

A Two-Stage Image Segmentation Method Using a Convex Variant of the Mumford–Shah Model and Thresholding*

Xiaohao Cai[†], Raymond Chan[†], and Tiejong Zeng[‡]

Abstract. The Mumford–Shah model is one of the most important image segmentation models and has been studied extensively in the last twenty years. In this paper, we propose a two-stage segmentation method based on the Mumford–Shah model. The first stage of our method is to find a smooth solution g to a convex variant of the Mumford–Shah model. Once g is obtained, then in the second stage the segmentation is done by thresholding g into different phases. The thresholds can be given by the users or can be obtained automatically using any clustering methods. Because of the convexity of the model, g can be solved efficiently by techniques like the split-Bregman algorithm or the Chambolle–Pock method. We prove that our method is convergent and that the solution g is always unique. In our method, there is no need to specify the number of segments K ($K \geq 2$) before finding g . We can obtain any K -phase segmentations by choosing $(K - 1)$ thresholds after g is found in the first stage, and in the second stage there is no need to recompute g if the thresholds are changed to reveal different segmentation features in the image. Experimental results show that our two-stage method performs better than many standard two-phase or multiphase segmentation methods for very general images, including antimass, tubular, MRI, noisy, and blurry images.

Key words. image segmentation, Mumford–Shah model, split-Bregman, total variation

AMS subject classifications. 52A41, 65D15, 68W40, 90C25, 90C90

DOI. 10.1137/120867068

1. Introduction. Let $\Omega \subset \mathbb{R}^2$ be a bounded open connected set, Γ be a compact curve in Ω , and $f : \Omega \rightarrow \mathbb{R}$ be a given image. Without loss of generality, we restrict the range of f to $[0,1]$, and hence $f \in L^\infty(\Omega)$. In [42, 43], Mumford and Shah proposed an energy minimization problem which approximates the true solution by finding optimal piecewise smooth approximations. More precisely, the energy minimization problem was formulated in [43] as

$$(1.1) \quad E_{\text{MS}}(g, \Gamma) = \frac{\lambda}{2} \int_{\Omega} (f - g)^2 dx + \frac{\mu}{2} \int_{\Omega \setminus \Gamma} |\nabla g|^2 dx + \text{Length}(\Gamma),$$

where λ and μ are positive parameters and $g : \Omega \rightarrow \mathbb{R}$ is continuous or even differentiable in $\Omega \setminus \Gamma$ but may be discontinuous across Γ . Here, the length of Γ can be written as $\mathcal{H}^1(\Gamma)$, the 1-dimensional Hausdorff measure in \mathbb{R}^2 ; see [4]. Because model (1.1) is nonconvex, it is very challenging to find or approximate its minimizer.

*Received by the editors February 22, 2012; accepted for publication (in revised form) October 16, 2012; published electronically February 19, 2013.

<http://www.siam.org/journals/siims/6-1/86706.html>

[†]Department of Mathematics, The Chinese University of Hong Kong, Shatin, Hong Kong (xhcai@math.cuhk.edu.hk, rchan@math.cuhk.edu.hk). The second author was partially supported by RGC 400412 and DAG 2060408.

[‡]Department of Mathematics, Hong Kong Baptist University, Kowloon Tong, Hong Kong (zeng@hkbu.edu.hk). This author was partially supported by NSFC 11271049, RGC 211710, RGC 211911, and RFGs of HKBU.

In [1, 2], the Mumford–Shah energy (1.1) was approximated by a sequence of simpler elliptic variational problems where the length of Γ was replaced by a phase field energy. Later, nonlocal approximation of (1.1) was proposed in [11, 12, 25, 41]. By using a family of continuous and nondecreasing functions, they avoid computing Γ explicitly. In particular, their methods solve an anisotropic variant of the Mumford–Shah model (1.1). In [10], numerical approaches based on a discrete functional were considered for solving (1.1). Recently, a novel primal-dual algorithm based on a convex representation of (1.1) was proposed. It can solve (1.1) accurately. However, for a 128×128 image, it requires 600 seconds on a Tesla C1060 GPU machine. Until now, the bottleneck of solving (1.1) has still been that the model itself is nonconvex.

Over the years, people have tried to simplify the model (1.1). For example, if we restrict $\nabla g \equiv 0$ on $\Omega \setminus \Gamma$, then it results in a piecewise constant Mumford–Shah model. In [16], the method of active contours without edges (Chan–Vese model) was introduced. It solves the piecewise constant Mumford–Shah model but restricts the solution to be a piecewise constant solution with only two constants. For works on the general piecewise constant Mumford–Shah model, see [31, 49, 50], etc. These methods work well for certain image segmentation tasks, for example for cartoon images. However, the main drawback of these methods is that they can easily get stuck in local minima. In order to overcome the problem, convex relaxation approaches [7, 14, 45] and the graph cut method [27] were proposed. There are also many other models related to the Chan–Vese model [16, 49], for example, the two-phase segmentation algorithms in [19, 52, 53] and multiphase segmentation algorithms in [3, 8, 21, 32, 33, 34, 35, 47, 48, 54]. Specifically, in [33], the piecewise constant Mumford–Shah model was convexified by using fuzzy membership functions. In [47], a new regularization term was introduced which allows choosing the number of phases automatically. In [52, 53, 54], efficient methods based on the fast continuous max-flow method were proposed. In [19], the length term was replaced by a term involving framelets. In [32], the continuous multiclass labeling approaches were discussed. Interested readers can read the references therein or see [4] for more details.

In this paper, we separate the task of segmentation into two stages. The first stage is to find a smooth image g that can facilitate the segmentation, and the second stage is to threshold g to reveal different segmentation features. To find g , instead of tackling the challenging problem of solving the Mumford–Shah model (1.1), we propose to use the model

$$(1.2) \quad \inf_g \left\{ \frac{\lambda}{2} \int_{\Omega} (f - \mathcal{A}g)^2 dx + \frac{\mu}{2} \int_{\Omega} |\nabla g|^2 dx + \int_{\Omega} |\nabla g| dx \right\},$$

where \mathcal{A} can be the identity operator (for noisy observed image f) or a blurring operator (if there are noise and blur in f). We will see that our model (1.2) is closely related to (1.1) and is convex with a unique smooth solution. In the first stage of our two-stage method, we solve (1.2). Once g is found, then in the second stage, the segmentation is obtained by segmenting g using properly chosen threshold(s). To segment g into K segments, $K \geq 2$, we require $(K - 1)$ thresholds which the users can provide themselves or obtain automatically by any clustering methods such as the K-means methods [28, 38] or the convex K-means method named SON clustering [36]. Figure 1 shows two multiphase segmentation results from our method using thresholds from MATLAB K-mean command KMEANS on g .

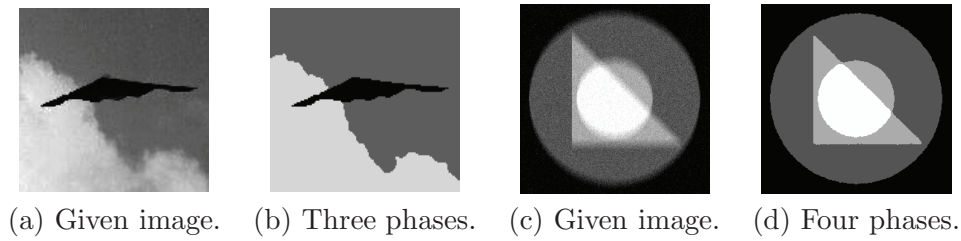


Figure 1. *Multiphase segmentation results given by our method.*

We will prove that, under mild conditions, our model (1.2) has one and only one solution g which can be solved very quickly by popular algorithms such as the split-Bregman algorithm [26] or the Chambolle–Pock method [13, 44]. One nice aspect of our method is that there is no need to recompute g if we have to change the thresholds in the second stage to reveal different features in the image. Another nice aspect is that there is no need to specify K in the first stage, i.e., before finding g . We can obtain any K -phase segmentation ($K \geq 2$) by choosing $(K - 1)$ thresholds after g is computed in the first stage. In contrast, multiphase methods such as those in [3, 8, 21, 32, 33, 34, 35, 45, 48, 54] require K to be given first; and, if K changes, the minimization problem has to be solved again.

Our tests in section 4 show that our method can segment different kinds of images: anti-mass image, tubular MRA image, brain MRI image, image with very high noise, and image with blur and noise. For the blur and noisy image, all the multiphase methods we tested [33, 47, 54, 3, 45] fail, while our method can provide a very good result; see Figures 1(c)–(d) or 10. We will see that our method is fast compared to popular two-phase segmentation methods [16, 19, 53] and multiphase segmentation methods [33, 47, 54, 3, 45].

Note that once g is obtained and the thresholds are given, the segmenting of g into K segments requires very little time. In fact, the complexity is proportional to the number of pixels in the image. Hence our method is quite suitable for users to play around with different thresholds to determine the number of segments they prefer and to reveal the different features within the image. However, we can also use the MATLAB-provided K-means method to compute the thresholds automatically for users who prefer an automated K -phase segmentation algorithm.

Our model provides a better understanding of the link between image segmentation and image restoration. Indeed, the effectiveness of our method suggests that for segmentation a key idea is to extract the cartoon part in the image, i.e., g , and then cluster g into different phases. Based on this two-stage idea, it is likely that more efficient segmentation methods can be developed in the future along this line. As pointed out by one of the reviewers, Esedoğlu and Tsai also proposed a two-stage approach in [21]—one that also uses smoothing followed by thresholding. Indeed, [21] proposed an extremely efficient PDE-based algorithm for minimizing the piecewise constant Mumford–Shah segmentation model. Their algorithm was inspired by the work of Merriman, Bence, and Osher (MBO) on diffusion generated motion by curvature [39, 40]. Similar to the MBO algorithm, each iteration of [21] requires a solution of a linear diffusion equation and then a thresholding. The main difference between our method and the approach in [21] is clear: in [21], the smoothing and thresholding are

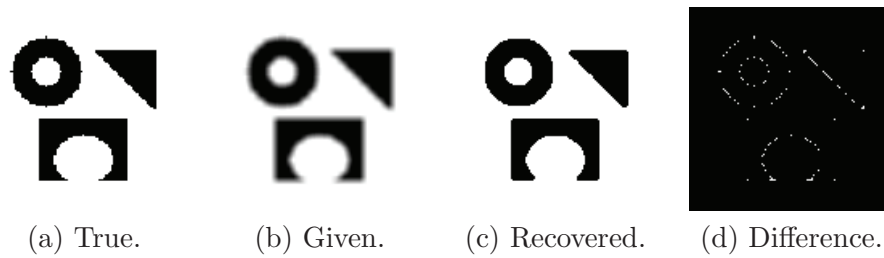


Figure 2. Segmentation from smooth image. (a) True 128×128 binary image; (b) given smoothed image of (a) by a Gaussian filter; (c) segmented binary result from (b) using threshold 0.5; (d) the difference image between (a) and (c), where nonzero pixel values are scaled to 1 to reveal them clearly.

done alternatively for a number of iterations, while in our two-stage method, the smoothing and the thresholding are done only once: smoothing in the first stage and thresholding in the second stage. For more details along the direction of smoothing and filtering, see [21, 39, 40] and references therein.

The rest of the paper is organized as follows. In section 2, we derive our convex model (1.2) which is based on the Mumford–Shah model. We then show that our model has a unique solution. In section 3, we give the detailed implementation of our method and show that the resulting algorithm converges. In section 4, we compare our method on various synthetic and real images with three two-phase segmentation algorithms [19, 16, 53] and five multiphase segmentation methods [33, 47, 54, 3, 45]. The relationship between our model and models in image restoration is discussed in section 5. Conclusions are given in section 6.

2. Two-stage method. Our model is motivated by the following simple but important observation about binary images: a binary image can be recovered quite well from its smoothed version by thresholding with a proper threshold. Figure 2 is an example to illustrate our point. Figure 2(a) is the true binary image, and 2(b) is its smoothed version obtained by a Gaussian filter with size [5,5] and standard deviation 3. Obviously, pixel values near the boundary are smoothed. However, by using a threshold of 0.5 to threshold Figure 2(b) back to a binary image, we obtain Figure 2(c). We see that all the pixels of Figure 2(a) except some on the boundary are correctly recovered; see the difference image in Figure 2(d). Inspired by this idea, we will modify model (1.1) step by step to arrive at our model (1.2). Briefly, our method consists of two stages. In the first stage, we will find the smooth minimizer of (1.2); then in the second stage, we apply a simple thresholding strategy to carry out the segmentation. In the following, we derive our model (1.2).

Assume that Γ is a Jordan curve. Let $\Sigma = \overline{\text{Inside}(\Gamma)}$; then $\Gamma = \partial\Sigma$. Model (1.1) can be written as

$$(2.1) \quad \begin{aligned} \tilde{E}(\Sigma, g_1, g_2) := & \frac{\lambda}{2} \int_{\Sigma \setminus \Gamma} (f - g_1)^2 dx + \frac{\mu}{2} \int_{\Sigma \setminus \Gamma} |\nabla g_1|^2 dx + \frac{\lambda}{2} \int_{\Omega \setminus \Sigma} (f - g_2)^2 dx \\ & + \frac{\mu}{2} \int_{\Omega \setminus \Sigma} |\nabla g_2|^2 dx + \text{Per}(\Sigma), \end{aligned}$$

where g_1 and g_2 are defined on $\Sigma \setminus \Gamma$ and $\Omega \setminus \Sigma$, respectively, and $\text{Per}(\cdot)$ denotes the perimeter of Σ ; i.e., $\text{Per}(\Sigma) = \text{Length}(\Gamma)$. Note that (2.1) is similar to (9) in [14]. Observe that once Σ

is fixed, then g_1 and g_2 are determined by the following two minimization problems:

$$(2.2) \quad \inf_{g_1 \in W^{1,2}(\Sigma \setminus \Gamma)} \left\{ \lambda \int_{\Sigma \setminus \Gamma} (f - g_1)^2 dx + \mu \int_{\Sigma \setminus \Gamma} |\nabla g_1|^2 dx \right\}$$

and

$$(2.3) \quad \inf_{g_2 \in W^{1,2}(\Omega \setminus \Sigma)} \left\{ \lambda \int_{\Omega \setminus \Sigma} (f - g_2)^2 dx + \mu \int_{\Omega \setminus \Sigma} |\nabla g_2|^2 dx \right\}.$$

For the definition of $W^{1,2}(\Omega)$, see [22, Chapter 5]. The existence and uniqueness of the solutions g_1 and g_2 are guaranteed by the following proposition.

Proposition 2.1. *Let $f \in L^2(\Omega)$. Then the two minimization problems (2.2) and (2.3) have unique minimizers.*

Proof. Since Σ is closed, both the sets $\Omega \setminus \Sigma$ and $\Sigma \setminus \Gamma$ are open. Using the conclusions of Proposition 1 in [4] or Proposition 3 in [18], we conclude that problems (2.2) and (2.3) have unique minimizers. ■

From the analysis above we can conclude that once the boundary Γ is fixed, i.e., Σ is fixed, then g_1 and g_2 are determined uniquely. Note that in [14], the Chan–Vese model is made convex once the mean values of f inside and outside of Γ are fixed. Here, motivated by Theorem 2 of [14], we can derive and prove the following similar theorem for the model (2.1) once g_1 and g_2 are fixed and smoothly extended to the whole Ω .

Theorem 2.2. *For any given fixed functions g_1 and $g_2 \in W^{1,2}(\Omega)$, a global minimizer for $\tilde{E}(\Sigma, g_1, g_2)$ in (2.1) can be found by carrying out the following convex minimization,*

$$(2.4) \quad \min_{0 \leq u \leq 1} \left\{ \int_{\Omega} |\nabla u| + \frac{1}{2} \int_{\Omega} \{ \lambda (f - g_1)^2 + \mu |\nabla g_1|^2 - \lambda (f - g_2)^2 - \mu |\nabla g_2|^2 \} u(x) \right\},$$

and setting $\Sigma = \{x : u(x) \geq \rho\}$ for almost every $\rho \in [0, 1]$.

Proof. See Appendix A. ■

From Theorem 2.2, we see that the term $\text{Per}(\Sigma)$ of (2.1) is replaced by a convex integral term $\int_{\Omega} |\nabla u|$. In other words, the boundary information of Γ in (1.1) can be extracted from the TV (total variation) term $\int_{\Omega} |\nabla u|$. This motivates us to use $\int_{\Omega} |\nabla g|$ to extract the boundary information $\text{Length}(\Gamma)$ in (1.1). Evidently, this approximation is also related to the fuzzy membership approach [7, 14, 33] to handle the Chan–Vese model. In the following, we therefore use $\int_{\Omega} |\nabla g|$ to approximate the boundary term (the last term) in the Mumford–Shah energy (1.1).

Next we consider simplifying the middle term in model (1.1). In (1.1), the solution is restricted to be a smooth function in $\Omega \setminus \Sigma$ and in $\Sigma \setminus \Gamma$. However, from the example given in Figure 2, we see that these smooth parts can be recovered quite well from a smooth function g in Ω by a proper thresholding. Therefore in the following, we look for solution $g \in W^{1,2}(\Omega)$. Then we have the next result.

Lemma 2.3. *If $g \in W^{1,2}(\Omega)$ and Γ is a closed curve with $m(\Gamma) = 0$, where $m(\cdot)$ is the Lebesgue measure, then $\int_{\Gamma} |\nabla g|^2 dx = 0$.*

Proof. Since $g \in W^{1,2}(\Omega)$, we have $\nabla g \in L^2(\Omega)$. Because of $m(\Gamma) = 0$, we get $\int_{\Gamma} |\nabla g|^2 dx = 0$ immediately. ■

Thus the middle term of model (1.1) becomes

$$(2.5) \quad \int_{\Omega \setminus \Gamma} |\nabla g|^2 dx = \int_{\Omega} |\nabla g|^2 dx - \int_{\Gamma} |\nabla g|^2 dx = \int_{\Omega} |\nabla g|^2 dx \quad \forall g \in W^{1,2}(\Omega).$$

In view of Theorem 2.2 and (2.5), we propose our segmentation model as

$$\inf_{g \in W^{1,2}(\Omega)} \left\{ \frac{\lambda}{2} \int_{\Omega} (f - g)^2 dx + \frac{\mu}{2} \int_{\Omega} |\nabla g|^2 dx + \int_{\Omega} |\nabla g| dx \right\},$$

where λ and μ are positive parameters. Since sometimes the given image is degraded by noise and blur, we extend this model to general cases by introducing a problem-related operator \mathcal{A} in its fidelity term. Then, finally, our model is

$$(2.6) \quad \inf_{g \in W^{1,2}(\Omega)} E(g) := \inf_{g \in W^{1,2}(\Omega)} \left\{ \frac{\lambda}{2} \int_{\Omega} (f - \mathcal{A}g)^2 dx + \frac{\mu}{2} \int_{\Omega} |\nabla g|^2 dx + \int_{\Omega} |\nabla g| dx \right\},$$

where \mathcal{A} may stand for the identity operator or a blurring operator. Obviously, if $\mu \neq 0$ in (2.6), g will be smooth. The following theorem shows the existence and uniqueness of g .

Theorem 2.4. *Let Ω be a bounded connected open subset of \mathbb{R}^2 with a Lipschitz boundary. Let $f \in L^2(\Omega)$ and $\text{Ker}(\mathcal{A}) \cap \text{Ker}(\nabla) = \{0\}$, where \mathcal{A} is a bounded linear operator from $L^2(\Omega)$ to itself and $\text{Ker}(\mathcal{A})$ is the kernel of \mathcal{A} . Then (2.6) has a unique minimizer $g \in W^{1,2}(\Omega)$.*

Proof. See Appendix B. ■

We remark that the condition $\text{Ker}(\mathcal{A}) \cap \text{Ker}(\nabla) = \{0\}$ actually restricts $\mathcal{A}1 \neq 0$. It means that $\mathcal{A}f \neq 0$ if f is a nonzero constant image. The condition holds for all blurring operators, as they are convolution operators with positive kernels.

We emphasize that model (2.6) can be minimized quickly by using currently available efficient algorithms such as the split-Bregman algorithm [26] or the Chambolle–Pock method [13, 44]. Once g is obtained, we enter into the second stage of our method, where we use thresholding to segment g into different phases. The thresholds can be determined by any clustering methods or be chosen by the users. We leave the implementation to section 3.

3. Numerical aspects. In this section, we first introduce the split-Bregman algorithm for solving our model (2.6). After that we give a strategy based on the K-means method to determine the thresholds automatically.

3.1. Solution of model (2.6) in the first stage. The discrete setting of our model (2.6) is

$$(3.1) \quad \min_g \left\{ \frac{\lambda}{2} \|f - \mathcal{A}g\|_2^2 + \frac{\mu}{2} \|\nabla g\|_2^2 + \|\nabla g\|_1 \right\},$$

where $\|\nabla g\|_1 := \sum_{i \in \Omega} \sqrt{(\nabla_x g)_i^2 + (\nabla_y g)_i^2}$ is the classical discrete TV seminorm. Here we adopt the backward difference with periodic boundary condition to approximate the discrete gradient operator ∇ ; i.e., for the first row of g , we define

$$(\nabla_x g)_i = \begin{cases} g(1, 1) - g(1, n), & i = 1, \\ g(1, i) - g(1, i - 1), & i = 2, \dots, n, \end{cases}$$

where n is the number of pixels of the first row of g and $g(1, i)$ represents the i th pixel of the first row of g . Similarly, we can define ∇_y . As (3.1) is convex, it can be solved by many methods such as the alternating direction method of multipliers, which is convergent and is well suited to distributed convex optimization; see [5, 23] and references therein. Specifically, its variant, the split-Bregman algorithm [26], is used widely to solve a very broad class of L_1 regularization problems. We can also use the Chambolle–Pock method [13, 44, 29] which provides a convergence rate. In the following, we derive the split-Bregman algorithm for solving (3.1). Clearly the algorithm converges, since our model (3.1) is a convex regularization problem; see [5, 23, 26] for more details of the convergence analysis.

Set $d_x = \nabla_x g$ and $d_y = \nabla_y g$ in (3.1), and this yields the constrained problem

$$\min_g \left\{ \frac{\lambda}{2} \|f - \mathcal{A}g\|_2^2 + \frac{\mu}{2} \|\nabla g\|_2^2 + \|(d_x, d_y)\|_1 \right\} \quad \text{s.t. } d_x = \nabla_x g \text{ and } d_y = \nabla_y g.$$

Using the 2-norm to weakly enforce the above constraints, it becomes

$$\min_{g, d_x, d_y} \left\{ \frac{\lambda}{2} \|f - \mathcal{A}g\|_2^2 + \frac{\mu}{2} \|\nabla g\|_2^2 + \|(d_x, d_y)\|_1 + \frac{\sigma}{2} \|d_x - \nabla_x g\|_2^2 + \frac{\sigma}{2} \|d_y - \nabla_y g\|_2^2 \right\}.$$

Applying the split-Bregman iteration to strictly enforce the constraints, we have at step $(k+1)$

$$(3.2) \quad (g^{k+1}, d_x^{k+1}, d_y^{k+1}) = \arg \min_{g, d_x, d_y} \left\{ \frac{\lambda}{2} \|f - \mathcal{A}g\|_2^2 + \frac{\mu}{2} \|\nabla g\|_2^2 + \|(d_x, d_y)\|_1 + \frac{\sigma}{2} \|d_x - \nabla_x g - b_x^k\|_2^2 + \frac{\sigma}{2} \|d_y - \nabla_y g - b_y^k\|_2^2 \right\},$$

$$(3.3) \quad b_x^{k+1} = b_x^k + (\nabla_x g^{k+1} - d_x^{k+1}), \quad b_y^{k+1} = b_y^k + (\nabla_y g^{k+1} - d_y^{k+1}).$$

The minimization (3.2) can be solved effectively by minimizing with respect to g and (d_x, d_y) alternatively. Hence we need to solve the following two minimization subproblems:

$$(3.4) \quad g^{k+1} = \arg \min_g \left\{ \frac{\lambda}{2} \|f - \mathcal{A}g\|_2^2 + \frac{\mu}{2} \|\nabla g\|_2^2 + \frac{\sigma}{2} \|d_x^k - \nabla_x g - b_x^k\|_2^2 + \frac{\sigma}{2} \|d_y^k - \nabla_y g - b_y^k\|_2^2 \right\},$$

$$(3.5) \quad (d_x^{k+1}, d_y^{k+1}) = \arg \min_{d_x, d_y} \left\{ \|(d_x, d_y)\|_1 + \frac{\sigma}{2} \|d_x - \nabla_x g^{k+1} - b_x^k\|_2^2 + \frac{\sigma}{2} \|d_y - \nabla_y g^{k+1} - b_y^k\|_2^2 \right\}.$$

Since the right-hand side of (3.4) is differentiable, g^{k+1} satisfies the following optimality condition:

$$(3.6) \quad (\lambda \mathcal{A}^* \mathcal{A} - (\mu + \sigma) \Delta) g = \lambda \mathcal{A}^* f + \sigma \nabla_x^T (d_x^k - b_x^k) + \sigma \nabla_y^T (d_y^k - b_y^k),$$

where \mathcal{A}^* is the conjugate transpose of \mathcal{A} and $\Delta = -(\nabla_x^T \nabla_x + \nabla_y^T \nabla_y)$. Since $\text{Ker}(\mathcal{A}) \cap \text{Ker}(\Delta) = \{0\}$, the matrix $[\lambda \mathcal{A}^* \mathcal{A} - (\mu + \sigma) \Delta]$ is positive definite and hence is invertible. Using the Gauss–Seidel method in [26] or the fast Fourier transforms to diagonalize the circulant matrices \mathcal{A}

and Δ (see [17]), (3.6) can be solved efficiently. For problem (3.5), it can be solved explicitly using a generalized shrinkage formula [26] as follows:

$$(3.7) \quad d_x^{k+1} = \max\left(s^k - \frac{1}{\sigma}, 0\right) \frac{s_x^k}{s^k}, \quad d_y^{k+1} = \max\left(s^k - \frac{1}{\sigma}, 0\right) \frac{s_y^k}{s^k},$$

where $s_x^k = \nabla_x g^{k+1} + b_x^k$, $s_y^k = \nabla_y g^{k+1} + b_y^k$ and $s^k = \sqrt{(s_x^k)^2 + (s_y^k)^2}$. The following algorithm summarizes the procedure of solving our minimization problem (3.1).

ALGORITHM 1. Solving (3.1) by the split-Bregman algorithm.

-
1. Initialize: $g^0 = f, d_x^0 = d_y^0 = b_x^0 = b_y^0 = 0$.
 2. Do $k = 0, 1, \dots$, until $\frac{\|g^k - g^{k+1}\|_F}{\|g^{k+1}\|_F} < \epsilon$.
 - (a) Compute g^{k+1} by solving (3.6).
 - (b) Compute d_x^{k+1} and d_y^{k+1} by the shrinkage formula (3.7).
 - (c) Update b_x^{k+1} and b_y^{k+1} by the formula (3.3).
 3. Output: g .
-

3.2. Determining the thresholds in the second stage. As mentioned before, our segmentation result is obtained by thresholding the solution g of (3.1) with proper threshold(s) ρ . For example, for two-phase segmentation, one may choose ρ to be the mean value of g , and then use this ρ to threshold g into two phases. Or the user can try different values of ρ to get the best result. Note that *there is no need to recompute the image g when we change ρ* . We just threshold the image g with the new ρ to get a new binary image.

In case one wants to choose the thresholds automatically, here we discuss how to choose them using clustering methods. There are many clustering methods, including the K-means methods [28, 30, 38] and the convex K-means method named SON clustering [36]. To standardize the discussions, we begin by normalizing the pixel values of g to $[0,1]$. We do this by using the linear-stretch formula:

$$(3.8) \quad \bar{g} = \frac{g - g_{\min}}{g_{\max} - g_{\min}},$$

where g_{\max} and g_{\min} represent maximum and minimum of g , respectively.

In the following, we use the MATLAB K-means command KMEANS as an example to introduce our strategy of choosing the thresholds automatically. The K-means method is a very efficient method for classifying a given set into K clusters, with K specified in advance. Suppose we want to segment \bar{g} into K segments, $K \geq 2$. We use the K-means method to classify the set of pixel values of \bar{g} into K clusters. Let the mean value of each cluster be $\hat{\rho}_1, \hat{\rho}_2, \dots, \hat{\rho}_K$, and without loss of generality, let $\hat{\rho}_1 \leq \hat{\rho}_2 \leq \dots \leq \hat{\rho}_K$. Then we define the $(K-1)$ thresholds as

$$(3.9) \quad \rho_i = \frac{\hat{\rho}_i + \hat{\rho}_{i+1}}{2}, \quad i = 1, 2, \dots, K-1.$$

The i th phase of \bar{g} , $1 \leq i \leq K$, is then given by $\{x \in \Omega : \rho_{i-1} < \bar{g}(x) \leq \rho_i\}$. To obtain the boundary of the i th phase, we set pixels in the i th phase to 1 and all the other pixels to zero;

then we invoke the command `CONTOUR` in `MATLAB`. Again we emphasize that if we change the value of K or the thresholds, there is *no need to recompute g and \bar{g}* .

4. Experimental results. In this section, we compare our segmentation model (3.1) with three two-phase segmentation methods proposed in [16, 19, 53] and five multiphase segmentation methods proposed in [33, 47, 54, 3, 45]. Methods [16] and [19] use TV and framelet regularization terms, respectively; therefore, we can compare the performance of these two different regularization approaches with ours. Methods [19, 53, 33, 47, 54, 3, 45] are effective segmentation methods all published in or after 2009. The codes we used are provided by the authors. Apart from some default settings, like the maximum number of iterations, the parameters in the codes are chosen by trial and error to give the best results of the respective methods.

For two-phase segmentation, we use ρ^M , ρ_1 , and ρ^U to denote the thresholds we used in the tests. They represent respectively the mean of the normalized image \bar{g} given in (3.8), the threshold obtained by K-means given in (3.9), and a threshold chosen by us, the user. For multiphase segmentation, we tried the thresholds ρ_i , obtained by K-means in (3.9), and ρ_i^U , chosen by us. The tolerance ϵ and the step size σ used in the split-Bregman algorithm in (3.2) were fixed to be 10^{-4} and 2, respectively. The parameters λ , μ are chosen empirically. All the results were tested on a MacBook with 2.4 GHz processor and 4GB RAM. The boundaries of all the results are shown with color and superimposed on the given images.

4.1. Two-phase segmentation.

Example 1: Antimass image. Figure 3(a) is the given image. Figures 3(b)–(d) are the results of methods [16, 19, 53], respectively. Figure 3(e) is our smooth solution g from Algorithm 1 using parameters $\lambda = 3$ and $\mu = 1$; see (3.1). Figures 3(f)–(i) are the segmentation results on the normalized \bar{g} (see (3.8)) with thresholds $\rho^M = 0.1898$, $\rho_1 = 0.2669$, and $\rho^U = 0.1, 0.2$, respectively. Note that ρ^M and ρ_1 are computed automatically. From the results, we see that our method can reveal different meaningful features in the image by choosing different ρ 's; this can be done without recomputing g . In contrast, for the methods of [16, 19, 53], one will need to solve the minimization models again if one wants to reveal different features in the image.

Example 2: Tubular image. Figure 4(a) is a given magnetic resonance angiography kidney image [24]. The boundaries of the vessels are blurry and vague, so that they are hard to detect. Figure 4(e) is the solution g from Algorithm 1 using $\lambda = 20$ and $\mu = 1$. Figures 4(f)–(h) are our segmentation results with thresholds $\rho^M = 0.1760$, $\rho_1 = 0.4019$, and $\rho^U = 0.2$, respectively. By comparing our results with the results from methods [16, 19, 53] in Figures 4(b)–(d), we see that our method can better detect and connect the blood vessels. Recently, we proposed a tight-frame method specifically for segmenting vessels [9]. Here we give the result of that method in Figure 4(i), and we see that it is comparable to our method.

Example 3: Image with high noise. In Figures 5(a) and (b), we give the clean and the noisy images, respectively. The noise we added is high: Gaussian noise with mean 0.6 and variance 0.25. Figures 5(c)–(e) give the results of methods [16, 19, 53], respectively, on the noisy image. We see that method [19], which uses tight-frame regularization, recovers these objects better than method [16], which uses TV regularization, and that method [53] fails completely. Figure 5(f) is our solution g when $\lambda = 4$ and $\mu = 1$. Figures 5(g)–(i) are the segmentation results

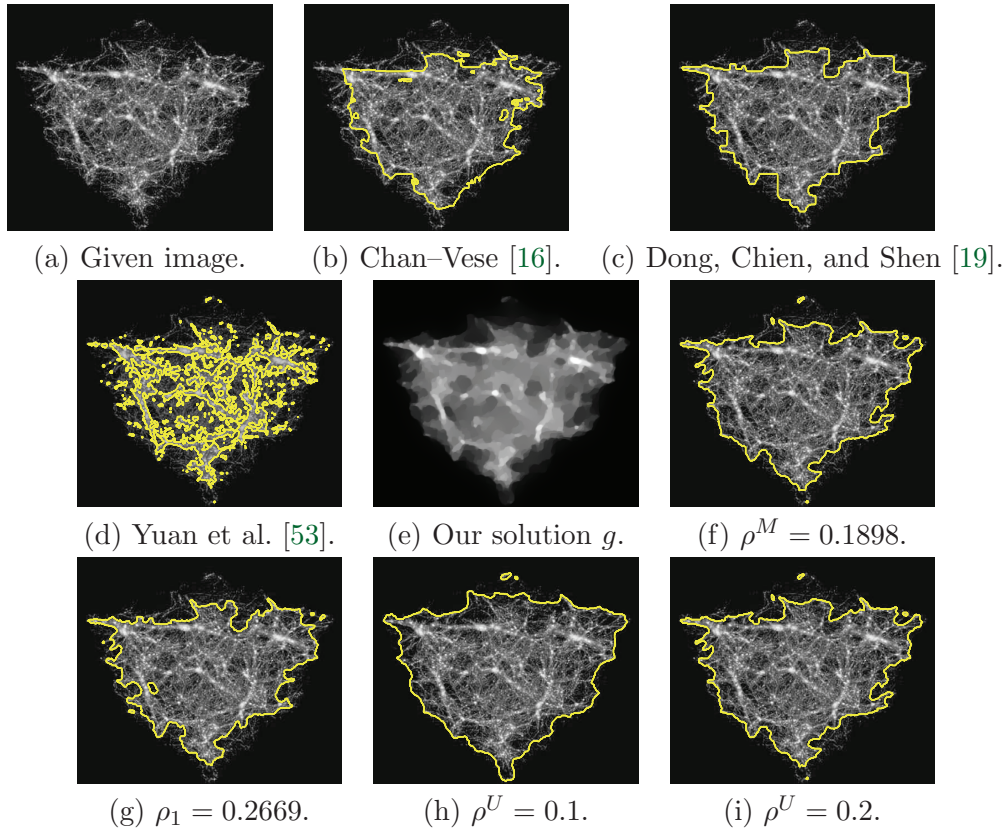


Figure 3. Antimass image segmentation. (a) Given 384×480 image; (b)–(d) results of methods [16], [19], and [53], respectively; (e) our smooth solution g ; (f)–(i) our segmentation results using thresholds $\rho^M = 0.1898$, $\rho_1 = 0.2669$, $\rho^U = 0.1$, and 0.2 , respectively.

Table 1

Iteration numbers and CPU time in seconds for two-phase segmentation.

Example	Chan–Vese [16]		Dong [19]		Yuan [53]		Our method	
	Iter.	Time	Iter.	Time	Iter.	Time	Iter.	Time
Figure 3	1000	263.73	300	83.82	64	6.01	172	18.38
Figure 4	1000	76.62	300	32.17	18	0.37	115	3.03
Figure 5	1000	23.42	300	10.18	108	0.42	63	0.49
Figure 6	1300	28.19	300	10.18	20	0.09	52	1.13
Figure 7	1500	31.78	300	10.18	24	0.10	65	1.21

with thresholds $\rho^M = 0.8308$, $\rho_1 = 0.6371$, and $\rho^U = 0.7$ respectively. Clearly, our results are all good and comparable to the method in [19], i.e., Figure 5(d). However, our method is much faster (see Table 1). Notice that the differences between our results (g)–(i) are small, indicating that our method is robust with respect to the threshold.

Example 4: Blurry and noisy image. To illustrate the robustness of our method with respect to the threshold, we tested our method on two blurry images: Figure 6 with motion blur and Figure 7 with Gaussian blur. For the motion blur, the motion is vertical and the

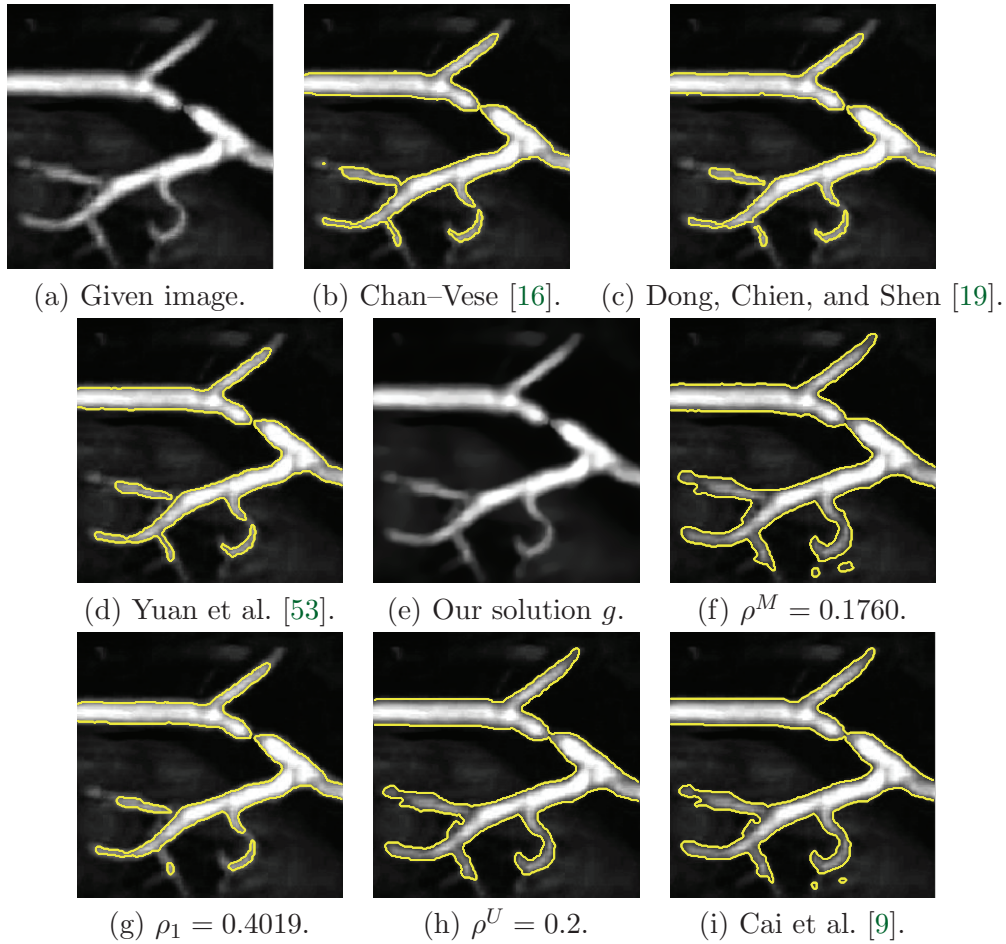


Figure 4. *Kidney vascular system segmentation.* (a) Given 256×256 image; (b)–(d) results of methods [16], [19], and [53], respectively; (e) our solution g ; (f)–(h) our segmentation of results using thresholds $\rho^M = 0.1760$, $\rho_1 = 0.4019$, and $\rho^U = 0.2$, respectively; (i) result of method [9].

filter size is 15. For the Gaussian blur, the filter used is of size $[15, 15]$ with standard deviation 15. For both images, we added a Gaussian noise with mean 10^{-3} and variance 2×10^{-3} . Figures 6(f) and 7(f) are our solutions g obtained by using $\lambda = 100$ and $\mu = 1$. From Figures 6(c)–(e) and 7(c)–(e), we see that all of the results of methods [16, 19, 53] are not good. More precisely, methods [16, 19] give incorrect boundaries (linking the ring and the horseshoe object together), while method [53] misses a large portion of the objects. In contrast, our boundary recovers the shapes of the objects very well; see Figures 6(g)–(i) and 7(g)–(i).

Table 1 gives the CPU time comparison of the methods. We see that our method is second only to the two-phase continuous max-flow method in [53]. But from Examples 1–4, we see that our method gives much better segmentation results than the method in [53]. We remark that for the examples we tested, the framelet method in [19] did not converge within the maximum number of iterations (300) using the given tolerance 10^{-3} specified in the code.

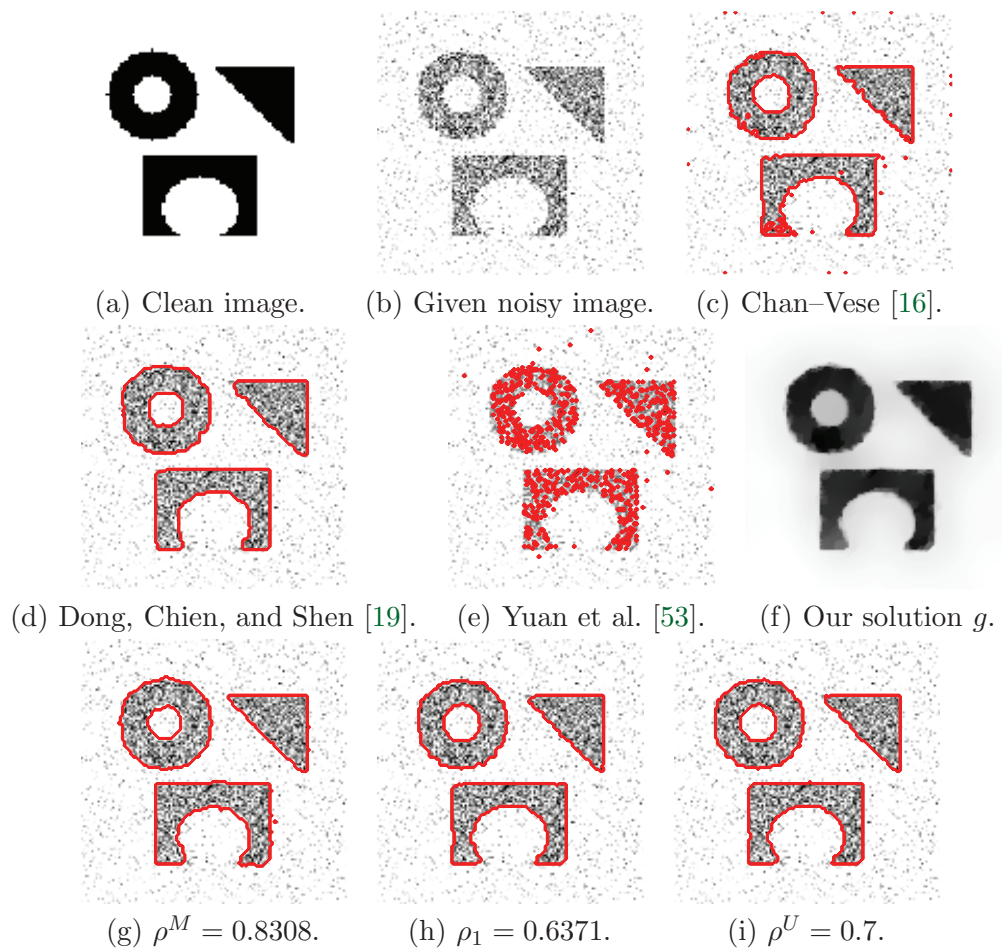


Figure 5. Noisy image segmentation. (a) Clean 128×128 image; (b) given noisy image; (c)–(e) results of methods [16], [19], and [53], respectively; (f) our solution g ; (g)–(i) our segmentation results using thresholds $\rho^M = 0.8308$, $\rho_1 = 0.6371$, and $\rho^U = 0.7$, respectively.

4.2. Multiphase segmentation.

Example 5: Three-phase image. Figure 8(a) is the given image, and Figures 8(b)–(f) are the three-phase segmentation results by methods of [33, 47, 54, 3, 45]. Figure 8(g) is our solution g obtained with $\lambda = 30$ and $\mu = 0.1$. Figures 8(i)–(k) are the boundaries of the three phases obtained from \bar{g} (defined in (3.8)) using thresholds $\rho_1 = 0.1929$ and $\rho_2 = 0.6009$, which are computed automatically by the K-means method (3.9). Figure 8(h) is a trinary representation of the three phases by using the mean value of each phase to represent that phase. We see that all results are good except the results of method [54] (Figure 8(d)), which separates the cloud in the lower right corner into two parts, and of method [3] (Figure 8(e)), which misses a large part of the cloud. We emphasize that for our method, we do not need to determine the number of phases K at the beginning. We can modify K after obtaining g , and compute the thresholds $\{\rho_i\}_{i=1}^{K-1}$ by any clustering methods to segment g into K segments. This is not the case for methods [3, 33, 54, 45], where one has to specify K before minimizing

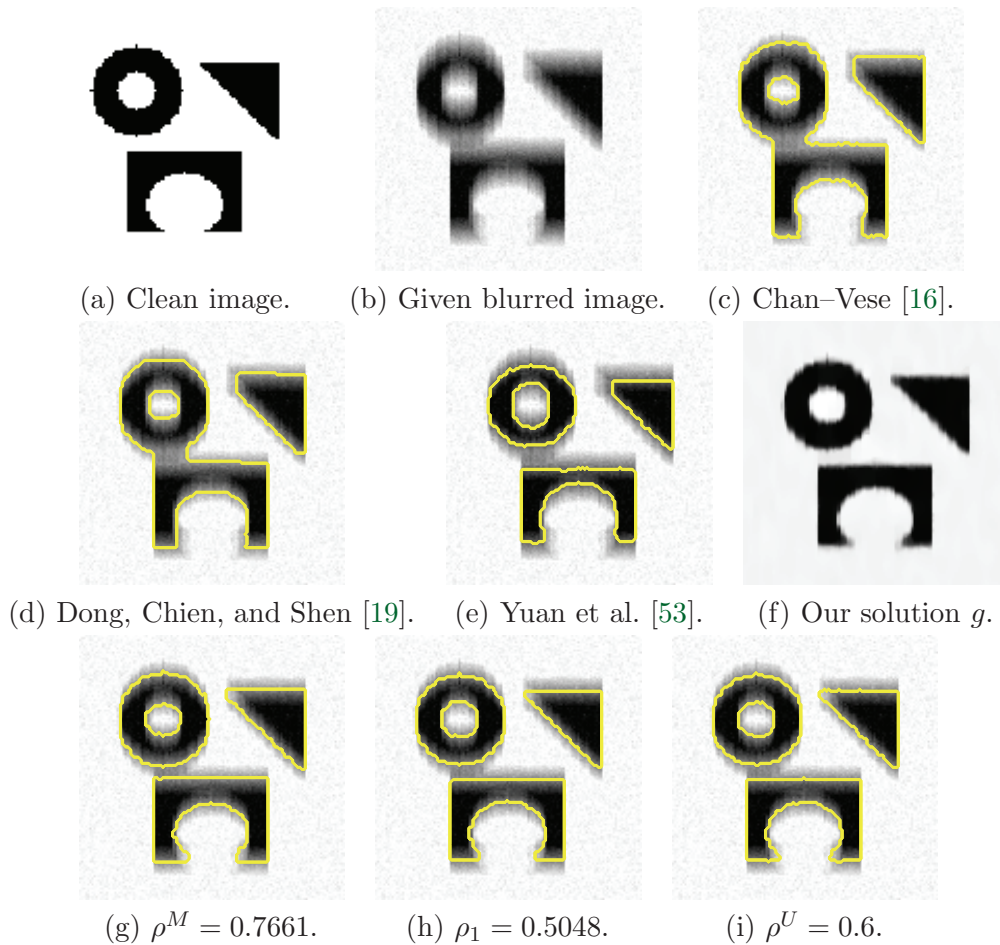


Figure 6. Segmentation of motion blurred image. (a) Clean 128×128 image; (b) given blurred and noisy image; (c)–(e) results of methods [16], [19], and [53], respectively; (f) our solution g ; (g)–(i) our results using thresholds $\rho^M = 0.7661$, $\rho_1 = 0.5048$ and $\rho^U = 0.6$, respectively.

their problems. Moreover, we found in our tests that method [33] is sensitive to initialization, where different initializations may give quite different results.

Example 6: Four-phase noisy image. Figures 9(a) and (b) give the clean and the noisy images (Gaussian noise with zero mean and variance 0.03). Figure 9(h) is our solution g obtained by using $\lambda = 4$ and $\mu = 0.1$. The thresholds computed automatically by K-means method (3.9) are $\rho_1 = 0.1652$, $\rho_2 = 0.4978$, $\rho_3 = 0.8319$. The corresponding four-phase segmentation is given in Figure 9(i), where the four phases of g are shown by using the mean values of each phase to represent the phase. Figures 9(j)–(m) give the boundaries of the phases. We see that the four phases are recovered almost exactly by methods [3, 45] and our method; see Figure 9(f), (g), and (i). In contrast, method [33] (Figure 9(c)) segments one phase incorrectly, method [47] (Figure 9(d)) fails, and method [54] (Figure 9(e)) gives oscillatory boundaries.

Example 7: Four-phase blurry and noisy image. The blurry and noisy image used is given in Figure 10(b). The blur is a motion blur where the motion is vertical and the filter size is

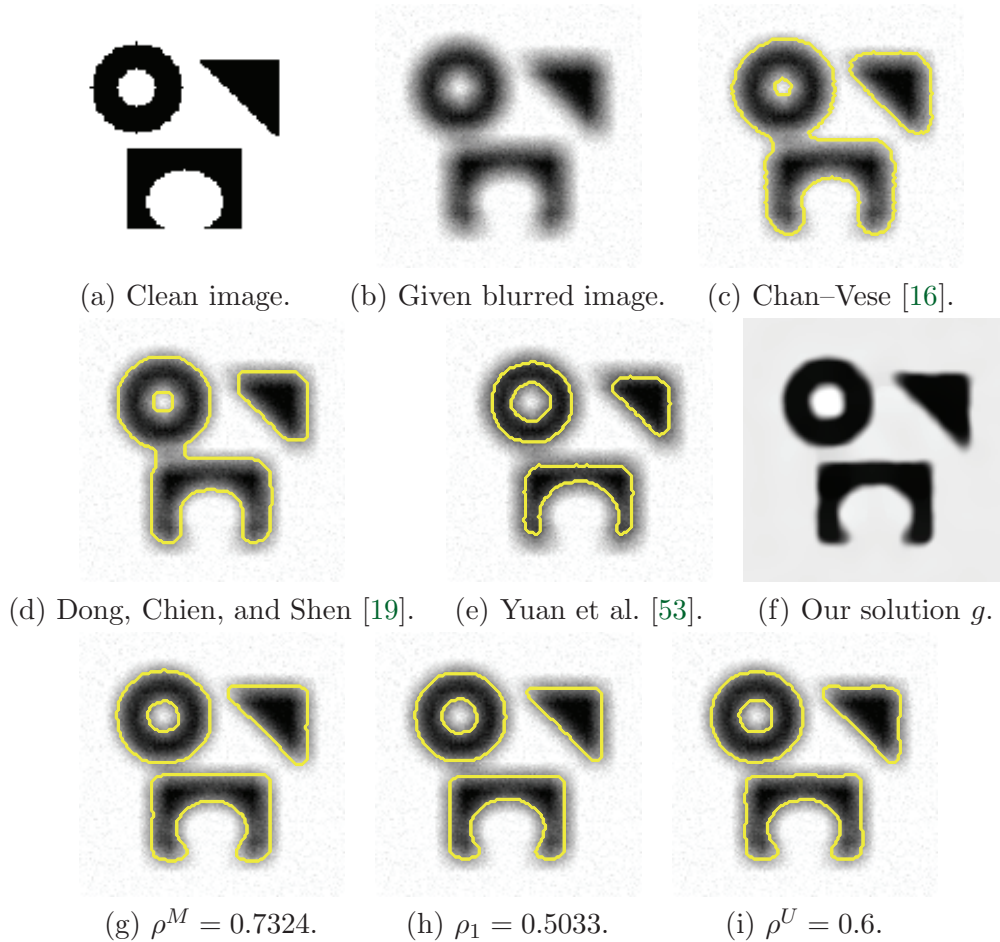


Figure 7. Segmentation of Gaussian blurred image. (a) Clean 128×128 image; (b) given blurred and noisy image; (c)–(e) results of methods [16], [19], and [53], respectively; (f) our solution g ; (g)–(i) our results using thresholds $\rho^M = 0.7324$, $\rho_1 = 0.5033$, and $\rho^U = 0.6$, respectively.

15. The noise is Gaussian noise with mean 10^{-3} and variance 2×10^{-3} . Figure 10(h) is our solution g obtained by using $\lambda = 40$ and $\mu = 1$. The thresholds from the K-means method (3.9) are $\rho_1 = 0.1704$, $\rho_2 = 0.4971$, $\rho_3 = 0.8248$. Figure 10(i) gives the corresponding four phases, and Figures 10(j)–(m) give the boundaries of the phases. We see that the four phases of the image are recovered almost exactly by our method; see Figure 10(i). But from Figures 10(c)–(g), we see that the results from all the other multiphase methods [33, 47, 54, 3, 45] are not good.

Example 8: Four-phase brain MRI image. Finally, we test the four-phase brain MRI image used in [45]; see Figure 11(a). The gray and white matter segmentation for this kind of image is very important in medical imaging. Figure 11(g) is our solution g obtained by using $\lambda = 40$ and $\mu = 1$. The thresholds from the K-means method (3.9) are $\rho_1 = 0.1627$, $\rho_2 = 0.4947$, $\rho_3 = 0.7757$, and Figure 11(h) gives the corresponding four phases. Figure 11(i) is the corresponding four phases using user-given thresholds $\rho_1^U = 0.1$, $\rho_2^U = 0.4$, $\rho_3^U = 0.7$,

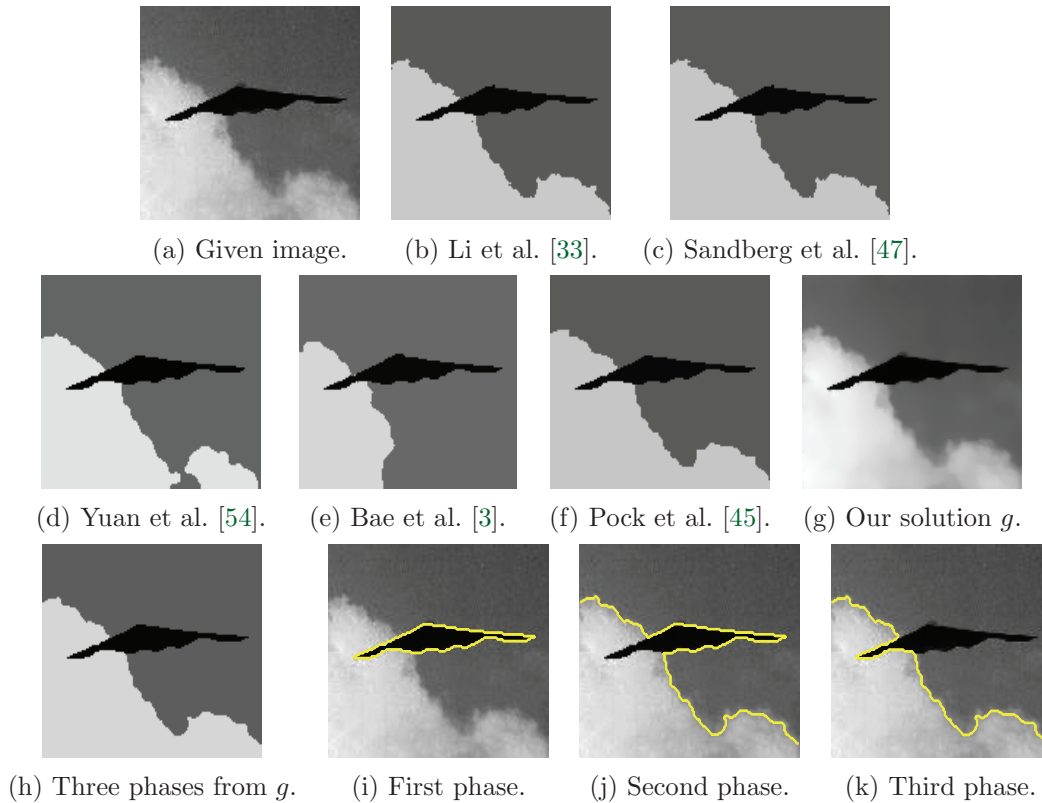


Figure 8. Three-phase segmentation. (a) Given 125×150 image; (b)–(f) results of methods [33], [47], [54], [3], and [45], respectively; (g) our solution g ; (h) three phases using thresholds $\rho_1 = 0.1929$, $\rho_2 = 0.6009$; (i)–(k) boundary of each phase of g .

Table 2

Iteration numbers and CPU time in seconds for multiphase segmentation.

Method	Figure 8		Figure 9		Figure 10		Figure 11	
	Iter.	Time	Iter.	Time	Iter.	Time	Iter.	Time
Li [33]	100	1.56	100	7.64	100	7.26	100	9.39
Sandberg [47]	2	3.15	12	90.59	13	93.79	6	56.21
Yuan [54]	32	0.58	134	14.51	57	5.82	80	12.23
Bae [3]	50	2.04	50	8.96	50	8.72	50	13.12
Pock [45]	50	1.08	70	10.85	70	11.51	50	8.71
Our method	62	0.57	112	3.04	78	2.90	46	2.75

and Figures 11(j)–(m) give the boundaries of the phases. We see that our method and the methods of [33, 47, 3, 45] all give very good results, while method [54] can not separate the third and the fourth phases. We note, however, that the methods of [33, 47, 3, 45] all have to solve the minimization problem again if K is changed, while ours does not. Moreover, from the timing in Table 2, our method is the fastest.

Table 2 gives the CPU time comparison of the methods. We see that our method is the fastest. Note that method [54] is comparable to ours in time, but from Examples 5–8, we see

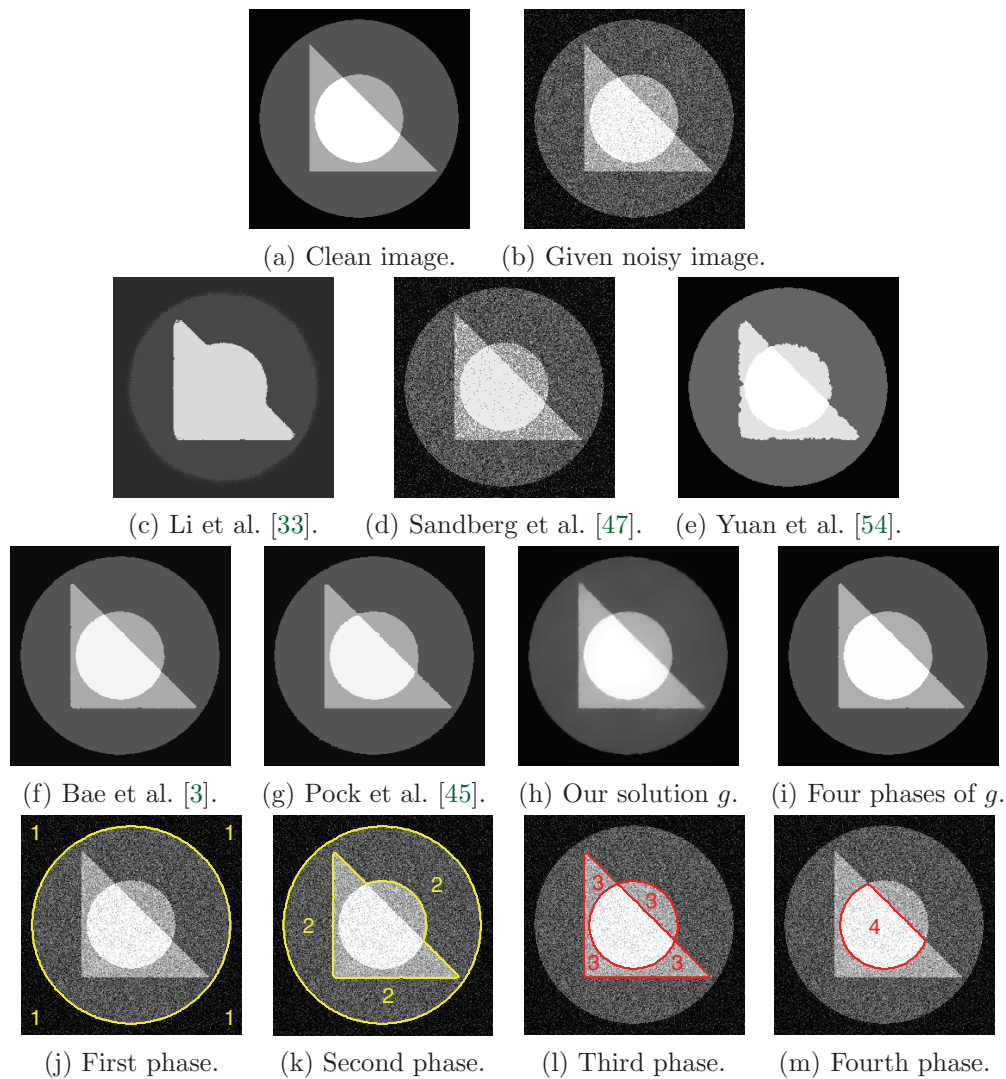


Figure 9. Four-phase segmentation for noisy image. (a) Clean 256×256 image; (b) given noisy image; (c)–(g) results of methods [33], [47], [54], [3], and [45], respectively; (h) our solution g ; (i) four phases using thresholds $\rho_1 = 0.1652$, $\rho_2 = 0.4978$, $\rho_3 = 0.8319$; (j)–(m) boundary of each phase of g .

that our method gives better segmentation. In fact, our model is based on the Mumford–Shah model (1.1), which admits more high-order information. But methods [53, 54] are basically using constants to approximate regions. This may explain why they fail in Figures 5(e), 8(d), and 11(d) and give poor results in other examples.

5. Relationship with image restoration. It is interesting to note that our model (2.6) itself can be regarded as an image restoration model to capture the cartoon part in the image and is closely related to the classical Rudin–Osher–Fatemi (ROF) image restoration model:

$$(5.1) \quad \inf_g \int_{\Omega} \left(\frac{\lambda}{2} (f - Ag)^2 dx + |\nabla g| \right);$$

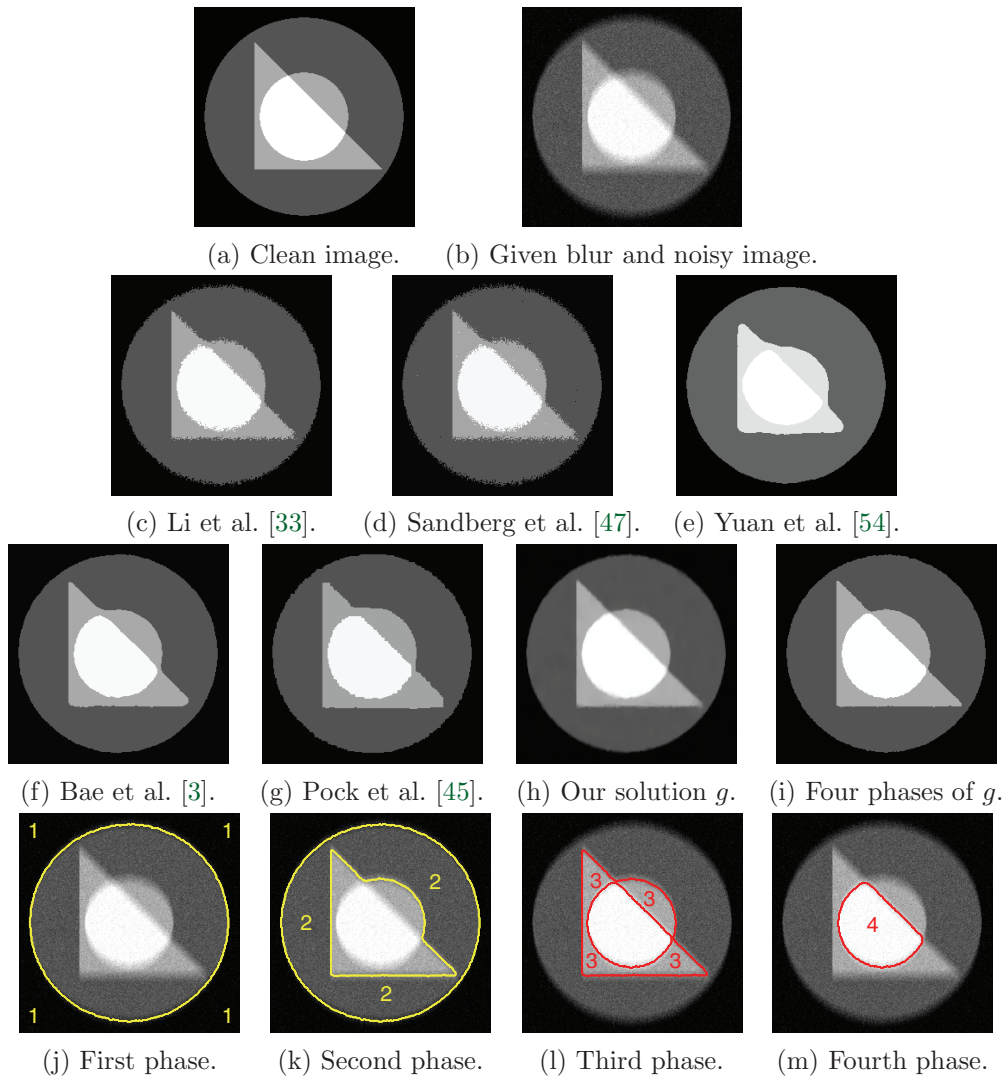


Figure 10. Four-phase segmentation for noisy and blurry image. (a) Clean 256×256 image; (b) given blurred and noisy image; (c)–(g) results of methods [33], [47], [54], [3], and [45], respectively; (h) our solution g ; (i) four phases using thresholds $\rho_1 = 0.1704$, $\rho_2 = 0.4971$, $\rho_3 = 0.8248$; (j)–(m) boundary of each phase of g .

see [46]. The only difference is that we have an extra term $\int_{\Omega} |\nabla g|^2$. One of the important properties of the ROF model is that it can preserve important edge information, but the staircase effect may be introduced. In order to avoid this, many works have been proposed; see [15, 37, 51, 6] for examples. In [15], Chan, Marquina, and Mulet proposed to solve the following minimization problem:

$$(5.2) \quad \inf_g \int_{\Omega} \left(\frac{\lambda}{2} (f - g)^2 + |\nabla g|_{\epsilon_1} + \mu \frac{(\Delta g)^2}{|\nabla g|_{\epsilon_2}^3} \right),$$

where $|\nabla g|_{\epsilon_i} = \sqrt{|\nabla g|^2 + \epsilon_i}$, $i = 1, 2$, with ϵ_i being small positive parameters. The additional higher-order derivative term can remove the staircase effect. In [37, 51], the authors used

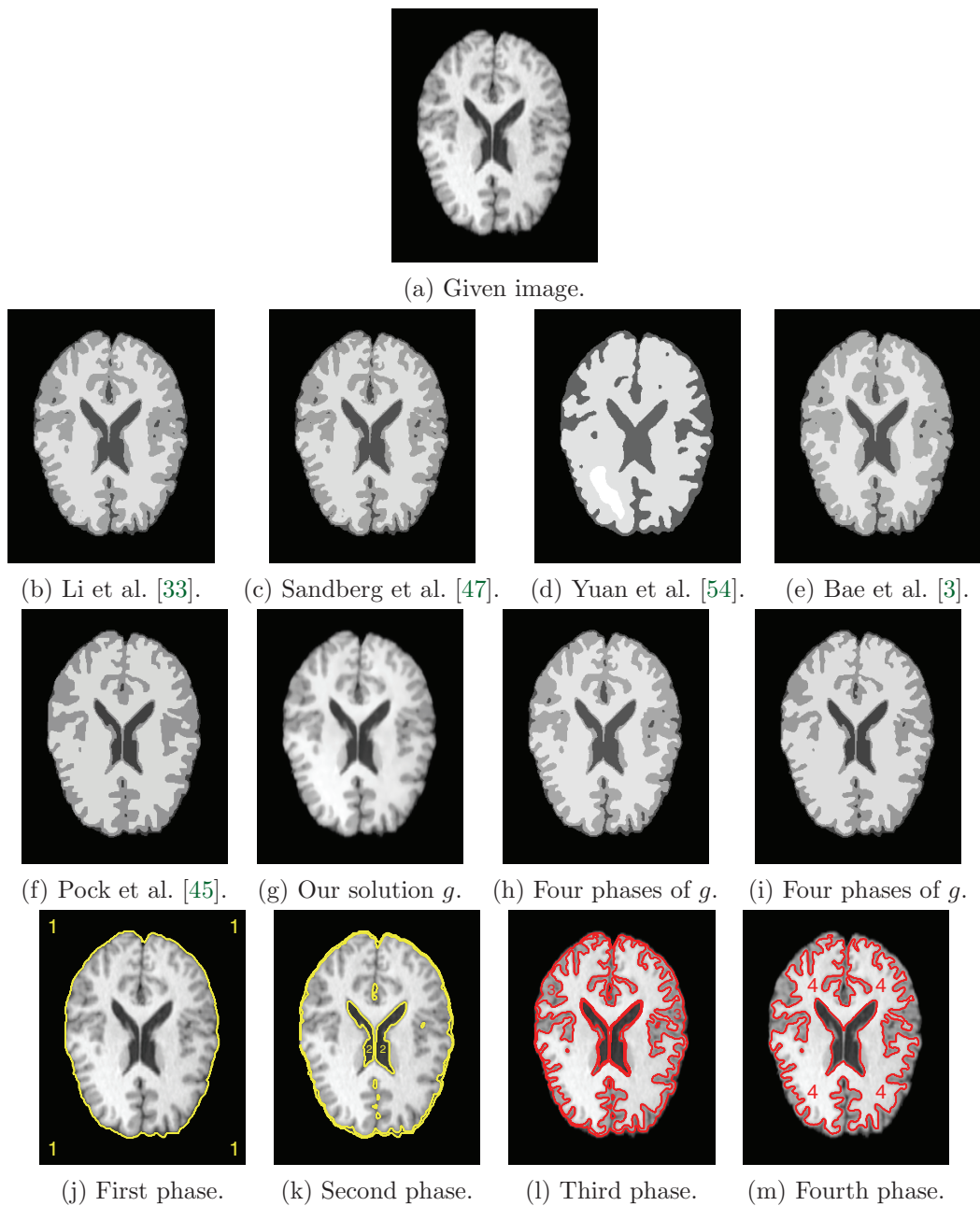


Figure 11. Four-phase gray and white matter segmentation for a brain MRI image. (a) Given 319×256 brain MRI image; (b)–(f) results of methods [33], [47], [54], [3], and [45], respectively; (g) our solution g ; (h)–(i) four phases using thresholds $\rho_1 = 0.1627$, $\rho_2 = 0.4947$, $\rho_3 = 0.7757$ and $\rho_1^U = 0.1$, $\rho_2^U = 0.4$, $\rho_3^U = 0.7$, respectively; (j)–(m) boundary of each phase in panel (i).

second-order derivatives to replace the TV regularization term of model (5.1). Recently a novel regularization model, the total generalized variation, was proposed in [6], which also involves higher-order derivatives. Obviously, the cost and difficulty of solving the given models

grow as the functionals became more and more complex.

In contrast, in our model (2.6), the staircase effect is reduced because of the middle term which contains the square of the first-order derivative and no other higher-order derivatives. Once the smooth solution g is found, a suitable thresholding then gives the image segmentation result. From the analysis of section 2 and the numerical results in section 4, we see that the smoothness in g does not affect the segmentation significantly. Nonetheless, it will be interesting to replace our model (2.6) used in the first stage of our method by improved TV models such as (5.2) in [15]. We leave this project to the future.

6. Conclusions. In this paper, we have proposed a two-stage method for segmentation that makes use of a *convex* model (2.6) based on the Mumford–Shah model. In the first stage, our method finds the unique smooth minimizer by the split-Bregman algorithm [26]. Then in the second stage, it uses a thresholding strategy to segment the image. Since our model (2.6) can be regarded as an image restoration model, our method unifies the image processing works of image segmentation and image restoration. Furthermore, our method combines the two-phase and multiphase segmentation into one single algorithm. In fact, one does not have to specify the number of phases before finding the solution to the model. One can segment the solution into different phases by choosing proper thresholds after the solution is obtained in the first stage. We have introduced a K-means method to choose the thresholds automatically. The experimental results show that our method is very effective and robust for many kinds of images, such as antimass, tubular, MRI, noisy, or blurry images.

Appendix A. Proof of Theorem 2.2. This proof basically follows the proof of Theorem 2 in [14]. Using the co-area formula and noting that $0 \leq u \leq 1$, we have $\int_{\Omega} |\nabla u| = \int_0^1 \text{Per}(\{x : u(x) > \rho\}) d\rho$. For the second term in (2.4), we proceed as follows:

$$\begin{aligned}
& \int_{\Omega} \{\lambda(f - g_1)^2 + \mu|\nabla g_1|^2 - \lambda(f - g_2)^2 - \mu|\nabla g_2|^2\} u(x) \\
&= \int_{\Omega} \{\lambda(f - g_1)^2 + \mu|\nabla g_1|^2 - \lambda(f - g_2)^2 - \mu|\nabla g_2|^2\} \int_0^1 \mathbf{1}_{[0, u(x)]}(\rho) d\rho dx \\
&= \int_0^1 \int_{\Omega} \{\lambda(f - g_1)^2 + \mu|\nabla g_1|^2 - \lambda(f - g_2)^2 - \mu|\nabla g_2|^2\} \mathbf{1}_{[0, u(x)]}(\rho) dx d\rho \\
&= \int_0^1 \int_{\Omega \cap \{x: u(x) > \rho\}} \{\lambda(f - g_1)^2 + \mu|\nabla g_1|^2 - \lambda(f - g_2)^2 - \mu|\nabla g_2|^2\} dx d\rho \\
&= \int_0^1 \int_{\Omega \cap \{x: u(x) > \rho\}} \{\lambda(f - g_1)^2 + \mu|\nabla g_1|^2\} dx d\rho - C \\
&\quad + \int_0^1 \int_{\Omega \cap \{x: u(x) > \rho\}^c} \{\lambda(f - g_2)^2 + \mu|\nabla g_2|^2\} dx d\rho,
\end{aligned}$$

where $C = \int_{\Omega} \{\lambda(f - g_2)^2 + \mu|\nabla g_2|^2\} dx$ is independent of u . Setting $\Sigma(\rho) = \overline{\{x : u(x) > \rho\}}$ and $\Gamma(\rho) = \partial\Sigma(\rho)$, we have

$$\begin{aligned}
 (A.1) \quad & \int_{\Omega} |\nabla u| + \frac{1}{2} \int_{\Omega} \{ \lambda(f - g_1)^2 + \mu |\nabla g_1|^2 - \lambda(f - g_2)^2 - \mu |\nabla g_2|^2 \} u(x) \\
 &= \int_0^1 \text{Per}(\Sigma(\rho)) d\rho + \frac{1}{2} \int_0^1 \int_{\Sigma(\rho) \setminus \Gamma(\rho)} \{ \lambda(f - g_1)^2 + \mu |\nabla g_1|^2 \} dx d\rho \\
 &\quad + \frac{1}{2} \int_0^1 \int_{\Omega \setminus \Sigma(\rho)} \{ \lambda(f - g_2)^2 + \mu |\nabla g_2|^2 \} dx d\rho - \frac{C}{2} \\
 &= \int_0^1 \tilde{E}(\Sigma(\rho), g_1, g_2) d\rho - \frac{C}{2},
 \end{aligned}$$

where $\tilde{E}(\Sigma(\rho), g_1, g_2)$ is given in (2.1). Hence, if $u(x)$ is a minimizer of the convex problem in (A.1), then the set $\Sigma(\rho)$ has to be the minimizer of the energy $\tilde{E}(\cdot, g_1, g_2)$ for almost every $\rho \in [0, 1]$. ■

Appendix B. Proof of Theorem 2.4. Recall that $E(g)$ is defined in (2.6). First we prove that $0 \leq \inf_g E(g) < \infty$. Indeed, the left-hand side is obvious. Moreover, if we choose $g_0 = 0$, we get

$$\inf_g E(g) \leq E(g_0) = \frac{\lambda}{2} \int_{\Omega} f^2 dx < \infty.$$

Thus the minimal value of $E(g)$ must exist.

Existence: Note that $W^{1,2}(\Omega)$ is a reflective Banach space, and $E(g)$ is convex and lower semicontinuous. Using Proposition 1.2 in [20], we just need to prove that $E(g)$ is coercive over $W^{1,2}(\Omega)$. For any $g \in W^{1,2}(\Omega)$, obviously $\|\nabla g\|_{L^2(\Omega)} = (\int_{\Omega} |\nabla g|^2 dx)^{\frac{1}{2}}$ is bounded by $\sqrt{\frac{2}{\mu} E(g)}$. In order to prove that $E(g)$ is coercive over $W^{1,2}(\Omega)$, we just have to prove that $\|g\|_{L^2(\Omega)}$ can also be bounded by $\sqrt{E(g)}$. Using the Poincaré inequality on $W^{1,2}(\Omega)$ (see [22]), we have

$$(B.1) \quad \|g - g_{\Omega}\|_{L^2(\Omega)} \leq C_{\Omega} \|\nabla g\|_{L^2(\Omega)} \leq C_{\Omega} \sqrt{\frac{2}{\mu} E(g)},$$

where C_{Ω} is a positive constant and $g_{\Omega} = \frac{1}{|\Omega|} \int_{\Omega} g(x) dx$. Moreover,

$$\begin{aligned}
 (B.2) \quad & g_{\Omega} \cdot \|\mathcal{A}1\|_{L^2(\Omega)} \leq \|f - \mathcal{A}g\|_{L^2(\Omega)} + \|f - \mathcal{A}(g - g_{\Omega})\|_{L^2(\Omega)} \\
 & \leq \sqrt{\frac{2}{\lambda} E(g)} + \|f\|_{L^2(\Omega)} + \|\mathcal{A}\| \cdot \|g - g_{\Omega}\|_{L^2(\Omega)} \\
 & \leq \|f\|_{L^2(\Omega)} + \left(\sqrt{\frac{2}{\lambda}} + C_{\Omega} \|\mathcal{A}\| \sqrt{\frac{2}{\mu}} \right) \sqrt{E(g)}.
 \end{aligned}$$

By the assumption $\text{Ker}(\mathcal{A}) \cap \text{Ker}(\nabla) = \{0\}$, we know that $\|\mathcal{A}1\|_{L^2(\Omega)}$ is nonzero. Thus g_{Ω} is bounded by a constant plus $\sqrt{E(g)}$ times a constant. Since

$$\|g\|_{L^2(\Omega)} \leq \|g_{\Omega}\|_2 + \|g - g_{\Omega}\|_2,$$

using (B.1) and (B.2), $\|g\|_{L^2(\Omega)}$ can also be bounded by a constant plus $\sqrt{E(g)}$ times a constant. Hence $\|g\|_{W^{1,2}(\Omega)}$ is bounded by a constant plus $\sqrt{E(g)}$ times a constant. This means that $E(g)$ is coercive.

Uniqueness: We borrow the idea in [55]. Suppose that g_1^* and g_2^* are both minimizers of $E(g)$. Since $E(g)$ is convex, for any $\theta \in (0, 1)$ we have

$$(B.3) \quad \theta E^*(g_1^*) + (1 - \theta)E(g_2^*) = E(\theta g_1^* + (1 - \theta)g_2^*).$$

Note that each term of $E(g)$ in (2.6) is convex; especially, the first two terms of $E(g)$ are strictly convex with respect to $\mathcal{A}g$ and ∇g , respectively. Therefore (B.3) implies that the following two equalities hold:

$$\begin{aligned} \frac{\theta\lambda}{2} \int_{\Omega} (f - \mathcal{A}g_1^*)^2 dx + \frac{(1-\theta)\lambda}{2} \int_{\Omega} (f - \mathcal{A}g_2^*)^2 dx &= \frac{\lambda}{2} \int_{\Omega} (f - \mathcal{A}(\theta g_1^* + (1-\theta)g_2^*))^2 dx, \\ \frac{\theta\mu}{2} \int_{\Omega} |\nabla g_1^*|^2 dx + \frac{(1-\theta)\mu}{2} \int_{\Omega} |\nabla g_2^*|^2 dx &= \frac{\mu}{2} \int_{\Omega} |\nabla(\theta g_1^* + (1-\theta)g_2^*)|^2 dx. \end{aligned}$$

We thus have $\mathcal{A}g_1^* = \mathcal{A}g_2^*$ and $\nabla g_1^* = \nabla g_2^*$. By the assumption $\text{Ker}(\mathcal{A}) \cap \text{Ker}(\nabla) = \{0\}$, we conclude that $g_1^* - g_2^* = 0$. ■

REFERENCES

- [1] L. AMBROSIO AND V. TORTORELLI, *Approximation of functionals depending on jumps by elliptic functionals via Γ -convergence*, Comm. Pure Appl. Math., 43 (1990), pp. 999–1036.
- [2] L. AMBROSIO AND V. TORTORELLI, *On the approximation of free discontinuity problems*, Boll. Un. Mat. Ital., 6 (1992), pp. 105–123.
- [3] E. BAE, J. YUAN, AND X. TAI, *Simultaneous Convex Optimization of Regions and Region Parameters in Image Segmentation Models*, UCLA CAM Report 11–83, University of California, Los Angeles, 2011.
- [4] L. BAR, T. CHAN, G. CHUNG, M. JUNG, N. KIRYATI, R. MOHIEDDINE, N. SOCHEN, AND L. A. VESE, *Mumford and Shah model and its applications to image segmentation and image restoration*, in Handbook of Mathematical Imaging, Springer, New York, 2011, pp. 1095–1157.
- [5] S. BOYD, N. PARIKH, E. CHU, B. PELEATO, AND J. ECKSTEIN, *Distributed optimization and statistical learning via the alternating direction method of multipliers*, Found. Trends Machine Learning, 3 (2010), pp. 1–122.
- [6] K. BREDIES, K. KUNISCH, AND T. POCK, *Total generalized variation*, SIAM J. Imaging Sci., 3 (2010), pp. 492–526.
- [7] X. BRESSON, S. ESEDOGLU, P. VANDERGHEYNST, J. THIRAN, AND S. OSHER, *Fast global minimization of the active contour/snake model*, J. Math. Imaging Vision, 28 (2007), pp. 151–167.
- [8] E. BROWN, T. CHAN, AND X. BRESSON, *A Convex Relaxation Method for a Class of Vector-Valued Minimization Problems with Applications to Mumford-Shah Segmentation*, UCLA CAM Report 10–43, University of California, Los Angeles, 2010.
- [9] X. CAI, R. CHAN, S. MORIGI, AND F. SGALLARI, *Vessel segmentation in medical imaging using a tight-frame based algorithm*, SIAM J. Imaging Sci., to appear.
- [10] A. CHAMBOLLE, *Image segmentation by variational methods: Mumford and Shah functional and the discrete approximations*, SIAM J. Appl. Math., 55 (1995), pp. 827–863.
- [11] A. CHAMBOLLE, *Finite differences discretization of the Mumford-Shah functional*, M2AN Math. Model. Numer. Anal., 33 (1999), pp. 261–288.
- [12] A. CHAMBOLLE AND G. DALMASO, *Discrete approximation of the Mumford-Shah functional in dimension two*, M2AN Math. Model. Numer. Anal., 33 (1999), pp. 651–672.
- [13] A. CHAMBOLLE AND T. POCK, *A first-order primal-dual algorithm for convex problems with applications to imaging*, J. Math. Imaging Vision, 40 (2011), pp. 120–145.
- [14] T. F. CHAN, S. ESEDOGLU, AND M. NIKOLOVA, *Algorithms for finding global minimizers of image segmentation and denoising models*, SIAM J. Appl. Math., 66 (2006), pp. 1632–1648.
- [15] T. CHAN, A. MARQUINA, AND P. MULET, *High-order total variation-based image restoration*, SIAM J. Sci. Comput., 22 (2000), pp. 503–516.

- [16] T. CHAN AND L. A. VESE, *Active contours without edges*, IEEE Trans. Image Process., 10 (2001), pp. 266–277.
- [17] R. H. CHAN AND M. K. NG, *Conjugate gradient methods for Toeplitz systems*, SIAM Rev., 38 (1996), pp. 427–482.
- [18] G. DAVID, *Singular Sets of Minimizers for the Mumford-Shah Functional*, Progr. Math., Birkhäuser-Verlag, Basel, Switzerland, 2005.
- [19] B. DONG, A. CHIEN, AND Z. SHEN, *Frame based segmentation for medical images*, Commun. Math. Sci., 32 (2010), pp. 1724–1739.
- [20] I. EKELAND AND R. TÉMAM, *Convex Analysis and Variational Problems*, Classics in Appl. Math. 28, SIAM, Philadelphia, 1999.
- [21] S. ESEDOĞLU AND Y. TSAI, *Threshold dynamics for the piecewise constant Mumford-Shah functional*, J. Comput. Phys., 211 (2006), pp. 367–384.
- [22] L. C. EVANS, *Partial Differential Equations*, American Mathematical Society, Providence, RI, 1998.
- [23] M. FIGUEIREDO AND J. BIOCAS-DIAS, *Restoration of Poissonian images using alternating direction optimization*, IEEE Trans. Image Process., 19 (2010), pp. 3133–3145.
- [24] E. FRANCHINI, S. MORIGI, AND F. SGALLARI, *Segmentation of 3D tubular structures by a PDE-based anisotropic diffusion model*, in Proceedings of MMCS 2008, M. Dæhlen, M. S. Floater, T. Lyche, J.-L. Merrien, K. Morken, and L. L. Schumaker, eds., Lecture Notes in Comput. Sci. 5862, Springer-Verlag, Berlin, Heidelberg, 2010, pp. 224–241.
- [25] M. GOBBINO, *Finite difference approximation of the Mumford-Shah functional*, Comm. Pure Appl. Math., 51 (1998), pp. 197–228.
- [26] T. GOLDSTEIN AND S. OSHER, *The split Bregman method for L_1 -regularized problems*, SIAM J. Imaging Sci., 2 (2009), pp. 323–343.
- [27] L. GRADY AND C. ALVINO, *Reformulating and optimizing the Mumford-Shah functional on a graph—A faster, lower energy solution*, in Proceedings of the European Conference on Computer Vision (ECCV), 2008, pp. 248–261.
- [28] J. HARTIGAN AND M. WANG, *A K -means clustering algorithm*, Appl. Statist., 28 (1979), pp. 100–108.
- [29] B. HE AND X. YUAN, *Convergence analysis of primal-dual algorithms for a saddle-point problem: From contraction perspective*, SIAM J. Imaging Sci., 5 (2012), pp. 119–149.
- [30] T. KANUNGO, D. MOUNT, N. NETANYAHU, C. PIATKO, R. SILVERMAN, AND A. WU, *An efficient k -means clustering algorithm: Analysis and implementation*, IEEE Trans. Pattern Anal. Mach. Intell., 24 (2002), pp. 881–892.
- [31] G. KOEPFLER, C. LOPEZ, AND J. M. MOREL, *A multiscale algorithm for image segmentation by variational method*, SIAM J. Numer. Anal., 31 (1994), pp. 282–299.
- [32] J. LELLMANN AND C. SCHNÖRR, *Continuous multiclass labeling approaches and algorithms*, SIAM J. Imaging Sci., 4 (2011), pp. 1049–1096.
- [33] F. LI, M. NG, T. Y. ZENG, AND C. SHEN, *A multiphase image segmentation method based on fuzzy region competition*, SIAM J. Imaging Sci., 3 (2010), pp. 277–299.
- [34] F. LI, C. SHEN, AND C. LI, *Multiphase soft segmentation with total variation and H^1 regularization*, J. Math. Imaging Vision, 37 (2010), pp. 98–111.
- [35] J. LIE, M. LYSAKER, AND X. TAI, *A binary level set model and some applications to Mumford-Shah image segmentation*, IEEE Trans. Image Process., 15 (2006), pp. 1171–1181.
- [36] F. LINDSTEN, H. OHLSSON, AND L. LJUNG, *Just Relax and Come Clustering! A Convergence of k -Means Clustering*, Technical Report LiTH-ISY-R-2992, Department of Electrical Engineering, Linköping University, Linköping, Sweden, 2011.
- [37] M. LYSAKER, A. LUNDERVOLD, AND X. TAI, *Noise removal using fourth-order partial differential equation with applications to medical magnetic resonance images in space and time*, IEEE Trans. Image Process., 12 (2003), pp. 1579–1590.
- [38] J. MACQUEEN, *Some methods for classification and analysis of multivariate observations*, in Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, University of California Press, 1967, pp. 281–297.
- [39] B. MERRIMAN, J. BENICE, AND S. OSHER, *Diffusion Generated Motion by Mean Curvature*, UCLA CAM Report 92-18, University of California, Los Angeles, 1992.
- [40] B. MERRIMAN, J. BENICE, AND S. OSHER, *Motion of multiple junctions: A level set approach*, J. Comput. Phys., 112 (1994), pp. 334–363.

- [41] M. MORINI AND M. NEGRI, *Mumford-Shah functional as Γ -limit of discrete Perona-Malik energies*, Math. Models Methods Appl. Sci., 13 (2003), pp. 785–805.
- [42] D. MUMFORD AND J. SHAH, *Boundary detection by minimizing functionals*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1985, pp. 22–26.
- [43] D. MUMFORD AND J. SHAH, *Optimal approximations by piecewise smooth functions and associated variational problems*, Comm. Pure Appl. Math., 42 (1989), pp. 577–685.
- [44] T. POCK, D. CREMERS, H. BISCHOF, AND A. CHAMBOLLE, *An algorithm for minimizing the Mumford-Shah functional*, in Proceedings of the 12th IEEE International Conference on Computer Vision, 2009, pp. 1133–1140.
- [45] T. POCK, D. CREMERS, A. CHAMBOLLE, AND H. BISCHOF, *A convex relaxation approach for computing minimal partitions*, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2009, pp. 810–817.
- [46] L. RUDIN, S. OSHER, AND E. FATEMI, *Nonlinear total variation based noise removal algorithms*, Phys. D, 60 (1992), pp. 259–268.
- [47] B. SANDBERG, S. KANG, AND T. CHAN, *Unsupervised multiphase segmentation: A phase balancing model*, IEEE Trans. Image Process., 19 (2010), pp. 119–130.
- [48] B. SHAFEI AND G. STEIDL, *Segmentation of images with separating layers by fuzzy c-means and convex optimization*, J. Visual Commun. Image Represent., 23 (2012), pp. 611–621.
- [49] A. TSAI, A. YEZZI, AND A. WILLSKY, *Curve evolution implementation of the Mumford-Shah functional for image segmentation, denoising, interpolation, and magnification*, IEEE Trans. Image Process., 10 (2001), pp. 1169–1186.
- [50] L. VESE AND T. CHAN, *A multiphase level set framework for image segmentation using the Mumford and Shah model*, Int. J. Comput. Vis., 50 (2002), pp. 271–293.
- [51] Y. YOU AND M. KAVEH, *Fourth-order partial differential equation for noise removal*, IEEE Trans. Image Process., 9 (2000), pp. 1723–1730.
- [52] J. YUAN, E. BAE, X. TAI, AND Y. BOYKOV, *A Study on Continuous Max-Flow and Min-Cut Approaches*, Technical Report CAM 10-61, UCLA, Los Angeles, 2010.
- [53] J. YUAN, E. BAE, X. TAI, AND Y. BOYKOV, *A study on continuous max-flow and min-cut approaches*, in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2010, pp. 2217–2224.
- [54] J. YUAN, E. BAE, X. TAI, AND Y. BOYKOV, *A continuous max-flow approach to Potts model*, in ECCV 2010: Proceedings of the 11th European Conference on Computer Vision, Springer, Berlin, 2010, pp. 332–345.
- [55] T. ZENG, X. LI, AND M. NG, *Alternating minimization method for total variation based wavelet shrinkage model*, Commun. Comput. Phys., 8 (2010), pp. 976–994.