{div}\boldsymbol{p}^n+\lambda g}{|\mathrm{div}\boldsymbol{p}^n+\lambda g|_\beta};$$
   (ii) Update $\boldsymbol{p}$ by the Chambolle-type algorithm
   $$\boldsymbol{p}^{n+1}=\frac{\boldsymbol{p}^n-\tau\nabla u^n}{1+\tau|\nabla u^n|};$$
3. Endfor until some stopping rule meets;
4. Set
   $$u=\mathcal{P}_B(u^n).$$

B. Multiphase Case

With the insights from the two-phase model, we now consider the full primal-dual model (9). Observe that, for any $\vec{u}\in\mathcal{A}$, we have

$$\begin{aligned}\sum_{i=1}^m u_i(\mathrm{div}\boldsymbol{p}_i+\lambda f_i)&\geq\min_{1\leq i\leq m}\{\mathrm{div}\boldsymbol{p}_i+\lambda f_i\}\sum_{i=1}^m u_i\\ &=\min_{1\leq i\leq m}\{\mathrm{div}\boldsymbol{p}_i+\lambda f_i\},\quad\forall\boldsymbol{x}\in\Omega\end{aligned} \tag{28}$$

which implies

$$E(\vec{u},\vec{\boldsymbol{p}})\geq\int_{\Omega}\min_{1\leq i\leq m}\{\mathrm{div}\boldsymbol{p}_i+\lambda f_i\}d\boldsymbol{x} \tag{29}$$

for all $(\vec{u},\vec{\boldsymbol{p}})\in\mathcal{A}\times\boldsymbol{S}$, where the energy functional $E(\cdot,\cdot)$ is given in (9). Hence, we deduce from (29) that

$$\begin{aligned}&\max_{\vec{\boldsymbol{p}}\in\boldsymbol{S}}E(\vec{u},\vec{\boldsymbol{p}})\\ &\quad\geq\max_{\vec{\boldsymbol{p}}\in\boldsymbol{S}}\left\{E_D(\vec{\boldsymbol{p}}):=\int_{\Omega}\min_{1\leq i\leq m}\{\mathrm{div}\boldsymbol{p}_i+\lambda f_i\}d\boldsymbol{x}\right\},\forall\vec{u}\in\mathcal{A}.\end{aligned} \tag{30}$$

Accordingly, *the global optimum* $(\vec{u}^*,\vec{\boldsymbol{p}}^*)$ *of the primal-dual model* (9) *is a pair in* $\mathcal{A}\times\boldsymbol{S}$ *such that the identity of* (29) *holds, and the global minimizer* $\vec{u}^*$ *of the original model* (3) *is in* $\mathcal{A}$ *such that the identity* (30) *holds*.

Similar to Theorem 1 in [2], we have the following result on the characterization of the global optimum $(\vec{u}^*,\vec{\boldsymbol{p}}^*)$.

Theorem 2.2: Let

$$\vec{\boldsymbol{p}}^*=\arg\max_{\vec{\boldsymbol{p}}\in\boldsymbol{S}}E_D(\vec{\boldsymbol{p}}). \tag{31}$$

- Suppose that $\min_{1\le i\le m}\{\mathrm{div}\boldsymbol{p}_i^* + \lambda f_i\}$ has a unique minimum value for all $\boldsymbol{x}\in\Omega$. Then, $\vec{u}^*$ must take the form

$$u_k^* = \begin{cases} 1, & k = \arg\min_{1\le i\le m}\{\mathrm{div}\boldsymbol{p}_i^* + \lambda f_i\}, \\ 0, & \text{otherwise}, \end{cases} \tag{32}$$

  for $1\le k\le m$ and $\boldsymbol{x}\in\Omega$.
- Suppose that $\min_{1\le i\le m}\{\mathrm{div}\boldsymbol{p}_i^* + \lambda f_i\}$ has more than one minimum values at some $\boldsymbol{x}\in\Omega$, say, k values: $\mathrm{div}\boldsymbol{p}_j^*+\lambda f_j$, $j\in\{j_1, j_2, \ldots, j_k\}$. Then, $\vec{u}^*$ must satisfy

$$\sum_{i=1}^{k} u_{j_i}^* = 1 \quad \text{and} \quad u_j^* = 0, \quad j\notin\{j_1,\ldots,j_k\}. \tag{33}$$

Proof: Since the first statement is a special case of the second one, it suffices to prove the second statement. By (29), we have $E(\vec{u},\vec{\boldsymbol{p}}^*)\ge E_D(\vec{\boldsymbol{p}}^*)$ for any $\vec{u}\in\mathcal{A}$. Thus, it is enough to show that $E(\vec{u}^*,\vec{\boldsymbol{p}}^*) = E_D(\vec{\boldsymbol{p}}^*)$. It is clear that, if $\vec{u}^*$ is defined by (33), then

$$\begin{aligned}\sum_{i=1}^{m} u_i^*\left(\mathrm{div}\boldsymbol{p}_i^* + \lambda f_i\right) &= \sum_{i=1}^{k} u_{j_i}^*\left(\mathrm{div}\boldsymbol{p}_{j_i}^* + \lambda f_{j_i}\right)\\ &= \sum_{i=1}^{k} u_{j_i}^* \min_{1\le i\le m}\{\mathrm{div}\boldsymbol{p}_i^* + \lambda f_i\}\\ &= \min_{1\le i\le m}\{\mathrm{div}\boldsymbol{p}_i^* + \lambda f_i\},\end{aligned}$$

for $\boldsymbol{x}\in\Omega$ and the corresponding $1\le k\le m$. Integrating the above identities over Ω leads to $E(\vec{u}^*,\vec{\boldsymbol{p}}^*) = E_D(\vec{\boldsymbol{p}}^*)$.

Next, we prove that (33) is necessary. If there is a u_l^* with $l\in\{j_1,j_2,\ldots,j_k\}$, which does not correspond to a minimum value, i.e., $\mathrm{div}\boldsymbol{p}_l^* + \lambda f_l > \min_{1\le i\le m}\{\mathrm{div}\boldsymbol{p}_i^* + \lambda f_i\}$, then

$$\begin{aligned}&\sum_{i=1}^{m} u_i^*(\mathrm{div}\boldsymbol{p}_i^* + \lambda f_i)\\ &\quad= \sum_{i=1}^{k} u_{j_i}^*\left(\mathrm{div}\boldsymbol{p}_{j_i}^* + \lambda f_{j_i}\right)\\ &\quad= (1-u_l^*)\min_{1\le i\le m}\{\mathrm{div}\boldsymbol{p}_i^* + \lambda f_i\} + u_l^*\left(\mathrm{div}\boldsymbol{p}_l^* + \lambda f_l\right)\\ &\quad= \min_{1\le i\le m}\{\mathrm{div}\boldsymbol{p}_i^* + \lambda f_i\}\\ &\qquad+ u_l^*\left(\mathrm{div}\boldsymbol{p}_l^* + \lambda f_l - \min_{1\le i\le m}\{\mathrm{div}\boldsymbol{p}_i^* + \lambda f_i\}\right).\end{aligned}$$

Therefore, the optimum value can be obtained if and only if $u_l^* = 0$. ∎

Remark 2.3: It is worthwhile to point out that the necessity of (32) and (33) is under the distributional sense. Indeed, we are free to change the value of $\vec{u}^*$ at a measure zero set of Ω without affecting the integral value. ∎

Remark 2.4: A similar analysis was conducted in [2] (see [2, Th. 1]) based on convex relaxation. Here, we provided a direct concise proof, which did not require switching the min-max using Karush–Kuhn–Tucher conditions [15] and thresholding techniques. On the other hand, we could show that (33) is also necessary. ∎

We find from (31) that it is essential to solve the dual problem defined in (28). However, it is nonsmooth; hence, it appears very challenging to directly work on this dual model. The following smoothed-dual model was used in [2]:

$$\max_{\vec{\boldsymbol{p}}\in\boldsymbol{S}}\left\{E_{D,\varepsilon}(\vec{\boldsymbol{p}}) := -\varepsilon\int_\Omega \log\sum_{i=1}^{m}\exp\left(\frac{-\mathrm{div}\boldsymbol{p}_i - \lambda f_i}{\varepsilon}\right)d\boldsymbol{x}\right\} \tag{34}$$

where $0<\varepsilon\ll 1$. It is important to note that (34) turns out to be the dual model of (7).

Hereafter, we shall take a different approach and derive the algorithm based on the ALM. It is worthwhile to point out that the ALM has been recently widely used for solving minimization problems in image processing (see, e.g., [37], [40], and [41] and the references therein). Here, we use it as a tool to derive the relation between the primal and dual variables via eliminating all the intermediate parameters and unknowns. This leads to an algorithm that involves the minimum number of parameters and can be implemented as efficiently as the Chambolle's algorithm in [8].

C. Description of the Algorithm

We start with (3) and rewrite it as

$$\min_{\vec{u}}\left\{\sum_{i=1}^{m}\int_\Omega |\nabla u_i|d\boldsymbol{x} + \lambda\sum_{i=1}^{m}\int_\Omega f_i u_i d\boldsymbol{x}\right\}.$$

As with the ALM for the ROF model (see, e.g., [37] and [40]), an essential step for dealing with the TV term is to introduce an intermediate variable and reformulate the above problem into the following constrained problem:

$$\begin{aligned}&\min_{\vec{\boldsymbol{q}},\vec{u}}\left\{\sum_{i=1}^{m}\int_\Omega |\boldsymbol{q}_i|d\boldsymbol{x} + \lambda\sum_{i=1}^{m}\langle f_i, u_i\rangle\right\},\\ &\text{subject to}\quad \boldsymbol{q_i} = \nabla u_i, u_i^2 = u_i, \sum_{i=1}^{m} u_i = 1.\end{aligned}$$

Note that the gradient operator on functions of bounded variation should be understood in the weak sense (cf. [14]). Using the notion of ALM yields the unconstrained problem

$$\begin{aligned}\min_{\vec{u},\vec{\boldsymbol{q}}}\max_{\vec{\boldsymbol{\lambda}}_1,\vec{\lambda}_2,\lambda_3}&\left\{\mathcal{L}(\vec{u},\vec{\boldsymbol{q}},\vec{\boldsymbol{\lambda}}_1,\vec{\lambda}_2,\lambda_3):\right.\\ &= \sum_{i=1}^{m}\left(\langle 1, |\boldsymbol{q_i}|\rangle + \lambda\langle f_i, u_i\rangle\right.\\ &\quad+\langle\boldsymbol{\lambda}_{1i}, \boldsymbol{q_i}-\nabla u_i\rangle + \frac{r_1}{2}\langle\boldsymbol{q_i}-\nabla u_i, \boldsymbol{q_i}-\nabla u_i\rangle\\ &\quad\left.+\langle\lambda_{2i}, u_i^2-u_i\rangle + \frac{r_2}{2}\langle u_i^2-u_i, u_i^2-u_i\rangle\right)\\ &\quad+\left\langle\lambda_3, \sum_{i=1}^{m}u_i - 1\right\rangle\\ &\quad\left.+\frac{r_3}{2}\left\langle\sum_{i=1}^{m}u_i - 1, \sum_{i=1}^{m}u_i - 1\right\rangle\right\}\end{aligned} \tag{35}$$

where $r_1, r_2, r_3 > 0$ are penalization constants, and the vector-valued functions $\vec{\boldsymbol{\lambda}}_1 = (\boldsymbol{\lambda}_{11},\ldots,\boldsymbol{\lambda}_{1m})$, $\vec{\lambda}_2 = (\lambda_{21},\ldots,\lambda_{2m})$

(with $\lambda_{2i} \geq 0$) and the scalar function λ_3 are Lagrange multipliers. The optimality conditions lead to the system

$$\frac{\partial \mathcal{L}}{\partial u_i} = \lambda f_i + \mathrm{div}\boldsymbol{\lambda_{1i}} + r_1 \mathrm{div}(\boldsymbol{q_i} - \nabla u_i) + \lambda_{2i}(2u_i - 1) + r_2\left(u_i^2 - u_i\right)(2u_i - 1) + \lambda_3 + r_3\left(\sum_{i=1}^{m} u_i - 1\right) = 0 \tag{36a}$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{q_i}} = \frac{\boldsymbol{q_i}}{|\boldsymbol{q_i}|} + \boldsymbol{\lambda}_{1i} + r_1(\boldsymbol{q_i} - \nabla u_i) = 0 \tag{36b}$$

$$\frac{\partial \mathcal{L}}{\partial \boldsymbol{\lambda}_{1i}} = \boldsymbol{q_i} - \nabla u_i = 0 \tag{36c}$$

$$\frac{\partial \mathcal{L}}{\partial \lambda_{2i}} = u_i^2 - u_i = 0 \tag{36d}$$

$$\frac{\partial \mathcal{L}}{\partial \lambda_3} = \sum_{i=1}^{m} u_i - 1 = 0. \tag{36e}$$

Thus, it follows from (36b) and (36c) that

$$\boldsymbol{\lambda}_{1i} = -\frac{\nabla u_i}{|\nabla u_i|}, \quad 1 \leq i \leq m. \tag{37}$$

In fact, $\vec{\boldsymbol{\lambda}}_1$ turns out to be the vector of dual variables. To develop an efficient algorithm, it is essential to express the primal variable $\vec{u}$ in terms of $\vec{\boldsymbol{\lambda}}_1$. Using (36c)–(36e), we can simplify (36a) into

$$\lambda_{2i}(2u_i - 1) = -(\lambda f_i + \mathrm{div}\boldsymbol{\lambda}_{1i} + \lambda_3), \tag{38}$$

which implies

$$\lambda_{2i}^2\left(4u_i^2 - 4u_i + 1\right) = (\lambda f_i + \mathrm{div}\boldsymbol{\lambda}_{1i} + \lambda_3)^2.$$

Then, by (36d) (note: $\lambda_{2i} \geq 0$)

$$\lambda_{2i} = |\lambda f_i + \mathrm{div}\boldsymbol{\lambda}_{1i} + \lambda_3| = |h_i + \lambda_3|$$

where $h_i := \mathrm{div}\boldsymbol{\lambda}_{1i} + \lambda f_i$. Plugging it into (38) leads to

$$(2u_i - 1)|h_i + \lambda_3| + (h_i + \lambda_3) = 0. \tag{39}$$

It is seen that the use of the ALM allows us to derive the (reduced) primal-dual system (37) and (39) with a free variable λ_3. In contrast with the usual Uzawa algorithm for (35), our algorithm based on (37) and (39) involves the minimum number of parameters and variables; hence, the iterative solver is easy to initialize, as to be shown shortly.

An essential step is to choose λ_3 and express the primal variables $\{u_i\}$ in terms of $\{\boldsymbol{\lambda}_{1i}\}$. For this purpose, let h_k (resp. h_j) be (one of) the smallest (resp. the second smallest) value of $\{h_i\}_{i=1}^m$ (note: if all $\{h_i\}_{i=1}^m$ are equal, then $h_j = h_k$). By choosing

$$-\lambda_3 = h_k + \frac{h_j - h_k}{2}, \tag{40}$$

we find from (39) that $u_k = 1$ and $u_i = 0$ for all the i's such that $h_i \geq h_j$. It is worthwhile to point out that (39) cannot identify u_i when $h_i + \lambda_3 = 0$, i.e., $h_j = h_k$. In the computation, we regularize (39) and find

$$u_i = \frac{1}{2} - \frac{h_i + \lambda_3}{2|h_i + \lambda_3|_\beta}, \tag{41}$$

where λ_3 is defined in (40), and $|h_i + \lambda_3|_\beta = |\mathrm{div}\boldsymbol{\lambda}_{1i} + \lambda f_i + \lambda_3| + \beta$ for sufficiently small $\beta > 0$ as in (25).

Next, for fixed $\vec{u}$, (37) can be solved by applying the iterative algorithm in [8] to

$$\nabla u_i + |\nabla u_i|\boldsymbol{\lambda}_{1i} = \mathbf{0}, \quad 1 \leq i \leq m, \tag{42}$$

with $\vec{\boldsymbol{\lambda}}_1 \in \boldsymbol{S}$, as described in (23) (24) (25) (26). Note that $\vec{\boldsymbol{\lambda}}_1$ is the "approximation" of $\vec{\boldsymbol{p}}$ in Section II-B.

Remark 2.5: As predicted by Theorem 2.2, if $\vec{\boldsymbol{\lambda}}_1^*$ solves (31), then by defining $\vec{u}^*$ as

$$u_k^* = \begin{cases} 1, & k = \min\left\{\arg\min_{1\leq i\leq m}\left\{\mathrm{div}\boldsymbol{\lambda}_{1i}^* + \lambda f_i\right\}\right\} \\ 0, & \text{otherwise} \end{cases}$$

for all $1 \leq k \leq m$ and $\boldsymbol{x} \in \Omega$, we have the following properties.

(i) $\vec{u}^*$ satisfies the optimality equation (39) with $\lambda_3 = -(\mathrm{div}\boldsymbol{\lambda}_{1k}^* + \lambda f_k)$.

(ii) Under the assumption of (32), that is, $\min_{1\leq i\leq m}\{\mathrm{div}\boldsymbol{\lambda}_{1i}^* + \lambda f_i\}$ has a unique minimum value for all $\boldsymbol{x} \in \Omega$, we find from Theorem 2.2 that $\vec{u}^*$ is expected to be the unique global minimizer.

(iii) Only one component of $\vec{u}^*$ takes the value 1, whereas the other $m - 1$ components are all zero. According to Theorem 2.2, the so-defined $\vec{u}^*$ is expected to be a global minimizer, since $\vec{u}^*$ satisfies (33). ■

Now, we are ready to present the full algorithm as follows.

Algorithm 2

1. Initialization: set $\vec{\boldsymbol{\lambda}}_1^0 = \mathbf{0}$ and choose $\beta, \tau, \lambda > 0$|;
2. For $n = 0, 1, \ldots$
   (i) Compute λ_3 by (40);
   (ii) Compute $\vec{u}^n$ by (41);
   (iii) Compute $\vec{\boldsymbol{\lambda}}_1^{n+1}$ by the Chambolle-type algorithm

$$\boldsymbol{\lambda}_{1i}^{n+1} = \frac{\boldsymbol{\lambda}_{1i}^n - \tau\nabla u_i^n}{1 + \tau\left|\nabla u_i^n\right|}, \quad 1 \leq i \leq m;$$

3. Endfor until some stopping rule meets;
4. Set $\vec{\boldsymbol{\lambda}}_1^* = \vec{\boldsymbol{\lambda}}_1^{n+1}$ and define

$$u_k^* = \begin{cases} 1, & k = \min\left\{\arg\min_{1\leq i\leq m}\left\{\mathrm{div}\boldsymbol{\lambda}_{1i}^* + \lambda f_i\right\}\right\} \\ 0, & \text{otherwise} \end{cases}$$

for all $1 \leq k \leq m$.

III. Comparison Study and Numerical Results

In this section, we demonstrate the performance of the proposed algorithm by testing it on some typical images and comparing it with some relevant methods.

A. Comparison Study

To provide more insights, we compare the algorithm with the global smoothed-dual (GSD) algorithm in [2] and examine the approach from the perspective of the level set methodology.

GSD Algorithm: The dual approach in [2] is essentially based on the model (7), which approximates the original nonconvex problem (3) when $0 < \varepsilon \ll 1$. The optimum of the dual variable is obtained by solving the model (34), which is a smoothed version of the nonsmooth dual problem (31). This leads to the representation of the primal variables $\{u_i\}$ in terms of the dual variables $\{\boldsymbol{p}_i\}$, i.e.,

$$u_i^\varepsilon = \frac{\exp\left(-\frac{\mathrm{div}\boldsymbol{p}_i+\lambda f_i}{\varepsilon}\right)}{\sum_{j=1}^m \exp\left(-\frac{\mathrm{div}\boldsymbol{p}_j+\lambda f_j}{\varepsilon}\right)} = \frac{1}{\sum_{j=1}^m \exp\left(-\frac{(\mathrm{div}\boldsymbol{p}_j+\lambda f_j)-(\mathrm{div}\boldsymbol{p}_i+\lambda f_i)}{\varepsilon}\right)} \tag{43}$$

for $1 \le i \le m$ and $\boldsymbol{x} \in \Omega$. Apparently, the vector $\vec{u}^\varepsilon$ with components $\{u_i^\varepsilon\}$ is in the convex simplex Δ_+ in (6). Suppose that $\mathrm{div}\boldsymbol{p}_i + \lambda f_i$ is the unique minimum value of $\{\mathrm{div}\boldsymbol{p}_j + \lambda f_j\}_{j=1}^m$. Then, we observe from (43) that $u_i^\varepsilon \to 1$ and $u_j^\varepsilon \to 0$ (for all $j \ne i$) as $\varepsilon \to 0^+$. Under this assumption, it is expected to obtain the global minimizer in the limiting process. Note that the expression (43) can be viewed as the counterpart of (39) (40) (41).

For comparison purposes, we recall the GSD algorithm in [2], where we remain the notation in [2], and $\vec{\boldsymbol{p}}$ plays the same role as $\vec{\boldsymbol{\lambda}}_1$ in Algorithm 2.

GSD Algorithm

1. Initialization: set $\vec{\boldsymbol{p}}^0 = \boldsymbol{0}$ and choose $\varepsilon,\ \delta,\ \lambda > 0$;
2. For $n = 0, 1, \ldots$
   (i) Compute $\vec{u}^n$ by (43) with $\boldsymbol{p}_i^n$ in place of $\boldsymbol{p}_i$;
   (ii) Compute $\vec{\boldsymbol{p}}^{n+1}$ by the projection method

$$\boldsymbol{p}_i^{n+1} = \mathrm{Proj}_{\boldsymbol{S}}\left(\boldsymbol{p}_i^n + \delta\nabla u_i^n\right), \quad 1 \le i \le m$$

   where $\mathrm{Proj}_{\boldsymbol{S}}$ is the projection operator upon the convex set $\boldsymbol{S}$;
3. Endfor until some stopping rule meets;
4. Set $\vec{\boldsymbol{p}}^* = \vec{\boldsymbol{p}}^{n+1}$ and define

$$u_k^* = \begin{cases} 1, & k = \min\left\{\arg\min_{1\le i\le m}\left\{\mathrm{div}\boldsymbol{p}_i^* + \lambda f_i\right\}\right\}, \\ 0, & \text{otherwise} \end{cases}$$

for all $1 \le k \le m$.

Perspective From the Level Set Method: It is interesting to interpret (3) from the perspective of the level set method, where $\{u_i\}$ can be regarded as a piecewise constant interpolation of the phases (or subregions $\{\Omega_i\}$), that is, u_i takes binary values, i.e., $u_i = 1$, if $\boldsymbol{x} \in \Omega_i$, and $u_i = 0$ elsewhere. This should be in contrast to the level set method [28] and its important variant [21] for image segmentation.

The multiphase level set framework in [38] labels the phases by the combinations of the signs of the level set functions (usually taken as the signed distance functions). Therefore, n level set functions $\{u_j\}_{j=1}^n$ can label $m = 2^n$ phases, and the corresponding indicator functions can be expressed as the Heaviside functions $H(u_j)$. Notice that the level set functions should satisfy the Ekiron equation: $|\nabla u_j| = 1$, and this constraint should be taken into account in the implementation. To avoid such a reinitialization, one may penalize this constraint as in [18] and [22]. For the purpose of comparison, we briefly review the new level set method in [18] for (3) with $m = 2$. Let u be the signed distance function such that $u > 0$ (resp. $u < 0$) if $\boldsymbol{x} \in \Omega_1$ (resp. $\boldsymbol{x} \in \Omega_2$). Then, u should satisfy $|\nabla u| = 1$, and the characteristic function of Ω_1 (resp. Ω_2) is $H(u)$ (resp. $1 - H(u)$). The essential idea in [18] is to minimize the penalized energy

$$\min_u \left\{ E_{\mathrm{Li}}(u) := \lambda \int_\Omega gH(u)d\boldsymbol{x} + \int_\Omega \delta(u)|\nabla u| d\boldsymbol{x} + \frac{\mu}{2}\int_\Omega \left(|\nabla u| - 1\right)^2 d\boldsymbol{x} \right\}, \mu > 0 \tag{44}$$

where δ is the Dirac delta function, and $g = (f_1 - f_2)/2$. As in [18], we can minimize E_{Li} by evolving the gradient flow [see (51) below].

The PCLSM proposed in [21] uses one level set function to label multiple phases. It can be viewed as a global polynomial interpolation of the phases, as opposed to the piecewise constant interpolation in (3). More precisely, it labels the phases by

$$u = i \quad \text{if} \quad \boldsymbol{x} \in \Omega_i, \quad 1 \le i \le m \tag{45}$$

and express the indicator function of Ω_i by the Lagrange basis polynomial

$$\varphi_i(u) = \prod_{1\le j\le m; j\ne i} \frac{u-j}{i-j} = \frac{K_m(u)}{K_m'(i)(u-i)}, \tag{46}$$

i.e., $\varphi_i(i) = 1$ and $\varphi_i(j) = 0$ for $i \ne j$, where $K_m(u) = \prod_{j=1}^m (u-j)$. Under this setting, the model (3) is translated to

$$\min_{u\in\{1,\ldots,m\}} \left\{ E_P(u) := \lambda \sum_{i=1}^m \int_\Omega f_i\varphi_i(u)d\boldsymbol{x} + \sum_{i=1}^m \mathrm{TV}\left(\varphi_i(u)\right) \right\}. \tag{47}$$

By imposing the constraint $K_m(u) = 0$, we obtain the primal-dual model (cf. [6])

$$\min_{K_m=0} \max_{\vec{\boldsymbol{p}}\in\boldsymbol{S}} \left\{ F_P(u, \vec{\boldsymbol{p}}) := \sum_{i=1}^m \left\langle \varphi_i(u), \mathrm{div}\boldsymbol{p}_i + \lambda f_i \right\rangle \right\}. \tag{48}$$

We see the resemblance between the primal-dual models (9) and (48). Notice that φ_i is a polynomial of u of degree m. As a result, the reduction using the ALM similar to the procedure in Section II-C leads to the coupled system, where it is not possible to derive the expression of u in terms of the dual variables. Hence, the algorithm becomes extremely complicated.

For the sake of comparison, we present the counterpart of (44) for the two-phase case. As described in Section II-A, we

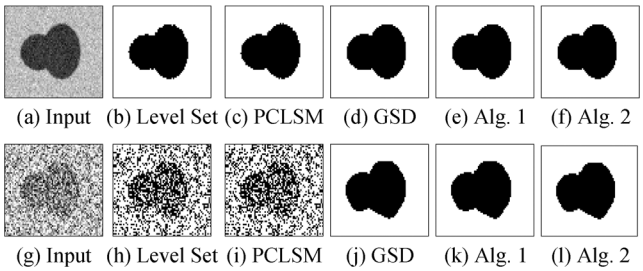


Fig. 1. Comparison of five algorithms. Here, we take $\lambda = 5 \times 10^{-5}$, $\tau = 0.1$, $\delta = 0.05$, $\tau_{\rm ls} = \tau_{\rm pclsm} = 10^{-5}$, $\epsilon = 1$, $\mu = 1$, $\bar{\mu} = 10^3$, $\varepsilon = 0.1$, and $f_i = |I - c_i|^2 (i = 1, 2)$. Row 1 (a)–(f): noise level $d = 0.001$ and $\beta = 0.01$; Row 2 (g)–(l): noise level $d = 0.1$ and $\beta = 0.1$. (a) Input. (b) Level Set. (c) PCLSM. (d) GSD. (e) Alg. 1. (f) Alg. 2. (g) Input. (h) Level Set. (i) PCLSM. (j) GSD. (k) Alg. 1. (l) Alg. 2.

label Ω_1 (resp. Ω_2) by $u = 1$ (resp. $u = -1$), and then, we can formulate (3) with $m = 2$ into (10). By penalizing the constraint $u^2 = 1$, we have the counterpart of (44), i.e.,

$$\min_u \left\{ E_{\rm PC}(u) := \lambda \int_\Omega g u d\boldsymbol{x} + \int_\Omega |\nabla u| d\boldsymbol{x} + \frac{\tilde{\mu}}{4} \int_\Omega (u^2 - 1)^2 d\boldsymbol{x} \right\}, \quad \tilde{\mu} > 0 \quad (49)$$

where $g = (f_1 - f_2)/2$. One can evolve the gradient flow to minimize (49). Thus,

$$\frac{\partial u}{\partial t} = \operatorname{div}\left(\frac{\nabla u}{|\nabla u|}\right) + \tilde{\mu}(1 - u^2)u - \lambda g, \quad (50)$$

which is closely related to the Allen–Cahn model for phase transitions [1] but with a nonlinear TV diffusion operator rather than a linear Laplace diffusion. It was named as the TV Allen–Cahn model in [33] and [39].

Next, we compare the performance of five algorithms: the level set method in [18] [cf. (44)], the PCLSM [cf. (49)], GSD Algorithm, and our proposed Algorithms 1–2. Here, we minimize (44) based on the code on the " http://www.engr.uconn.edu/cmli/research/" for [18] and also test the algorithms on the image (but with noise) from this site [see Fig. 1(a) and (g)]. More precisely, the work [18] minimized (44) by evolving the gradient flow

$$u^{n+1} = u^n + \tau_{\rm ls}\left(\operatorname{div}\left(\frac{\nabla u^n}{|\nabla u^n|}\right) + \frac{\mu}{\delta_\epsilon(u^n)}\left(\Delta u^n - \operatorname{div}\left(\frac{\nabla u^n}{|\nabla u^n|}\right)\right) - \lambda g\right) \quad (51)$$

where $\tau_{\rm ls}$ is the time step size, and the smoothed Delta function δ_ϵ is defined as in the code for [18]. For ease of comparison, we evolve (50) as (51). Thus,

$$u^{n+1} = u^n + \tau_{\rm pclsm}\left(\operatorname{div}\left(\frac{\nabla u^n}{|\nabla u^n|}\right) + \tilde{\mu}\left(1 - (u^n)^2\right)u^n - \lambda g\right). \quad (52)$$

TABLE I
COMPARISON OF FIVE ALGORITHMS

Figure 1(a)	iterations	time (second)	energy
Level Set [18]	306	2.0	0.331×10^3
PCLSM (49)	415	2.3	0.407×10^3
GSD	17	0.4	0.313×10^3
Alg. 1	11	0.3	0.306×10^3
Alg. 2	14	0.3	0.309×10^3
Figure 1(g)	iterations	time (second)	energy
Level Set [18]	1000	5.6	2.32×10^3
PCLSM (49)	1000	4.8	1.31×10^3
GSD	72	1.4	0.827×10^3
Alg. 1	40	0.9	0.833×10^3
Alg. 2	53	1.0	0.832×10^3

Here, spatial differential operators are discretized, and initial conditions are taken as in [18].

We compare the numerical energy corresponding to the original model (2) defined by

$$E(u^n) = \int_\Omega |\nabla u^n| d\boldsymbol{x} + \lambda \int_\Omega g u^n d\boldsymbol{x} \quad \text{with} \quad g = (f_1 - f_2)/2$$

where u^n is respectively computed by (51), (52), GSD, Algorithm 1, and Algorithm 2 for the same λ. Correspondingly, we use the stopping rule: $|E(u^{n+1}) - E(u^n)| < \varsigma$. We tabulate in Table I the iteration number, computational time, and numerical energy $E(u^n)$ obtained by five algorithms (terminated with $\varsigma = 10^{-2}$, and we refer to the caption of Fig. 1 for the computational setup) for two images with different noise levels in Fig. 1(a) and (g). We present in Fig. 1 the final segmented images. We find that, at low noise level, all five algorithms produce satisfactory segmentation results with similar numerical energy, but the latter three algorithms converge much faster. However, when we increase the level of noise from 0.001 to 0.1, the former two algorithms fail to converge, whereas GSD and our proposed algorithms lead to acceptable results, and are quite robust to large noise.

B. More Numerical Tests

The previous comparison study demonstrated the advantages of GSD [2] and Algorithms 1–2. Indeed, the recent work [2] conducted a comprehensive comparison of GSD with several popular methods including the Alpha expansion and Alpha-Beta swap [3], [4], the method of Pock *et al.* [31], and the algorithm in Lellmann *et al.* [16]. The GSD has exhibited the strength over these algorithms in most comparison tests. In what follows, we put our emphasis on the comparison with GSD and examine the performance of Algorithms 1–2 over the methods in [3], [4], and [16].

To set up a relatively sound criterion for comparison, we choose some typical images used in the aforementioned papers for testing the algorithms, unify the choice of the parameters of the model, and compare the numerical energy associated with the original model (3). Moreover, we use the relative dynamic error

$$E_{l^1}(\vec{\boldsymbol{p}}^{n+1}, \vec{\boldsymbol{p}}^n) := \frac{\|\vec{\boldsymbol{p}}^{n+1} - \vec{\boldsymbol{p}}^n\|_{l^1}}{\|\vec{\boldsymbol{p}}^{n+1}\|_{l^1}} \leq \eta \quad (53)$$

where $||\vec{\boldsymbol{p}}||_{l^1} = \max_i(\sum |\boldsymbol{p}_i|)$ (with $\vec{\boldsymbol{\lambda}}_1$ in place of $\vec{\boldsymbol{p}}$ for Algorithm 2), and the summation is over all the pixels, for a prescribed tolerance $\eta > 0$ as the stopping rule, for all the comparisons below.

Two-Class/Phase Case: As opposed to the GSD algorithm and Algorithm 2 (with $m = 2$), Algorithm 1 only requires evolving one pair of primal and dual variables. Accordingly, the computational cost can be halved, and the algorithm is expected to be more stable. Moreover, it can be viewed as very analogous to the Chambolle's dual algorithm [8] (cf. Remark 2.2). Indeed, we shall demonstrate that, for the time step size $\tau \leq 1/8$ (the theoretical prediction by [8]), the method works well, whereas the GSD algorithm is relatively restrictive to the time step size, as one will see from all the numerical results below. This is mainly due to the way of computing the primal variable [cf. (41) against (43)] and updating the dual variables, which is quite different between the two approaches. We further remark that Algorithms 1–2 are expected to have a performance similar to the Chambolle's method, but the convergence analysis is much more involved due to the strong nonlinearity (cf. Remark 2.2), which we wish to report in a future work.

Assume that the intensity values c_1 and c_2 are given and $f_i = |c_i - I|^2 (i = 1, 2)$, where I is the intensity of the input image as before. The parameters, time step sizes, and noise levels for three sets of tests are listed as follows. Recall that λ is the parameter in the continuous Potts model (3), and τ, β (resp. δ, ε) are involved in the Algorithms 1–2 (resp. the GSD algorithm). We first test the input "UOL" image of size 256×256 with a noise level from low to high. Notice that, in all tests, the noise is of "Gaussian" type with zero mean and different variance d. We adopt the following setup.

Set 1. Take $\lambda = 10^{-4}$, $\tau = \delta = 10^{-1}$, $\beta = 10^{-5}$, $\varepsilon = 10^{-2}$, and noise level: $d = 0.05$.

Set 2. Take $\lambda = 7 \times 10^{-5}$, $\tau = 0.1$, $\delta = 0.05$, $\beta = 10^{-5}$, $\varepsilon = 10^{-2}$, and noise level: $d = 0.3$.

Set 3. Take $\lambda = 5 \times 10^{-5}$, $\tau = 0.1$, $\delta = 0.01$, $\beta = 10^{-5}$, $\varepsilon = 0.05$, and noise level: $d = 0.5$.

We present in Fig. 2 the input images with noise and segmentation results by three algorithms at the iteration terminated by (53) with $\eta = 10^{-2}$. We observe from Fig. 2(a)–(d) that, when the noise level is low, three algorithms exhibit a similar performance and converge quite fast. However, when we increase the noise level, Algorithm 1 yields better segmentation [or classification, see the magnified portion of the letter **U** in Fig. 2(m)–(p)] and converges relatively faster. However, the time step size δ in the GSD algorithm should be chosen much smaller than τ in Algorithms 1 and 2. We also point out that the algorithm is not sensitive to the parameter β (refer to Section III-B4 below for a discussion). We plot the decay of numerical energy corresponding to the original energy functional (3), i.e., $E(\vec{u}^n)$ in Fig. 3 for the comparison tests: **Sets 1–3.** Observe that Algorithms 1–2 enjoy a faster decay of energy.

Multiphase Case: We now turn to the comparison of Algorithm 2 and the GSD algorithm for multiphase images. In this case, we take $f_i = |c_i - I|$ with $1 \leq i \leq 4$ for a given $\{c_i\}$ and test the four-phase image [cf. Fig. 4(a)] of size 90×90 with three typical geometric objects and different noise levels. We consider two sets of tests with the following setup.

Set 4. Take $\lambda = 0.20$, $\beta = 10^{-5}$, $\tau = 0.1$, $\delta = 0.05$, and $\varepsilon = 0.1$, and noise level: $d = 0.005$.

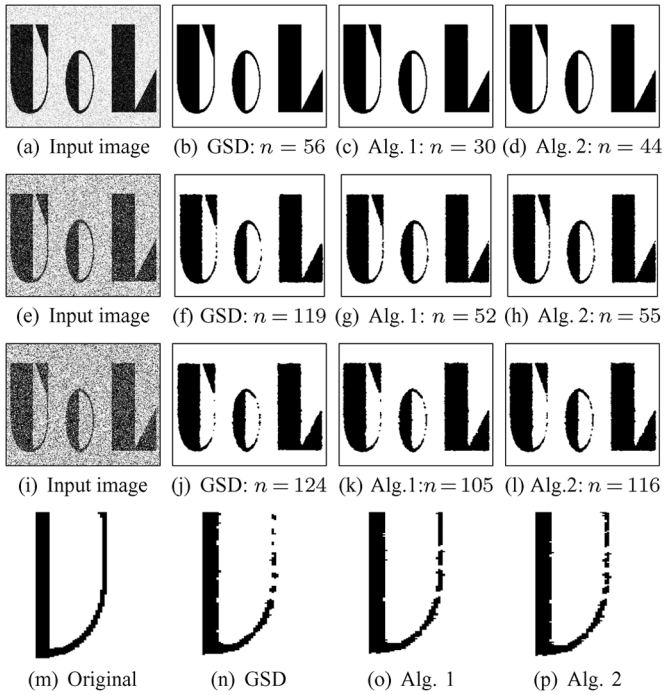


Fig. 2. Comparison of Algorithms 1–2 and the GSD method for the two-phase case. Row 1 (a)–(d) for **Set 1**, Row 2 (e)–(h) for **Set 2**, Row 3 (i)–(l) for **Set 3**, and Row 4 (m)–(p) the magnified portion of the images in Row 2. The iteration is stopped by (53) with $\eta = 10^{-2}$. The Algorithms 1–2 converge faster, allow use of larger time step size, and are not sensitive to the choice of the parameter β. (a) Input image. (b) GSD: $n = 56$. (c) Alg. 1: $n = 30$. (d) Alg. 2: $n = 44$. (e) Input image. (f) GSD: $n = 119$. (g) Alg. 1: $n = 52$. (h) Alg. 2: $n = 55$. (i) Input image. (j) GSD: $n = 124$. (k) Alg. 1: $n = 105$. (l) Alg. 2: $n = 116$. (m) Original. (n) GSD. (o) Alg. 1. (p) Alg. 2.

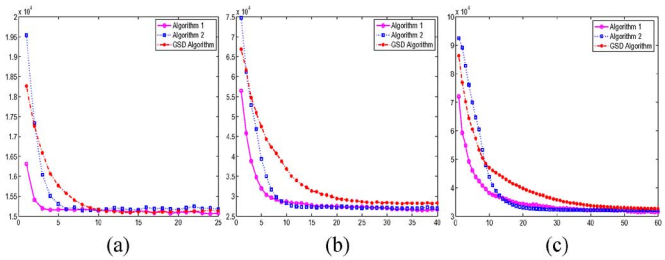


Fig. 3. Comparison of decay of numerical energy for **Sets 1–3**. (a) Energy decay (**Set 1**). (b) Energy decay (**Set 2**). (c) Energy decay (**Set 3**).

Set 5. Take $\lambda = 0.05$, $\beta = 0.01$, $\tau = 0.1$, $\delta = 0.05$, and $\varepsilon = 0.1$, and noise level: $d = 0.05$.

In Fig. 4, we plot the segmentation results by two algorithms with stopping rule $\eta = 5 \times 10^{-3}$. Once again, we observe the advantages of Algorithm 2. The comparison of numerical energy decay is depicted in Fig. 5, and we see a faster decay rate for the proposed algorithm. Indeed, we find that Algorithm 2 is robust for noise (in general, we choose the parameter β bigger if the noise level is high) and works for large time step size as the Chambolle's algorithm. In Table II, we tabulate the number of iterations (to meet the stopping rule (53) with $\eta = 5 \times 10^{-3}$), the computational time, and numerical energy at the last step of iteration, for two algorithms. In fact, the cost for per iteration of two algorithms is almost the same, but Algorithm 2 allows use of a large time step size; hence, it saves some computational time.

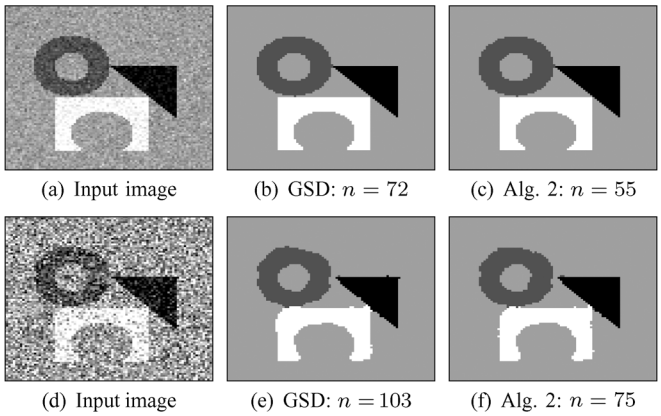


Fig. 4. Comparison of Algorithm 2 and the GSD algorithm for the four-phase case. Row 1 (a)–(c) for **Set 4** and Row 2 (d)–(f) for **Set 5**. The iteration is terminated by (53) with $\eta = 5 \times 10^{-3}$. Both algorithms work for large noise level. As before, Algorithm 2 converges faster and produces slightly better results with larger time step. (a) Input image. (b) GSD: $n = 72$. (c) Alg. 2: $n = 55$. (d) Input image. (e) GSD: $n = 103$. (f) Alg. 2: $n = 75$.

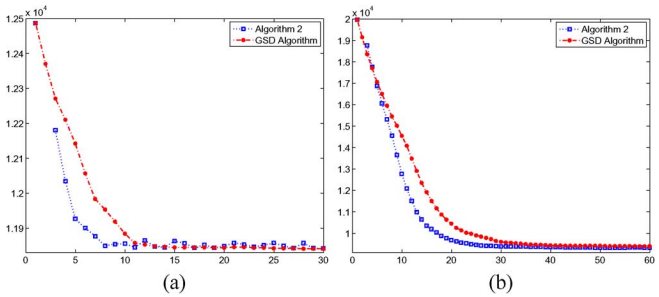


Fig. 5. Comparison of decay of numerical energy for test **Set 4** and **Set 5**. (a) Decay of energy (**Set 4**). (b) Decay of energy (**Set 5**).

TABLE II
COMPARISON OF ALGORITHM 2 AND GSD

	iterations		time (second)		energy	
	GSD	Alg. 2	GSD	Alg. 2	GSD	Alg. 2
Set 4	72	55	2.3	1.8	1.093×10^4	1.094×10^4
Set 5	103	75	3.6	2.6	7.177×10^3	7.163×10^3

We next test more multiphase images taken from the aforementioned papers and compare the quality of segmentation with the Alpha expansion and Alpha-Beta swap [3], [4] and the algorithm in Lellmann *et al.* [16]. Alpha expansion and Alpha-Beta swap have been regarded as the state of the art for approximately minimizing the discrete version of (2) with the anisotropic TV term. Indeed, due to the different nature of algorithms, it is hard to give a very quantified criterion for comparison; hence, we just depict the segmentation results in Fig. 6. We find from the first row in Fig. 6 that the Alpha method and our algorithm give better results than the other three. We also observe from the second row in Fig. 6 that the first three algorithms give similar results, whereas our algorithm produces the best segmentation. From the last row in Fig. 6, we see that our algorithms and GSD outperform the other methods.

Triple-Junction Experiments: Next, we test a very typical example relative to triple-junctions, which has been considered by many authors (see, e.g., [2], [9], [16], [23], [29], and [42]). The task is to inpaint the edges (or boundaries) of the subregion covered by the disk in Fig. 7(a) and (d). It is expected to

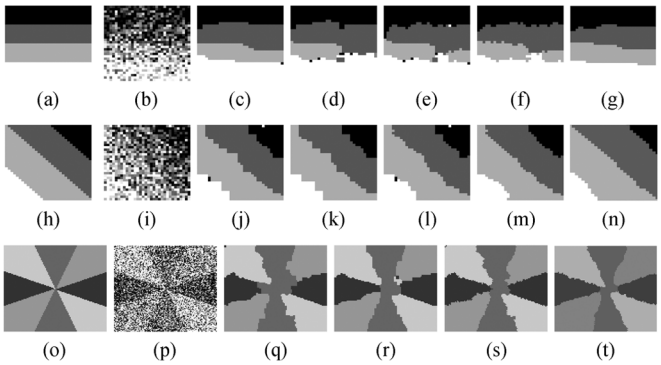


Fig. 6. Comparison of the Alpha, Alpha-Beta (Alp-Be), Lellmann's method (Lellm), GSD, and Algorithm 2. (a) Original. (b) Input. (c) Alpha. (d) Alp-Be. (e) Lellm. (f) GSD. (g) Alg. 2. (h) Original. (i) Input. (j) Alpha. (k) Alp-Be. (l) Lellm. (m) GSD. (n) Alg. 2. (o) Original. (p) Input. (q) Alpha. (r) Alp-Be. (s) GSD. (t) Alg. 2.

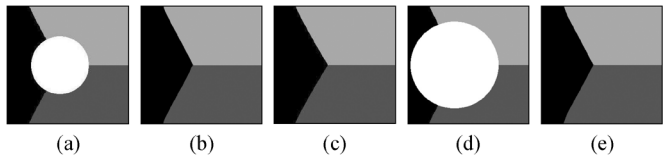


Fig. 7. Completion of edges. This shows the proposed algorithm can produce the global optimum. (a) Input. (b) Alg. 2: $n = 300$. (c) Result in [23]. (d) Input. (e) Alg. 2: $n = 600$.

generate a triple junction, that is, the completed three edges suppose to form three 120 ° angles at the junction. In the computation, the data terms f_i inside the disk are taken to be zero, and $f_i = |c_i - I|^2$ outside the disk, and the parameters in Algorithm 2 are chosen as $\lambda = 10^{-4}$, $\beta = 0.5$, and $\tau = 0.1$. It is important to point out that, in this case, the nonsmooth problem $\min_{1 \le i \le m}\{\text{div}\boldsymbol{p}_i^* + \lambda f_i\}$ may not have a unique minimum value (see the second case of Theorem 2.2) in the covered subregion; hence, in theory, the algorithm may fail to find the global optimum. However, by slightly increasing the diffusion effect tuned by the parameter β, we are able to obtain a very satisfactory completion of the edges inside the disk. This also indicates that the TV of the labeling functions (or level set functions) provides a good characterization of the edge set, and the algorithm produces very accurate approximation, even with a large portion of incomplete (or missing) edges [see Fig. 7(e)], namely, achieve the global optimum. We also quote the result in [23] (obtained by other means) and visualize a good agreement.

Selection of the Parameters and Fidelity Terms: Hereafter, we provide some insights into the selection of parameters β and λ, and the fidelity terms f_i. It is seen from the previous numerical tests that the parameter β in Algorithms 1–2 plays an important part for smoothing and regularizing the output images. A natural question is: what is the range of β for which the algorithms perform well? From our experiments, we have found that for two-phase cases, it can be chosen quite flexibly. In principle, the algorithms perform better if we choose larger β for a higher level of noise, particularly for multiphase cases. To further illustrate this, we test Algorithm 2 with different β on the images with two levels of noise in Fig. 8(a) and (g) and fix τ and other parameters. In all cases, the algorithm is terminated by (53) with $\eta = 5 \times 10^{-3}$. We record in Fig. 9 the history of decay of the

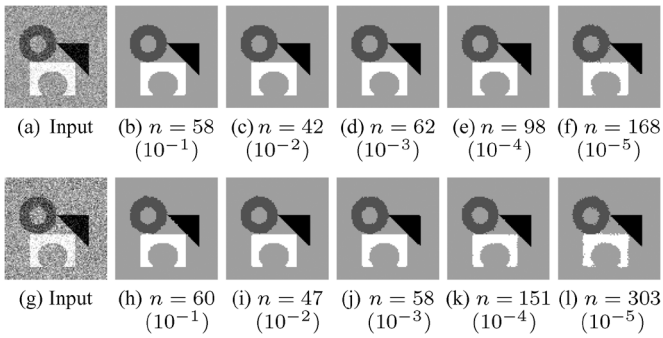


Fig. 8. Results obtained by Algorithm 2 with different β. In the computation, we take $\tau = 0.1$ and $f_i = |c_i - I|(i = 1, \ldots, 4)$. Row 1 (a)–(f): noise level $d = 0.01$ and $\lambda = 0.2$; Row 2 (g)–(l): noise level $d = 0.02$ and $\lambda = 0.1$. The iteration is terminated by (53) with $\eta = 5 \times 10^{-3}$, and we indicate the number of iterations and the values of β (in the parentheses). (a) Input. (b) $n = 58\ (10^{-1})$. (c) $n = 42\ (10^{-2})$. (d) $n = 62\ (10^{-3})$. (e) $n = 98\ (10^{-4})$. (f) $n = 168\ (10^{-5})$. (g) Input. (h) $n = 60\ (10^{-1})$. (i) $n = 47\ (10^{-2})$. (j) $n = 58\ (10^{-3})$. (k) $n = 151\ (10^{-4})$. (l) $n = 303\ (10^{-5})$.

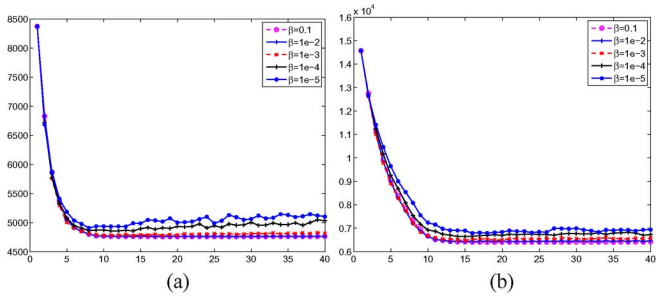


Fig. 9. Decay of numerical energy for different β. (a) Energy for Fig. 8(b)–(f). (b) Energy for Fig. 8(h)–(l).

energy and plot in Fig. 8 the segmentation results. We see that, for $\beta = 10^{-k}$ with $k = 1, 2, 3$, the algorithm produces similar results with a similar amount of iterations, whereas if β is too small, the convergence rate and quality of segmentation degrade.

We next discuss the sensitivity to the parameter λ, which is born with the model (3) with $f_i = |c_i - I|$. In Fig. 10(a)–(f), we fix the noise level d and the parameters τ, β but choose different λ ranging from 0.01 to 0.2, whereas in Fig. 10(g)–(l), we vary the noise level but fix $\lambda = 0.1$ in Algorithm 2. We see that Algorithm 2 enjoys a very similar performance in all tests. Indeed, the choice of λ is relatively flexible from our various experimental experiences. Some delicate analysis and adaptive determination of λ for the ROF-type model can be found in, e.g., [13] and [27], where λ is known as a scale variable. However, this requires significant computational cost to adapt λ.

Finally, we give some observations on the choice of the fidelity terms $f_i = |c_i - I|^p (p = 1, 2)$. We remark that some in-depth analysis for the TV-regularized L^p model can be found in, e.g., [5], [10], and [35]. In general, the quadratic case $(p = 2)$ is computationally easier to handle and more desirable for Gaussian noise removal, whereas the l_1-norm $(p = 1)$ is more capable of preserving contrast. In this context, $\{c_i\}$ are preassigned, and we realize that the use of $p = 1$ or 2 largely affects the choice of the parameter λ. Observe that

$$\lambda_2 \int_\Omega |c_i - I|^2 u_i d\boldsymbol{x} \sim \lambda_2 |c_i - I|_{\boldsymbol{\xi} \in \Omega} \int_\Omega |c_i - I| u_i d\boldsymbol{x},$$

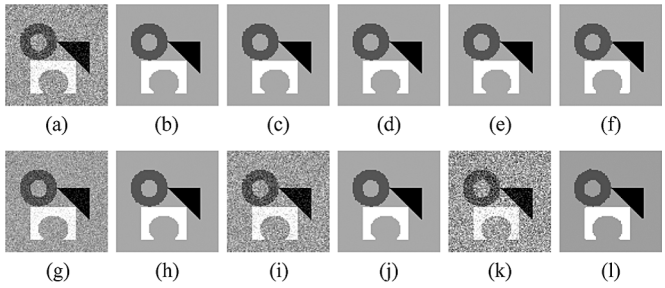


Fig. 10. Results obtained by Algorithm 2 with fixed $\tau = 0.1$, $\beta = 0.01$. Row 1: noise level $d = 0.1$ for all tests, but different λ. Row 2: $\lambda = 0.1$ for all tests, but noise level $d = 0.005$, 0.1, and 0.2 from left to right. The iteration is terminated by (53) with $\eta = 5 \times 10^{-3}$. (a) Input. (b) $n = 42$ $(\lambda = 0.2)$. (c) $n = 41$ $(\lambda = 0.1)$. (d) $n = 43$ $(\lambda = 0.05)$. (e) $n = 40$ $(\lambda = 0.02)$. (f) $n = 38$ $(\lambda = 0.01)$. (g) Input. (h) $n = 39$. (i) Input. (j) $n = 41$. (k) Input. (l) $n = 47$.

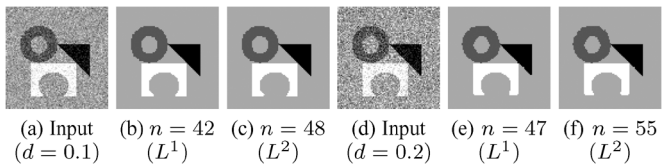


Fig. 11. Results obtained by Algorithm 2 with different $f_i = |c_i - I|, |c_i - I|^2$, and $\lambda_1 = 200\lambda_2$. Here, $\tau = 0.1$, $\beta = 0.01$. (a)–(c): $\lambda_1 = 0.2$; (d)–(f): $\lambda_1 = 0.1$. The iteration is terminated by (53) with $\eta = 5 \times 10^{-3}$. (a) Input $(d = 0.1)$. (b) $n = 42$ (L^1). (c) $n = 48$ (L^2). (d) Input $(d = 0.2)$. (e) $n = 47$ (L^1). (f) $n = 55$ (L^2).

where, for clarity, we set $\lambda = \lambda_p$ for $p = 1, 2$. For piecewise constant image I, we can empirically choose $\lambda_1 = \max_{i \neq j} |c_i - c_j| \lambda_2$. In Fig. 11, we set $\lambda_1 = 200\lambda_2$ and see that Algorithm 2 for two models work almost equally well.

Applications to Image Segmentation: Finally, we apply the methods to the PCMSM (1), for which $\{c_i\}$ are unknowns computed from the mean values

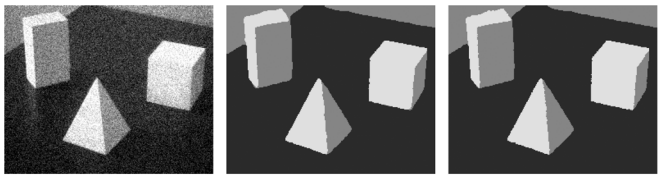
$$c_i = \frac{\int_\Omega I u_i d\boldsymbol{x}}{\int_\Omega u_i d\boldsymbol{x}}, \quad 1 \leq i \leq m. \tag{54}$$

For fixed c_i, the model (1) is a special case of (2) with $f_i = |c_i - I|^2$. Therefore, with a slight modification of Algorithm 2, we can obtain the following algorithm for the multiphase piecewise constant image segmentation.

Algorithm 3

1. Initialization: set $\vec{\boldsymbol{\lambda}}_1^0 = \mathbf{0}$ and choose β, τ, $\lambda > 0$, and $\vec{c}^0$;
2. For $n = 0, 1, \ldots$
   - (i) Compute λ_3^n by (40) with $f_i^n = |c_i^n - I|^2$ in place of f_i;
   - (ii) Compute $\vec{u}^n$ by (41) with $h_i = \mathrm{div}\boldsymbol{\lambda}_{1i}^n + \lambda f_i^n$ and $\lambda_3 = \lambda_3^n$;
   - (iii) Compute $\vec{\boldsymbol{\lambda}}_1^{n+1}$ by

$$\boldsymbol{\lambda}_{1i}^{n+1} = \frac{\boldsymbol{\lambda}_{1i}^n - \tau \nabla u_i^n}{1 + \tau |\nabla u_i^n|}, \quad 1 \leq i \leq m;$$



(a) Input image (b) GSD-Seg: $n = 115$ (c) Alg. 3: $n = 87$

Fig. 12. Comparison of Algorithm 3 and GSD-Seg, where $\tau = 0.1, \delta = 0.02$, and $f_i = |c_i - I|^2$ and other parameters: (a)–(c) (three phases of size 205×255): $\lambda = 5 \times 10^{-4}$, $\beta = 10^{-2}$, and $\varepsilon = 0.1$. Given this noise input image, it is challenging to segment the faces of the objects. The iteration of both algorithms is terminated by (53) with $\eta = 5 \times 10^{-3}$. Algorithm 3 slightly outperforms GSD-Seg. (a) Input image. (b) GSD-Seg: $n = 115$. (c) Alg. 3: $n = 87$.

TABLE III
COMPARISON OF ALGORITHM 3 AND GSD-SEG

iterations		time (second)		energy	
GSD-Seg	Alg. 3	GSD-Seg	Alg. 3	GSD-Seg	Alg. 3
115	87	15	11	1.320×10^4	1.314×10^4

(iv) Compute $\vec{c}^{n+1}$ by

$$c_i^{n+1} = \frac{\int_\Omega I u_i^n d\boldsymbol{x}}{\int_\Omega u_i^n d\boldsymbol{x}}, \quad 1 \le i \le m;$$

3. Endfor until some stopping rule meets;
4. Set $\vec{\boldsymbol{\lambda}}_1^* = \vec{\boldsymbol{\lambda}}_1^{n+1}$, $\vec{c}^* = \vec{c}^{n+1}$ and define

$$u_k^* = \begin{cases} 1, & k = \min\left\{\arg\min_{1 \le i \le m} \left\{\mathrm{div}\boldsymbol{\lambda}_{1i}^* + \lambda f_i^*\right\}\right\}, \\ 0, & \text{otherwise}, \end{cases}$$

for all $1 \le k \le m$, where $f_i^* = |c_i^* - I|^2$.

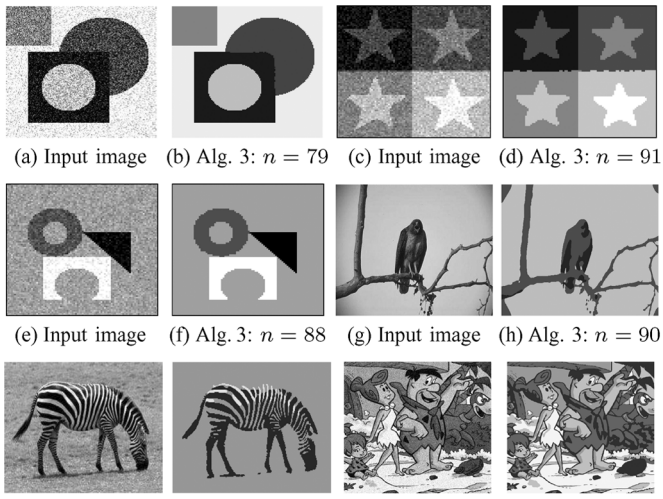
Similarly, the GSD algorithm can be modified for the segmentation problem (see [2, Algorithm 2]), and for clarity, we term it as GSD-Seg in short.

We first compare Algorithm 3 with GSD-Seg for segmenting the "desk" image in Fig. 12(a) of size 205×255 containing three typical geometric objects with three phases. For this example, it is not an easy task to segment the faces of the objects, for instance, the two faces of the tetrahedron. In the test, both algorithms are terminated by (53) with $\eta = 5 \times 10^{-3}$ (refer to the caption of Fig. 12 for the selection of parameters and time step size). As before, both algorithms produce quite satisfactory and accurate segmentation, whereas Algorithm 3 converges relatively faster. We also tabulate in Table III the number of iterations (to meet the stopping rule (53) with $\eta = 5 \times 10^{-3}$), the computational time, and numerical energy at the last step of iteration, for two algorithms. Once again, the cost for per iteration of two algorithms is almost the same, but Algorithm 3 allows use of a large time step size.

We next test some more images that are used as examples in most of the papers that we mentioned before. We refer to the caption of Fig. 13 for the choice of the parameters in Algorithm 3. In all cases, the algorithm is terminated by (53) with $\eta = 5 \times 10^{-3}$. We see that, in all tests, Algorithm 3 provides quite satisfactory segmentation and converges fast.

IV. CONCLUDING REMARKS

In this paper, we proposed a direct primal-dual approach toward global minimization of the continuous Potts model for multiclass labeling problems with applications to multiphase image segmentation. Different from the existing works, the underlying analysis and algorithms were based on a binary setting that did not require convex relaxation. Using the augmented Lagrangian technique, we were able to find the relations between the primal and dual variables that involve almost the minimum number of parameters. The proposed algorithms could be viewed as the counterparts of the Chambolle's algorithm in the context of labeling and segmentation. Indeed, the time step size could be taken as large as that of the Chambolle's algorithm, and the computational cost turned out to be almost of the same amount. Moreover, the smoothing parameter involved in the algorithm was quite flexible to choose. Various numerical results demonstrated the advantages of the methods over the existing approaches.



(a) Input image (b) Alg. 3: $n = 79$ (c) Input image (d) Alg. 3: $n = 91$

(e) Input image (f) Alg. 3: $n = 88$ (g) Input image (h) Alg. 3: $n = 90$

(i) Input image (j) Alg. 3: $n = 76$ (k) Input image (l) Alg. 3: $n = 89$

Fig. 13. More tests on Algorithm 3, where $\tau = 0.1$, $f_i = |c_i - I|^2$ and other parameters: (a)–(b) (five phases): $\lambda = 10^{-4}$, $\beta = 10^{-3}$, size: 150×150; (c)–(d) (five phases): $\lambda = 10^{-3}$, $\beta = 10^{-2}$, size: 90×90; (e)–(f) (four phases): $\lambda = 10^{-3}$, $\beta = 10^{-5}$, size: 90×90; (g)–(h) (three phases): $\lambda = 10^{-3}$, $\beta = 10^{-2}$, size: 321×481; (i)–(j) (three phases): $\lambda = 10^{-3}$, $\beta = 10^{-2}$, size: 167×250; and (k)–(l) (five phases): $\lambda = 10^{-3}$, $\beta = 10^{-2}$, size: 512×512. The iteration is terminated by (53) with $\eta = 5 \times 10^{-3}$. (a) Input image. (b) Alg. 3: $n = 79$. (c) Input image. (d) Alg. 3: $n = 91$. (e) Input image. (f) Alg. 3: $n = 88$. (g) Input image. (h) Alg. 3: $n = 90$. (i) Input image. (j) Alg. 3: $n = 76$. (k) Input image. (l) Alg. 3: $n = 89$.

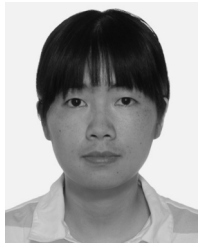
ACKNOWLEDGMENT

The authors would like to thank the editor and anonymous referees for their valuable comments and suggestions.

REFERENCES

[1] S. M. Allen and J. W. Cahn, "A microscope theory for antiphase boundary motion and its application to antiphase domain coarsening," *Acta Metall.*, vol. 27, no. 6, pp. 1085–1095, Jun. 1979.
[2] E. Bae, J. Yuan, and X. C. Tai, "Global minimization for continuous multiphase partitioning problems using a dual approach," *Int. J. Comput. Vis.*, vol. 92, no. 1, pp. 112–129, Mar. 2011.
[3] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1124–1137, Sep. 2004.
[4] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1222–1239, Nov. 2001.
[5] X. Bresson, S. Esedoglu, P. Vandergheynst, J. P. Thiran, and S. Osher, "Fast global minimization of the active contour/snake model," *J. Math. Imag. Vis.*, vol. 28, no. 2, pp. 151–167, Jun. 2007.

[6] E. S. Brown, T. F. Chan, and X. Bresson, Convex formulation and exact global solutions for multi-phase piecewise constant Mumford–Shah image segmentation UCLA, Applied Mathematics, Los Angeles, CA, CAM-report-09-66, Jul. 2009.
[7] E. S. Brown, T. F. Chan, and X. Bresson, A convex relaxation method for a class of vector-valued minimization problems with applications to Mumford–Shah image segmentation UCLA, Applied Mathematics, Los Angeles, CA, CAM-report-10-43, Jul. 2010.
[8] A. Chambolle, "An algorithm for total variation minimization and applications," *J. Math. Imag. Vis.*, vol. 20, no. 1, pp. 89–97, Jan. 2004.
[9] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *J. Math. Imag. Vis.*, vol. 40, no. 1, pp. 120–145, May 2011.
[10] T. F. Chan and S. Esedoglu, "Aspects of total variation regularized L^1 function approximation," *SIAM J. Appl. Math.*, vol. 65, no. 5, pp. 1817–1837, May–Jul. 2005.
[11] T. F. Chan and J. H. Shen, "Image processing and analysis," in *Variational, PDE, Wavelet, and Stochastic Methods*. Philadelphia, PA: SIAM, 2005.
[12] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Trans. Image Process.*, vol. 10, no. 2, pp. 266–277, Feb. 2001.
[13] Y. Q. Dong and M. Hintermuller, "Multi-scale total variation with automated regularization parameter selection for color image restoration," in *Scale Space and Variational Methods in Computer Vision*. Berlin, Germany: Springer-Verlag, 2009, pp. 271–281.
[14] L. C. Evans and R. F. Gariepy, "Measure theory and fine properties of functions," in *Studies in Advanced Mathematics*. Boca Raton, FL: CRC Press, 1992.
[15] J. B. Hiriart-Urruty and C. Lemaréchal, *Convex Analysis and Minimization Algorithms: Fundamentals*. New York: Springer-Verlag, 1993.
[16] J. Lellmann, J. Kappes, J. Yuan, F. Becker, and C. Schnörr, Convex multi-class image labeling by simplex-constrainted total variation HCI, Heidelberg, Germany, Tech. Rep., 2008.
[17] J. Lellmann and C. Schnörr, Continuous multiclass labeling approaches and algorithms Univ. Heidelberg, Heidelberg, Germany, Tech. Rep., 2010.
[18] C. Li, C. Xu, C. Gui, and M. D. Fox, "Level set evolution without re-initialization: A new variational formulation," in *Proc IEEE Conf. CVPR*, 2005, pp. 430–436.
[19] F. Li and M. K. Ng, "Kernel density estimation based multiphase fuzzy region competition method for texture image segmentation," *Commun. Comput. Phys.*, vol. 8, no. 3, pp. 623–641, 2010.
[20] F. Li, M. K. Ng, T. Y. Zeng, and L. C. Shen, "A multiphase image segmentation method based on fuzzy region competition," *SIAM J. Imag. Sci.*, vol. 3, no. 3, pp. 277–299, 2010.
[21] J. Lie, M. Lysaker, and X. C. Tai, "A variant of the level set method and applications to image segmentation," *Math. Comput.*, vol. 75, no. 255, pp. 1155–1174, Jul. 2006.
[22] C. X. Liu, F. F. Dong, S. F. Zhu, D. K. Kong, and K. F. Liu, "New variational formulations for level set evolution without reinitialization with applications to image segmentation," *J. Math. Imag. Vis.*, vol. 41, no. 3, pp. 194–209, Nov. 2011, 10.1007/s10851-011-0269-2.
[23] J. Liu, X. C. Tai, H. Y. Huang, and Z. D. Huan, "A fast segmentation method based on constraint optimization and its applications: Intensity inhomogeneity and texture segmentation," *Pattern Recognit.*, vol. 44, no. 9, pp. 2093–2108, Sep. 2011.
[24] B. Merriman, J. K. Bence, and S. Osher, "Motion of multiple functions: A level set approach," *J. Comput. Phys.*, vol. 112, no. 2, pp. 334–363, Jun. 1994.
[25] D. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Commun. Pure Appl. Math.*, vol. 42, no. 5, pp. 577–685, Jul. 1989.
[26] M. Nikolova, T. F. Chan, and S. Esedoglu, "Algorithms for finding global minimizers of image segmentation and denoising models," *SIAM J. Appl. Math.*, vol. 66, no. 5, pp. 1632–1648, 2006.
[27] S. Osher and R. P. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*. New York: Springer-Verlag, 2002.
[28] S. Osher and R. P. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*. New York: Springer-Verlag, 2003.
[29] T. Pock, A. Chambolle, H. Bischof, and D. Cremers, "A convex relaxation approach for computing minimal partitions," in *Proc. IEEE Conf. CVPR*, 2009, pp. 810–817.
[30] T. Pock, D. Cremers, H. Bischof, and A. Chambolle, "Global solutions of variational models with convex regularization," *SIAM J. Imag. Sci.*, vol. 3, no. 4, pp. 1122–1145, 2010.
[31] T. Pock, T. Schoenemann, G. Graber, H. Bischof, and D. Cremers, "A convex formulation of continuous multi-label problems," in *Proc. ECCV*, 2008, pp. 792–805.
[32] R. B. Potts, "Some generalized order–disorder transformations," *Proc. Cambridge Philos. Soc.*, vol. 48, no. 1, pp. 106–109, 1952.
[33] Z. J. Rong, L. L. Wang, and X. C. Tai, "Adaptive wavelet collocation methods for image segmentation using TV-Allen–Cahn type models," *Adv. Comput. Math.*, 2011, to be published, to be published.
[34] L. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D*, vol. 60, no. 1–4, pp. 259–268, Nov. 1992.
[35] Y. Y. Shi, L. L. Wang, and X. C. Tai, "Geometry of total variation regularized L^p-model," *J. Comput. Appl. Math.*, vol. 236, no. 8, pp. 2223–2234, Feb. 2012, 10.1016/j.cam.2011.09.043.
[36] X. C. Tai, O. Christiansen, P. Lin, and I. Skjlaaen, "Image segmentation using some piecewise constant level set methods with mbo type of projection," *Int. J. Comput. Vis.*, vol. 73, no. 1, pp. 61–76, Jun. 2007.
[37] X. C. Tai and C. Wu, "Augmented Lagrangian method, dual methods and split Bregman iteration for ROF model," in *Scale Space and Variational Methods in Computer Vision*. Berlin, Germany: Springer-Verlag, 2009, pp. 502–514.
[38] L. A. Vese and T. F. Chan, "A multiphase level set framework for image segmentation using the Mumford and Shah model," *Int. J. Comput. Vis.*, vol. 50, no. 3, pp. 271–293, Dec. 2002.
[39] L. L. Wang and Y. Gu, "Efficient dual algortihms for image segmentation using TV-Allen–Cahn type models," *Commun. Comput. Phys.*, vol. 9, no. 4, pp. 859–877, Apr. 2011.
[40] Y. L. Wang, J. F. Yang, W. T. Yin, and Y. Zhang, "A new alternating minimization algorithm for total variation image reconstruction," *SIAM J. Imag. Sci.*, vol. 1, no. 3, pp. 248–272, Jul. 2008.
[41] C. Wu and X. C. Tai, "Augmented lagrangian method, dual methods, and split Bregman iteration for ROF, vectorial TV, and high order models," *SIAM J. Imag. Sci.*, vol. 3, no. 3, pp. 300–339, Jul. 2010.
[42] C. Zach, D. Gallup, J. Frahm, and M. Niethammer, "Fast global labeling for real-time stereo using multiple plane sweeps," in *Proc. VMV Workshop*, 2008, pp. 243–252.



Ying Gu received the B.S. degree in mathematics and applied mathematics from Soochow University, Jiangsu, China, in 2008. She is currently working toward the Ph.D. degree at Nanyang Technological University, Singapore.

Her research interests include image processing based on variational techniques and partial differential equations.

Li-Lian Wang received the Ph.D. degree in computational mathematics from Shanghai University, Shanghai, China, in 2000.

He is currently an Associate Professor with the Division of Mathematical Sciences at the School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore. His research interests include numerical methods for partial differential equations and image processing based on variational techniques and partial differential equations.

Xue-Cheng Tai received the licenciate degree and the Ph.D. degree in applied mathematics from Jyvaskalya University, Jyvaskyla, Finland, in 1989 and 1991, respectively.

His Ph.D. thesis was on inverse problems and parallel computing. After holding several research positions in Europe, he became an Associate Professor, in 1994, at the University of Bergen, Bergen, Norway, and a Professor, in 1997. He has been also a part-time Senior Scientist at a private company "Rogaland Research" and "Christian Michelsen Research." He is currently a member of the "Center for Mathematics for Applications" in Oslo and a member of the "Center of Integrated Petroleum Research" in Bergen. Since 2007, he has been also with Nanyang Technological University, Singapore. His research interests include numerical partial differential equation (PDE) for image processing, multigrid and domain decomposition methods, iterative methods for linear and nonlinear PDE problems, and parallel computing. He has guided numerous master's and Ph.D. students and published more than 80 scientific papers. He has been a Reviewer and an Editor for a number of international journals.