Multiple-Region Segmentation Without Supervision by Adaptive Global Maximum Clustering

Sunhee Kim and Myungjoo Kang

***Abstract*—In this paper, we propose a new method of segmenting an image into several sets of pixels with similar intensity values called regions. A multiple-region segmentation problem is unstable because the result considerably depends on the number of regions given *a priori*. Therefore, one of the most important tasks in solving the problem is automatically finding the number of regions. The method we propose is able to find the reasonable number of distinct regions not only for clean images but also for noisy ones. Our method is made up of two procedures. First, we develop the adaptive global maximum clustering. In this procedure, we deal with an image histogram and automatically obtain the number of significant local maxima of the histogram. This number indicates the number of different regions in the image. Second, we derive a simple and fast calculation to segment an image composed of distinct multiple regions. Then, we split an image into multiple regions according to the previous procedure. Finally, we show the efficiency of our method by comparing it with other previous methods.**

***Index Terms*—Adaptive global maximum clustering (AGMC), image histogram, k-means clustering, multiple-region segmentation.**

I. Introduction

IMAGE segmentation is the process of labeling all pixels in accordance with certain characteristics as one of the fundamental tasks in computer vision. The set of pixels with the same label forms a region. The image is classified into several regions by traits such as intensity, shape, or texture so that the pixels in one region have a similar trait. Various models and successful methods have been proposed to solve the image segmentation problem.

Our approach is based on one of them, i.e., the Mumford–Shah minimal partition model [1]. In [2], Chan and Vese applied the level set framework [3] to effectively minimize the Mumford–Shah functional, and thus, they could automatically manage topological changes of the set of zero level. Moreover, they could detect blurry boundaries of regions or the boundaries of tiny regions that are not defined by image gradient. However, it is inevitable that the computational cost is high because their minimization problem results in solving a nonlinear parabolic partial differential equation. To rectify this deficiency, Gibou and Fedkiw [4] suggested a fast algorithm to solve the Chan–Vese model [2] under appropriate assumptions. Dealing with noisy images, they first treated nonlinear diffusion preprocessing to denoise the given image and then segment the denoised image, whereas the Chan–Vese model performs segmentation and denoising at the same time. Gibou and Fedkiw modified the Chan–Vese model to the k-means algorithm, particularly $k = 2$, and improved the speed of the segmentation procedure by dropping the length term in the Chan–Vese model. There is another model, i.e., the Song–Chan model [5], which was devised to improve the computation of the Chan–Vese model, and they computed the energy in the Chan–Vese model directly. Because they used the full terms of the Chan–Vese model, they could remove noise in segmentation procedure. However, those methods in [2], [4], and [5] divided the given image into only two distinct regions and would not work well for the multiphase segmentation.

Later, Chan and Vese expanded their two-phase active contours model to multiphase level set framework in [6] and [7]. In the above works, m level set functions and 2^m constraints are required to detect 2^m distinct regions. This causes the issues of the lack of storage and computational cost. In [8], Lie et al. developed the piecewise constant level set method (PCLSM) for the multiphase segmentation problem. The remarkable property of this approach is that it uses only one level set function to describe all the regions. Inspired by [4] and [8], we want to propose a method to solve the multiphase segmentation problem by using one function to identify all the phases. However, in [8], the number n of distinct phases plays an important role in the segmentation result, and this means that the PCLSM is very sensitive to the number n fixed *a priori*. In practical experiments, we do not know the correct number n beforehand, and therefore, we have tried to devise a method to identify distinct regions without input number n.

In [17] and [18], the authors suggested that one way to find the best number and thresholds of classes of the image histogram is to split an image into multiple regions. They improved time complexity of dynamic programming and separated a given histogram into more than two classes. Their approach can estimate the number of classes by observing the change of the values of the objective function. Their results in [17] and [18] are the same as those of original dynamic programming. Since our interest is threshold values rather than time complexity, we focused on the objective functions, which is essential in finding the number of

Manuscript received March 22, 2011; revised August 15, 2011 and November 10, 2011; accepted November 17, 2011. Date of publication December 09, 2011; date of current version March 21, 2012. This work was supported in part by the Basis Science Research Program and in part by the Mid-career Researcher Program through the National Research Foundation of Korea under Grant 2011-0002404 and Grant 2010-0000158 funded by the Ministry of Education, Science, and Technology of the Korean Government. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Gang Hua.

The authors are with the Department of Mathematical Sciences, Seoul National University, Seoul 151-747, Korea (e-mail: sunny068@snu.ac.kr; mkang@snu.ac.kr).

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.2011.2179058

1057-7149/$26.00 © 2011 IEEE

classes. There is another method, i.e., a phase balancing model [16], to get the number of different phases in the image. The authors introduced an inverse scale term to define new regularization and used a brute-force algorithm. They employed the image itself to get the number of different phases and segmented the image at the same time. The main difference between our model and the phase balancing model is that we use the histogram as in [17] and [18] instead of the image to estimate the number of different regions. We will show some numerical results for the above two methods to compare with our method.

This paper is organized as follows. In Section II, we will first review two-phase models and multiphase models for image segmentation. Then, we will describe the new method composed of two procedures, namely, the segmentation procedure and the adaptive global maximum clustering (AGMC) procedure, in Section III. Next, we will show the numerical results to prove the efficiency of our method, including comparison with two previous methods. Finally, Section V is the conclusion of paper.

II. Two-Phase Models and Multiphase Models for Image Segmentation

Let $u_0 : \Omega \to \mathbb{R}$ be a 2-D gray level image with or without noise. Assume that the intensities of the image change smoothly away from the discontinuities and vary rapidly near them. We call a set of discontinuities *an edge*. The set of pixels, which are separated by the edges, is called *a phase* or *a region*. The question for image segmentation is how to reasonably divide the given image into edges and regions. Mumford and Shah proposed the minimization problem in [1]

$$\inf_{u,C}\left\{E^{MS}(u,C)=\mu\,\text{length}(C)+\int_{\Omega}|u-u_0|^2dx\,dy+\nu\int_{\Omega-C}|\nabla u|^2dx\,dy\right\} \quad (1)$$

where μ and ν are nonnegative parameters to decompose a given image into optimal piecewise smooth approximations. For the minimizers u and C of problem (1), C is an edge set and $\Omega = (\bigcup_i \Omega_i) \cup C$ so that u is smooth in each region Ω_i.

In the simplest case, if we consider a constant function u instead of a smooth function, u is a constant c_i on each region Ω_i of $\Omega \setminus C$, i.e., $u = \sum_{i=1}^{n} c_i \chi_{\Omega_i}$. Thus, problem (1) can be reduced to the minimal partition problem or the piecewise constant Mumford–Shah problem

$$\inf_{u,C}\left\{E^{PMS}(u,C)=\mu\,\text{length}(C)+\sum_{i=1}^{n}\int_{\Omega_i}|c_i-u_0|^2dx\,dy\right\}. \quad (2)$$

In [2], Chan and Vese used the level set framework to solve the above problem (2) in the case of $n = 2$. They defined an edge C by a zero-level set of a Lipschitz function ϕ, i.e., $C = \{(x, y) \in \Omega : \phi(x, y) = 0\}$. Thus, problem (2) was reformulated by a level set function ϕ, and they finally solved a nonlinear parabolic equation obtained by minimizing the functional $E^{CV}(c_1, c_2, \phi)$

$$\inf_{c_1,c_2,\phi}\left\{E^{CV}(c_1,c_2,\phi)=\nu\int_{\Omega}|\nabla H(\phi)|\,dx+\int_{\Omega}|c_1-u_0|^2H(\phi)dx+\int_{\Omega}|c_2-u_0|^2\,(1-H(\phi))\,dx\right\} \quad (3)$$

where $H(\phi)$ is the Heaviside function, which is equal to 1 if $\phi \geq 0$ and equal to 0 if $\phi < 0$. One image can be divided into two regions, i.e., inside and outside of ϕ, according to the sign of ϕ as $\text{inside}(C) = \{(x, y) \in \Omega : \phi(x, y) > 0\}$ and $\text{outside}(C) = \{(x, y) \in \Omega : \phi(x, y) < 0\}$. If we keep ϕ fixed and minimize E^{CV} with respect to c_1 and c_2, we have

$$c_1 = \text{average}(u_0), \text{inside}(C)$$
$$c_2 = \text{average}(u_0), \text{outside}(C).$$

Thus, an optimal piecewise constant function u splits the image into only two distinct regions of which each intensity has the optimal value of c_1 or c_2. In addition, if we minimize E^{CV} with respect to ϕ by keeping c_1 and c_2 fixed, we have the following nonlinear partial differential equation:

$$\phi_t = \delta_\varepsilon(\phi)\left\{\nu\nabla\cdot\left(\frac{\nabla\phi}{|\nabla\phi|}\right)-(u_0-c_1)^2+(u_0-c_2)^2\right\} \quad (4)$$

where $\delta_\varepsilon(\cdot)$ is a regularized version of the delta function.

In [4], Gibou and Fedkiw suggested a fast algorithm to solve the Chan–Vese model for images without noise. If there is no noise in the given image, the first term related to the length of contours is no longer important, and thus, we can ignore that term. Ultimately, the segmentation problem is to solve the following simple ordinary differential equation (ODE):

$$\phi_t = -(u_0-c_1)^2+(u_0-c_2)^2 \quad (5)$$

which is equivalent to the two-means clustering algorithm. They merely solved (5) instead of solving the complicated nonlinear partial differential equation (4) corresponding to problem (3).

In [5], Song and Chan also proposed a fast method to solve the Chan–Vese model (3) and calculated the energy in (3) directly to decide the sign of ϕ, which is the level set function that determines inside or outside contours. They do not need to compute the corresponding Euler–Lagrange equation and use the full energy including the length term. They therefore can remove noise in the given image very fast. However, those methods split the given image into only two regions and they would not work well for the multiphase segmentation.

In order to segment the image into more than three regions, Chan and Vese expanded their two-phase active contours model to a multiphase level set framework in [6] and [7]. However, these methods require m level set functions and 2^m phases to detect 2^m regions. To reduce the storage, Lie et al. [8] exploited the PCLSM for problem (2). The noticeable feature of the PCLSM is to use only one piecewise constant function to recognize all the regions. The authors assumed that a piecewise constant level

set function ϕ satisfies $\phi = i$ in Ω_i for $i = 1, 2, \cdots, n$. They considered the following characteristic functions of ϕ:

$$\psi_i(\phi) = \frac{1}{\alpha_i} \prod_{j=1, j\neq i}^{n} (\phi - j) \quad \text{and} \quad \alpha_j = \prod_{k=1, k\neq j}^{n} (i - k)$$

where ψ_i, for $1 \leq i \leq n$, is a polynomial function of degrees $n-1$ in ϕ, and $\psi_i = 1$ in Ω_i and $\psi_i = 0$ in $\Omega \setminus \Omega_i$. For $n > 2$, problem (2) is minimized for a fixed C when $c_i = \text{average}(u_0)$ in Ω_i, as in the case of $n = 2$. With these characteristic functions ψ_i, they expressed u and the arc length of $\partial\Omega_i$ as follows:

$$\begin{aligned} u(\mathbf{c}, \phi) &= \sum_{i=1}^{n} c_i \psi_i(\phi) \\ \text{length}(\partial\Omega_i) &= \int_{\Omega} |\nabla \psi_i| dx. \end{aligned} \tag{6}$$

To avoid a vacuum and an overlap among the different regions, they introduced a polynomial $K(\phi)$ of degree n

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

By using $K(\phi) = 0$ as a constraint, the constrained minimization problem is presented by

$$\min_{\mathbf{c}, \phi, K(\phi)=0} \left\{ \frac{1}{2} \int_{\Omega} |u - u_0|^2 dx + \nu \sum_{i=1}^{n} \int_{\Omega} |\nabla \psi_i(\phi)| \, dx \right\} \tag{7}$$

where $\mathbf{c} = (c_1, c_2, \ldots, c_n)$ and $u = u(\mathbf{c}, \phi)$ from (6), and can be solved by the augmented Lagrangian method. For the augmented Lagrangian functional, they used the gradient descent method in [8]. Later, the speed of the convergence of the above problem (7) was improved by using Newton's method in [9].

In the multiple-region segmentation problem, if the number of regions to be classified is given *a priori*, we could obtain the almost desirable result. However, in real applications, we do not know the number beforehand. In addition, the result is very sensitive to the given number. Although we experiment with the same image, the results would be dissimilar if different numbers are given, as shown in Figs. 1 and 2. If we use $n = 4$, we can get a reasonable result in Fig. 1. However, if we use $n = 5$, we detect an unnecessary part in Fig. 2(e) and thus cannot segment the image correctly. It is therefore essential to automatically find the number of distinct regions.

In [17] and [18], the authors improved the time complexity of dynamic programming and separated a given histogram into more than two classes. For the dynamic programming approach, it is required to divide the original problem into subproblems, and then obtain the original solution from the solutions of the subproblems. They also accounted for how to find the correct number of classes in the histogram. A histogram $\mathbf{h} = \{h(l)\}_{l=1}^{L}$ of the image designates the distribution of gray levels in the image, and $h(l)$ is the number of occurrences of each gray level $l = 1, 2, \cdots, L$. For convenience, a normalized histogram

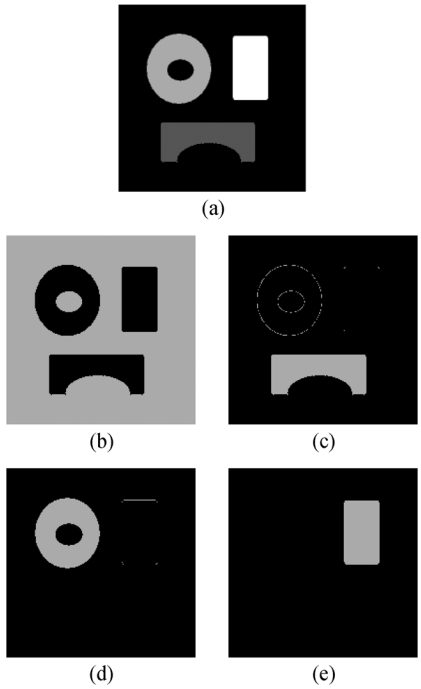

Fig. 1. (a) Segmentation result. (b)–(e) Gray parts are regions. We used $n = 4$ and obtained four regions, which is the desired result.

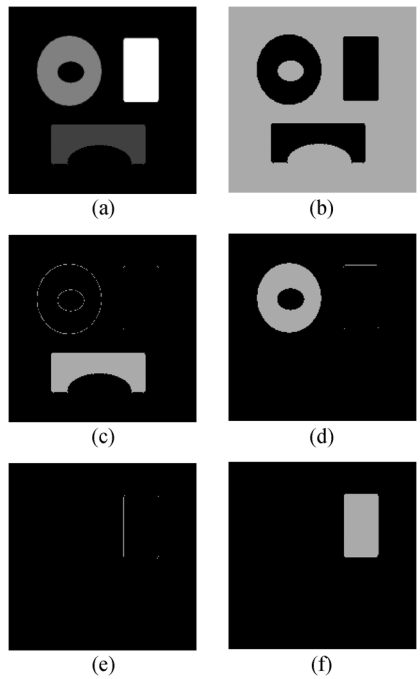

Fig. 2. (a) Segmentation result. (b)–(f) Gray parts are regions. We used $n = 5$ and obtained five regions. The unnecessary part (e) came out as one region.

$p(l) = h(l)/N$, where $N = \sum_{l=1}^{L} h(l)$ is the total number of image pixels, is utilized. The multilevel thresholding method finds optimal thresholds, $l_1, l_2, \ldots, l_{M-1}$, which classify an interval $[1, L]$ into M classes, $C_1, C_2, \ldots, C_M$. Hence, for $m = 1, \ldots, M$, each class C_m is determined by two adjacent thresholds, i.e., $C_m = (l_{m-1}, l_m]$, where $l_0 = 0$ and $l_M = L$.

For a given class $(a, b]$, weight $\omega(a, b]$, mean $\mu(a, b]$, and variance $\sigma^2(a, b]$ are defined as

$$\omega(a,b] = \sum_{l=a+1}^{b} p(l)$$
$$\mu(a,b] = \sum_{l=a+1}^{b} \frac{p(l)\cdot l}{\omega(a,b]}$$
$$\sigma^2(a,b] = \sum_{l=a+1}^{b} \frac{p(l)\cdot (l-\mu(a,b])^2}{\omega(a,b]}.$$

The multilevel thresholding problem is described by minimizing or maximizing an objective function E_M

$$\arg\min_{1\leqslant l_1<\cdots<l_{M-1}<L} E_M(l_1,\cdots,l_{M-1})$$

or

$$\arg\max_{1\leqslant l_1<\cdots<l_{M-1}<L} E_M(l_1,\cdots,l_{M-1}).$$

A prerequisite for the dynamic programming approach is that a class cost f in an objective function has to depend only on the boundary of class, called by l_{m-1} and l_m, as follows:

$$E_M(l_1,\cdots,l_{M-1}) = \sum_{m=1}^{M} f(l_{m-1}, l_m]$$

where $1q \leqslant l_1 < \cdots < l_{M-1} < L$.

First, the subproblem for a gray level t at a stage s is written by

$$E_s^*(t) = \min_{1\leqslant l_1<\cdots<l_{s-1}<t} E_s(t)$$
$$E_s(t) = \sum_{m=1}^{s-1} f(l_{m-1}, l_m] + f(l_{s-1}, t].$$

The above subproblem can be transformed into a recursive problem

$$E_s^*(t) = \min_{s-1\leqslant l_{s-1}<t} \left\{E_{s-1}^*(l_{s-1}) + f(l_{s-1}, t]\right\}$$

by rewriting the objective function $E_s(t)$ of the subproblem. They then have the recursive optimal cost described as

$$E_s^*(t) = \begin{cases} f(0,t], & \text{if } s=1 \\ \min\limits_{s-1\leqslant l_{s-1}<t} \left\{E_{s-1}^*(l_{s-1}) + f(l_{s-1}, t]\right\}, & \text{if } s>1. \end{cases} \quad (8)$$

The aim of the multilevel thresholding method is to explore the optimal path to minimize (8) when $s = M$ and $t = L$. The authors in [17] and [18] exploited a *trellis structure,* as shown in Fig. 3, and elucidated how to find optimal thresholds and implement the algorithm. The blue dots are called nodes and represented by (s, t) where $E_s^*(t)$ are computed. To avoid the thresholds at the same points, they considered the following constraint:

At stage s, different $s-1$ thresholds must be allotted

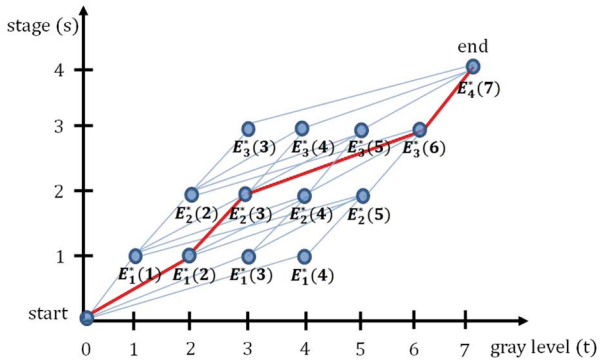

Fig. 3. Trellis structure. Blue dots are nodes, light blue lines represent possible paths, and red lines are optimal paths obtained from backtracking (in online version).

and thus the whole structure is rhomboid in shape. For each node (s, t) in the trellis, two values, namely, $E_{\text{val}}(s, t)$ and $E_{\text{pos}}(s, t)$, are defined. $E_{\text{val}}(s, t)$ denotes the optimal partial sum up to the node (s, t), and $E_{\text{pos}}(s, t)$ represents the position of the best node to arise from. Both values are computed as follows:

$$E_{\text{val}}(s,t) = \min_{s-1\leqslant l_{s-1}<t} \left\{E_{s-1}^*(l_{s-1}) + f(l_{s-1}, t]\right\}$$
$$= E_s^*(t),$$
$$E_{\text{pos}}(s,t) = \arg\min_{s-1\leqslant l_{s-1}<t} \left\{E_{s-1}^*(l_{s-1}) + f(l_{s-1}, t]\right\}.$$

In the first stage $(s = 1)$, only the values of cost function f are needed in the computation of the value $E_{\text{val}}(1, t)$ by (6) and $E_{\text{pos}}(1, t) = 0$. In stage $1 < s < M$, in accordance with the constraint, the authors calculated the values $E_{\text{val}}(s, t)$ and $E_{\text{pos}}(s, t)$ for nodes, which are one stage below and to the left of the current node (s, t). In the final stage $(s = M)$, for the last node $(s, t) = (M, L)$, only the $E_{\text{val}}(M, L)$ and $E_{\text{pos}}(M, L)$ values are computed. After evaluating optimal values $E_{\text{val}}(s, t)$ and $E_{\text{pos}}(s, t)$ for all nodes (s, t), which form a rhomboid, they backtrack the optimal path by starting at the final stage. To sum up, the multilevel thresholding problem is solved by the following procedure:

Step 1) $(s = 1)$ Compute $E_{\text{val}}(1, t) = f(0, t]$ and $E_{\text{pos}}(1, t) = 0$ for $t = 1, 2, \ldots, L - M + 1$.

Step 2) $(s = 2, \ldots, M)$ At stage $s = 2, \ldots, M - 1$, compute

$$E_{\text{val}}(s,t) = \min_{s-1\leqslant l<t} \{E_{\text{val}}(s-1, l) + f(l, t]\} \quad (9)$$
$$E_{\text{pos}}(s,t) = \arg\min_{s-1\leqslant l<t} \{E_{\text{val}}(s-1, l) + f(l, t]\} \quad (10)$$

for $t = s, s+1, \ldots, L - M + s$.At stage $s = M$, compute only $E_{\text{val}}(M, L)$ and $E_{\text{pos}}(M, L)$ as in (9) and (10).

Step 3) (backtracking) With $l_s^* = L$, search $l_{s-1}^* = E_{\text{pos}}(s, l_s^*)$ for $s = M, M-1, \ldots, 2$.

They examined four kinds of cost functions, which are proposed by Kapur, Otsu, Kittler and Illingworth, and Li. Moreover, they indicated a way to find the best number of classes of

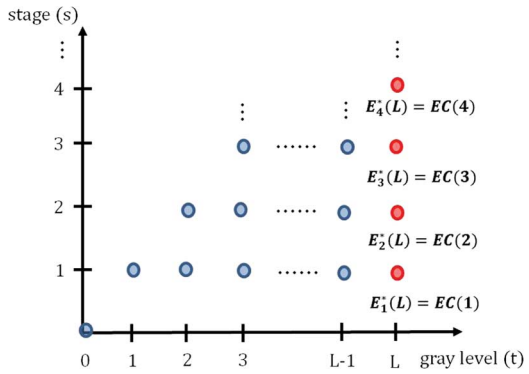

Fig. 4. Construction of the end cost functions. Red circles are nodes where each end cost function is computed in each stage.

the histogram and discovered that the cost function of Kittler and Illingworth is most appropriate for finding the number.

An end cost function $EC(s)$ is defined as the optimal partial sum up to the end node (s, L) for each stage s, as illustrated in Fig. 4

$$\begin{aligned} EC(s) &= E_{\text{val}}(s, L) \\ &= E_s^*(L) \\ &= \min_{s-1 \leqslant l < L} \{E_{\text{val}}(s-1, l) + f(l, L]\} \end{aligned}$$

for $s = 1, \ldots, M$.

The other function G_α to measure differences to end cost function was introduced

$$G_\alpha(s) = 4\frac{\alpha(s-1)}{L}\log\left(\frac{L}{4}\right) + \frac{v(s)}{L}\log\left(\frac{\frac{1}{12}\left(v^2(s)-1\right)}{\left(\frac{v(s)}{L}\right)^2}\right)$$

$$v(s) = L - 2\alpha(s-1)$$

where α controls the slope of $G_\alpha(s)$. The best number M of classes can be obtained from

$$M = \arg\max_s CFF(s) \quad CFF(s) = EC(s) - G_\alpha(s)$$

with $\alpha = 7$.

In [16], a phase balancing model, which is a model to automatically find the number of distinct phases of an image without a histogram, was suggested. The aim of the model is to approximate the given image u_0 to the combination u of distinct phases as follows:

$$u = \sum_{i=1}^{n} c_i \chi_i$$

where χ_i denotes the ith phase, which is a characteristic function, and c_i is the average intensity over a phase χ_i. The authors introduced the inverse scale term s_i to define new regularization

$$s_i = \frac{P(\chi_i)}{|\chi_i|}$$

where $P(\chi_i)$ is the perimeter of a phase χ_i, and $|\chi_i|$ counts the number of pixels in χ_i. The proposed problem is to minimize the following functional:

$$\begin{aligned} &E^{PB}[n, \chi_i, c_i | u_0] \\ &\quad = \mu\left(\sum_{i=1}^{n} \frac{P(\chi_i)}{|\chi_i|}\right)\sum_{i=1}^{n} P(\chi_i) + \sum_{i=1}^{n} |u_0 - c_i|^2 \chi_i \end{aligned} \quad (11)$$

where μ is a positive parameter that controls the tradeoff between the regularization and fidelity terms and it affects the number of phases.

To minimize (11), they directly compared the difference in the functional when (x, y) moves from one phase χ_p to another phase χ_q

$$\begin{aligned} \triangle_{pq} E^{PB} = \mu(T_p \triangle S + S_q \triangle T) \\ + (u - c_q)^2 \frac{m_q}{m_q + 1} - (u - c_p)^2 \frac{m_p}{m_p - 1} \end{aligned}$$

where $T_p \triangle S + S_q \triangle T$ is the change of the first term in (11). Only if $\triangle_{pq} E^{PB}$ is negative, one moves (x, y) to phase χ_q. They set an initial phase $\chi_1 = \Omega$ and the number of phase $n = 1$. Then, for each $(x, y) \in \Omega$, they check the following criterion:

$$if \ \min_{1 \leqslant q \leqslant n+1, q \neq p} \triangle_{pq} E^{PB} < 0, \text{move } (x, y) \text{ to phase } \chi_{q^*}$$

where $q^* = \arg\min_{1 \leqslant q \leqslant n+1, q \neq p} \triangle_{pq} E^{PB}$, and $n+1$ denotes the new empty phase. They update $n = q^*$, $m_i = |\chi_i|$, and c_i for $i = 1, \ldots, n$ and then iterate this procedure.

III. Proposed Method

Here, we present a new, simple, fast, and unsupervised multiple-region segmentation method. By developing the AGMC method in Section III-A, we are able to find the number of different regions automatically. From the AGMC method, we have not only the number of distinct regions but also subintervals of $[1, L]$ associated with gray levels of distinct regions. The set of subintervals is employed to define an initial ϕ for our segmentation model, which will be introduced in Section III-B and gives a better segmentation result.

A. AGMC

We use an image histogram to get the number of different regions as in [17] and [18]. In most images, there are too many local maxima of image histograms, as shown in the x-axis in Fig. 5. In all the plotted figure, the x-axis denotes the gray levels l and the y-axis denotes the number of occurrences of each gray level $h(l)$. For consistency, we rescaled the range of h as $1 \leq h(l) \leq 256$. All red circle points in Fig. 5(b) and (d) are local maxima. However, we need to find only significant local maxima since those maxima are necessary to discriminate regions.

To extract the significant local maxima, we first search for an interval, including the global maximum of an original histogram, and then fix the interval. We call such a fixed interval *a cluster*. Next, we remove the cluster, gained from the previous searching process, from the original histogram to find another

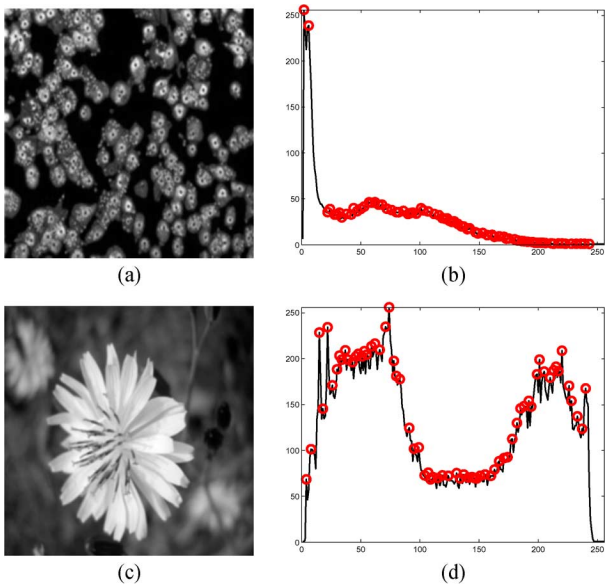

Fig. 5. (a) and (c) are images. (b) and (d) are their image histograms, respectively. Black solid lines are the histograms, and all red circle points are local maxima.

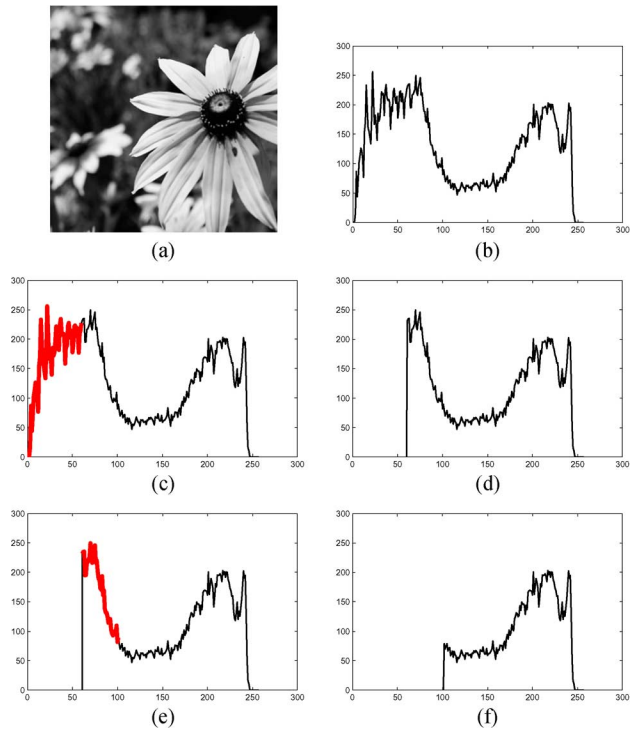

Fig. 6. (a) Original image. (b) Its image histogram. (c) Domain of the thick red line is the first cluster. (d) Reduced histogram, which is the histogram to be used at the second iteration. (e) Domain of the thick red line is the second cluster. (f) Histogram at the third step. Note that the histogram changes every iteration of the AGMC process as (d) and (f).

new cluster. We then search for the new cluster of the reduced histogram and repeat this process until we have the desired result. Fig. 6 explains this process.

Since the histogram changes every iteration, which is precisely the reduced version of the original histogram, the global maximum also adaptively changes every iteration. We therefore call such a maximum *an adaptive global maximum* that corresponds to one of the significant local maxima of the original histogram. This whole process is a series of clustering a gray level interval $[1, L]$ into several subintervals so that the original histogram has the adaptive global maximum over each subinterval. Thus, we call this process *the AGMC* process.

The goal of this process is

$$[1, L] = \left(\bigcup_{i=1}^{n} I_i \right) \bigcup R$$

where each subinterval $I_i = [a_i, b_i]$ is a cluster containing the ith adaptive global maximum, which is the global maximum of the ith histogram. R is the domain of the $(n+1)$th histogram such that $\max_{l \in R} h(l)$ is very small compared with the original histogram, or R is empty. Such very small histogram values are usually useless in the detection of regions.

To find a cluster that is a subinterval with an adaptive global maximum at each iteration, we fix $k = 2$ and repetitively implement the standard k-means clustering under the rules described below. The standard k-means clustering method is a process to solve the following minimization problem:

$$\underset{I_1, \cdots, I_k}{\arg\min} \sum_{j=1}^{k} \sum_{x \in I_j} |x - d_j|^2$$

where $|x - d_j|^2$ is a distance between a data point x and the jth cluster center d_j. The aim of the k-means clustering is to congregate a set of distributed data into k clusters. For an image histogram, the k-means clustering problem is described as

$$\underset{I_1, \cdots, I_k}{\arg\min} \sum_{j=1}^{k} \sum_{l \in I_j} h(l)(l - d_j)^2 \tag{12}$$

where h is an image histogram, and I_j is a subinterval of gray levels such that $\bigcup_{j=1}^{k} I_j = [1, L]$. Each center d_j is computed by

$$d_j = \frac{\sum_{l \in I_j} l \cdot h(l)}{\sum_{l \in I_j} h(l)}, \qquad \text{for } j = 1, \ldots, k. \tag{13}$$

The k value, the total number of clusters, has to be given *a priori*, and the result depends on the initial centers.

We resolve those problems by fixing $k = 2$, setting the initial two centers as starting index of I and ending index of I, and repeating the k-means clustering in the following way. Here, I is the domain of histogram. Note that, in our method, the two-means clustering is applied to the reduced histograms and the original histogram. Thus, the starting and the ending indexes are changeable, not fixed as 1 and 256. Once we implement the two-means clustering, the histogram is divided into two clusters I_1 and I_2 since $k = 2$ is chosen. Then, we have two maxima, i.e., one is obtained in I_1 and the other is obtained in I_2. It is clear that if the maximum value of the histogram in one cluster I_1 is larger than the maximum value of the histogram in the other cluster I_2, then cluster I_1 contains the global maximum of the histogram. This gives the first rule.

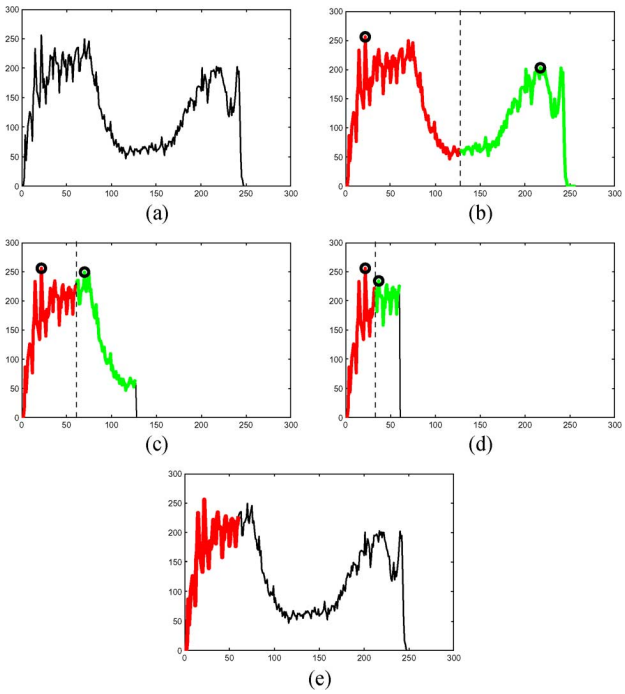

Fig. 7. Process to find the first cluster of the original histogram by the *Rule 1* and *Rule 2*. (a) Original image histogram. (b) The domains of the thick red line on the left side and the thick green line on the right side are clusters I_1 and I_2 called the red cluster and the green cluster, respectively. Each black circle is the maximum of the histogram over each cluster. We choose the red cluster I_1 by *Rule 1*. (c) Another new two clusters by applying the two-means clustering to the histogram over the red cluster of (b). We choose the domain of the thick red line, the red cluster (in online version), as a new cluster by *Rule 1*. (d) New two clusters by applying the two-means clustering to the histogram over the red cluster of (c). (e) First cluster of the original histogram by *Rule 2*.

Rule 1) Choose a cluster

$$I^* = \begin{cases} I_1, & \text{if } \max_{l\in I_1} h(l) > \max_{l\in I_2} h(l) \\ I_2, & \text{otherwise.} \end{cases}$$

Then, we again perform the two-means clustering over the chosen cluster and repeat this process until *Rule 2* holds.

Rule 2) $$\left|\arg\max_{l\in I_1} h(l) - \arg\max_{l\in I_2} h(l)\right| < \sigma.$$

Parameter σ designates the least difference in intensities of distinct regions, which guarantees that the regions with similar intensities are not split for some large σ. If σ is very small, we can even separate regions with similar intensities. With the smaller σ, we get the larger number of regions. Thus, this parameter has an effect on the number of different regions.Every time *Rule 2* is satisfied, we get a cluster, which is a subinterval including one of the significant local maxima of the original histogram. Fig. 7 shows an illustration of a process to find a cluster by *Rule 1* and *Rule 2*. Since we desire to get only the significant local maxima of the original histogram, we have to stop the AGMC procedure if *Rule 3* holds.

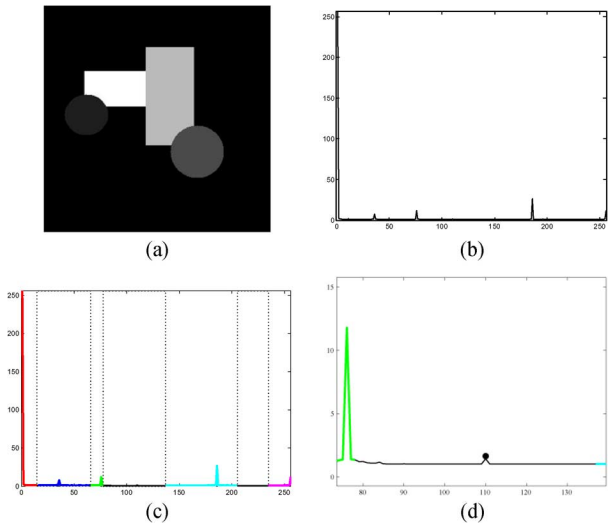

Fig. 8. (a) Original image. (b) Image histogram. (c) Fourth and sixth blocks are the remaining histograms after five iterations, and the other blocks are clusters. (d) Magnified version of the histogram (c) over the interval [70, 145]. The remaining histogram is too small compared with the original histogram and usually useless in the detection of regions.

Rule 3) $$\max h^i < \omega \,\text{mean}(h^0)$$

where h^i is the ith histogram, which is the reduced histogram after the $(i-1)$ iterations, and h^0 is the original histogram. *Rule 3* signifies that h^i is too small compared with the original histogram and such an h^i is usually useless in the detection of regions. This prevents us from finding small local maxima, for example, the black dot in Fig. 8(d) , which is equal to $h(110)$. We cannot even observe any variation of the histogram near $l = 110$ in Fig. 8(c). This means that small local maxima such as this black dot are negligible in searching for significant local maxima in the original histogram, and *Rule 3* gives the criterion. As ω gets larger, the method would mainly find the larger local maxima of the histogram and such maxima are histogram values expressing main regions in the image.

The algorithm of the AGMC procedure is as follows:

Require: Let $\mathbf{h} = \{h(l)\}_{l=1}^{L}$ be an original image histogram, $L = 256$, $\mathbf{h}^0 = \mathbf{h}$, $k = 2$, $i = 0$, and $I = [1, L]$.

repeat

$i \leftarrow i + 1$ and $I^{\text{old}} \leftarrow I$

repeat

(i) Implement two-means clustering by minimizing (12) for $\mathbf{h}(I) = \{h^0(l) : l \in I\}$. Here, $\mathbf{h}(I)$ means the histogram whose domain is restricted to cluster I.

(ii) Choose a cluster I^* between two clusters, i.e., I_1 and I_2, obtained from (i) according to *Rule 1*.

(iii) $I \leftarrow I^*$

until *Rule 2* holds.

Set a cluster $D_i \leftarrow I^*$ and $I \leftarrow I^{\text{old}} \setminus D_i = I^{\text{old}} \setminus I^*$.

until *Rule 3* holds for $\mathbf{h}(I)$ or $\mathbf{h}(I) = \mathbf{0}$.

B. Segmentation Model

Here, we try to design a simple, fast, and unsupervised multiphase segmentation model by modifying the Mumford–Shah model. Assume that a given 2-D image with or without noise $u_0 : \Omega \to \mathbb{R}$ consists of n distinct regions, i.e., $\Omega = \bigcup_{i=1}^{n} \Omega_i$, where the underlying structure of u_0 is constant in each Ω_i. From the AGMC procedure, we automatically get the number n of distinct regions and the n clusters $\{D_i\}_{i=1}^{n}$. We compute d_i by (13) for each cluster D_i, and these $\{d_i\}_{i=1}^{n}$ are indicators to define initial ϕ. We naturally define an initial ϕ as follows:

$$\phi(x,y) = \begin{cases} 1, & \text{if } u_0(x,y) \le \frac{d_1+d_2}{2} \\ i, & \text{if } \frac{d_{i-1}+d_i}{2} < u_0(x,y) \le \frac{d_i+d_{i+1}}{2} \\ n, & \text{if } \frac{d_{n-1}+d_n}{2} < u_0(x,y) \end{cases} \quad (14)$$

where $2 \le i \le n-1$.

If we ignore noise in the segmentation process, which implies that we consider noise a part of intensities, we can figure out the simplified version of piecewise constant Mumford–Shah functional in (2) as

$$\inf_{c_i} \sum_{i=1}^{n} \int_{\Omega_i} |c_i - u_0|^2 dx. \quad (15)$$

Then, we want to find a unique piecewise constant function of which each constant region represents a different region. Define

$$\psi_i(\phi) = H(\phi - i) - H(\phi - (i+1)), \qquad 1 \le i \le n$$

where $H(\phi)$ is equal to 1 if $\phi \ge 0$ and equal to 0 if $\phi < 0$. Then

$$\psi_i(\phi) = \begin{cases} 1, & \text{if } \phi = i \\ 0, & \text{otherwise.} \end{cases}$$

This means that $\psi_i(\phi)$ is equal to 1 if $(x,y) \in \Omega_i$ and equal to 0 if $(x,y) \in \Omega \setminus \Omega_i$.

Our segmentation model is a modification of the energy in problem (15) with these ψ_i for $1 \le i \le n$

$$\min_{c_i,\phi} \left\{ E(c_i,\phi|u_0) = \int_{\Omega} \sum_{i=1}^{n} |c_i - u_0|^2 \psi_i(\phi) dx = \int_{\Omega} \sum_{i=1}^{n} |c_i - u_0|^2 \{H(\phi - i) - H(\phi - (i+1))\} dx \right\}. \quad (16)$$

If we minimize the energy in (16) with respect to c_i for a fixed ϕ, $1 \le i \le n$, we can easily check that

$$c_i(\phi) = \frac{\int_{\Omega} u_o \{H(\phi - i) - H(\phi - (i+1))\} dx}{\int_{\Omega} \{H(\phi - i) - H(\phi - (i+1))\} dx} = \text{average}(u_0) \text{ in } \Omega_i.$$

For fixed $c_1, c_2, \ldots, c_n$, minimizing F with respect to ϕ, we have the following ODE:

$$\phi_t = |c_1 - u_0|^2 \delta_1 + |c_n - u_0|^2 \delta_{n+1} + \sum_{i=1}^{n-1} \left\{ -|c_i - u_0|^2 + |c_{i+1} - u_0|^2 \right\} \delta_{i+1} \quad (17)$$

where $\delta(z) = (d/dz)H(z)$ and $\delta_j = \delta(\phi - j)$ are equal to 1 if $\phi = j$ and equal to 0 elsewhere. Note that $\delta_{n+1} = 0$ since $1 \le \phi(x,y) \le n$ for all $(x,y) \in \Omega$. From our assumption of the range of ϕ, we can disregard the second term in the ODE (17)

$$\phi_t = |c_1 - u_0|^2 \delta_1 + \sum_{i=1}^{n-1} \left\{ -|c_i - u_0|^2 + |c_{i+1} - u_0|^2 \right\} \delta_{i+1}. \quad (18)$$

In region $\Omega_i = \{(x,y) : \phi(x,y) = i\}$ for $2 \le i \le n$, the ODE (18) is equivalent to

$$\phi_t = -|c_{i-1} - u_0|^2 + |c_i - u_0|^2 \equiv E_i.$$

This is the same form as the ODE in the work by Gibou and Fedkiw [4]. They dealt with the two-region segmentation problem and derived the following ODE:

$$\phi_t = -|c_1 - u_0|^2 + |c_2 - u_0|^2 \equiv F \quad (19)$$

where c_1 and c_2 are the intensity averages in $\{x : \phi(x) = 1\}$ and $\{x : \phi(x) = -1\}$, respectively. In [4], the authors pointed out that they are interested only in the discontinuities of ϕ, which represent the contour, and they thus ignored the ODE (19) entirely. By setting $\phi = 1$ when $F > 0$ and setting $\phi = -1$ when $F < 0$, they obtained the same result by solving ODE (19) entirely.

We observe that the attentive point in their algorithm is not the use of different signs of ϕ, but rather the use of the two different values of ϕ. We thus replace the different signs of ϕ by two different positive values of ϕ. Following the idea of Gibou and Fedkiw, we will set $\phi = \alpha$ when $E_i > 0$ and $\phi = \beta$ when $E_i < 0$, where $\alpha > \beta$. To avoid a vacuum, we assign the value i to α or β, which presents two possibilities to update the value of ϕ

$$\phi = \begin{cases} \alpha = i, & \text{when } E_i > 0 \\ \beta = i - 1, & \text{when } E_i < 0 \end{cases} \quad (20)$$

or

$$\phi = \begin{cases} \alpha = i + 1, & \text{when } E_i > 0 \\ \beta = i, & \text{when } E_i < 0. \end{cases} \quad (21)$$

If we choose (21) to update ϕ, we may violate the range of a level set function ϕ, which is $1 \le \phi \le n$. For example, in region $\Omega_n = \{(x,y) : \phi(x,y) = n\}$, the value of ϕ may have $n+1$ when $E_n > 0$, and this disobeys the range of ϕ. We therefore choose (20) to update ϕ and conclude that in region $\Omega_i = \{(x,y) : \phi(x,y) = i\}$, for $1 \le i \le n$, it is enough to update ϕ only when $E_i < 0$.

For this reason, we do not need to update the value of ϕ in region Ω_1. Because in region $\Omega_1 = \{(x,y) : \phi(x,y) = 1\}$, we just solve

$$\phi_t = |c_1 - u_0|^2 \equiv E_1.$$

Since E_1 is always positive, there is no need to update the value of $\phi(x,y)$ in Ω_1. From the definition (14) of initial ϕ, we note that

$$\begin{aligned}\Omega_1 &= \{(x,y) : \phi(x,y) = 1\} \\ &= \left\{(x,y) : u_0(x,y) \le \frac{d_1+d_2}{2}\right\}. \\ &\ni \{(x,y) : u_0(x,y) = \min(u_0)\}.\end{aligned}$$

This means that the pixels in the region where u_0 has a minimum value keep $\phi = 1$ as time goes.

The main ODE associated with our problem (16) is

$$\phi_t = -|c_{i-1} - u_0|^2 + |c_i - u_0|^2, \qquad \text{for} \quad 2 \le i \le n \quad (22)$$

in region $\Omega_i = \{(x,y) : \phi(x,y) = i\}$. Now, we derive the simple and fast calculation of the above problem (22) in the following lemmas.

Lemma 1: Let u_0 be a given grayscale image defined in an image domain $\Omega \subset \mathbb{R}^2$. Assume that the underlying structure of the given image is being composed of n distinct regions, i.e., $\Omega = \bigcup_{i=1}^n \Omega_i$. Define a piecewise constant level set function ϕ such that ϕ satisfies (14). *Then, for $2 \le i \le n$*

$$\phi_t = -|c_{i-1} - u_0|^2 + |c_i - u_0|^2$$

is negative if $(c_{i-1} + c_i/2) < u_0$ as long as $\mathbf{c} = \{c_i\}_{i=1}^n$ forms an increasing sequence. Similarly, ϕ_t is negative if $(c_{i-1} + c_i/2) > u_0$ as long as $\mathbf{c}$ forms a decreasing sequence.

Proof: For each i and (x,y) such that $\phi(x,y) = i$

$$\begin{aligned}\phi_t(x,y) &= -|c_{i-1} - u_0|^2 + |c_i - u_0|^2 \\ &= -\left(c_{i-1}^2 - 2c_{i-1}u_0 + u_0^2\right) + \left(c_i^2 - 2c_i u_0 + u_0^2\right) \\ &= (c_{i-1} - c_i)\left(2u_0 - (c_{i-1} + c_i)\right).\end{aligned}$$

If $\mathbf{c} = \{c_i\}_{j=i}^n$ forms an increasing sequence, $c_{i-1} < c_i$ for $i = 2, \ldots, n$. Then, ϕ_t is negative; hence, ϕ is decreasing if $(c_{i-1} + c_i/2) < u_0$. Similarly, if $\mathbf{c}$ forms a decreasing sequence, $c_{i-1} > c_i$ for $i = 2, \ldots, n$ and then ϕ_t is negative; hence, ϕ is decreasing if $(c_{i-1} + c_i/2) > u_0$. ■

From Lemma 1, we know that it is enough to just solve a simple inequality instead of computing energy E_i if $\mathbf{c}$ has an increasing or decreasing form. If we could keep $\mathbf{c}$ increasing, we could update the value of ϕ^{new} to $\phi^{\text{old}} - 1$ only when $(c_{i-1} + c_i/2) < u_0$. We can easily assure that $\mathbf{c}$ maintains its increasing property while time evolves by updating ϕ as in the following Lemma 2.

Lemma 2: In region $\Omega_i = \{(x,y) : \phi(x,y) = i$ for $2 \le i \le n$, if we update

$$\phi^{\text{new}}(x,y) = i - 1, \qquad \text{when } \frac{c_{i-1} + c_i}{2} < u_0$$

then the increasing property of $\mathbf{c} = \{c_i\}_{i=1}^n$ is preserved.

In particular, in the case of $n = 2$, the ODE (22) is simply

$$\phi_t = -|c_1 - u_0|^2 + |c_2 - u_0|^2.$$

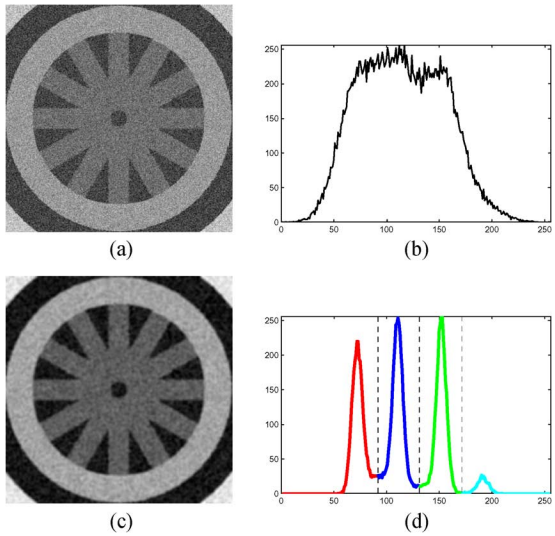

Fig. 9. (a) Original noisy image. (b) Its image histogram. (c) Smooth version of the noisy image. (d) Four clusters obtained from the AGMC method with image (c). We can easily see the clusters that contain indexes of significant local maxima.

By Lemma 2, $\phi^{\text{new}}(x,y) = 1$ if $(c_1 + c_2/2) < u_0(x,y)$ for $\Omega_2 = \{(x,y) : \phi(x,y) = 2\}$. This gives the same result as the Gibou–Fedkiw algorithm.

The segmentation algorithm is as follows:

Require: Define initial ϕ by (14).

repeat

(i) Update ϕ from Lemma 2.

(ii) Compute $\mathbf{c} = \{c_i\}_{i=1}^n$:

$$c_i = \text{average}(u_0) \quad \text{in } \Omega_i.$$

until $|\phi^{\text{new}} - \phi^{\text{old}}| < \varepsilon$

IV. Numerical Results

We show the efficiency of our method that gives reasonable results without the prefixed number of distinct regions in an image by comparison with other methods. It can automatically find the number of different regions and the meaningful clusters that have an important role in defining initial ϕ of our segmentation procedure. Another advantage of our method is that the reasonable number of regions can be obtained from noisy images and clean images. For a noisy image, we smooth the image and then use the histogram of the smooth image rather than the histogram of the noisy one. Then, we can notice more clearly where the site of significant local maxima of the image histogram is, as shown in Fig. 9. This helps to easily find adaptive global maximum clusters. Although the intensities of every pixel were slightly changed by smoothing the image, it does not extremely affect the segmentation result. The exact locations of the adaptive global maxima are not crucial because we ultimately need not the exact indexes of maxima but clusters, which are subintervals of [1, 256], containing the indexes of maxima. In all our experiments, the range of the histogram was scaled from 1 to 256 for convenience. Although we know only the approximate locations of clusters, we could acquire enough good results because the desired locations will be found through the segmentation procedure.

Moreover, we can utilize the histogram of the smooth image for the clean image as well. Then, it makes one remove unnecessary parts to search for maxima, for example, the red points in Fig. 10.

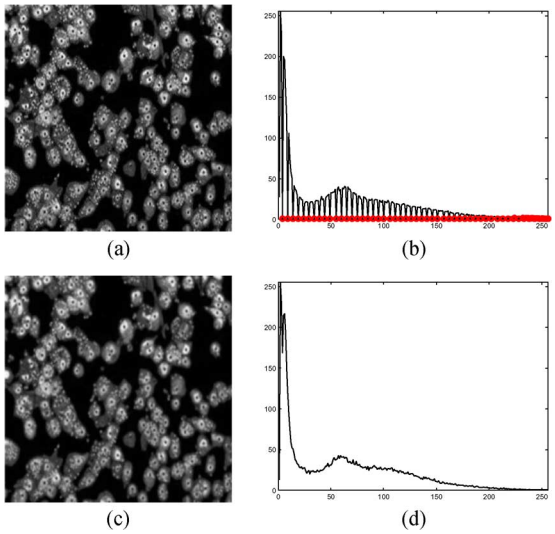

Fig. 10. (a) Original clean image. (b) Its image histogram. (c) Smooth version of the clean image. (d) Its image histogram. We can remove points such as the red circles in (b), which are undesired in finding maxima.

Therefore, for both clean and noisy images, we can use the histogram of the smooth images in the AGMC procedure to gain the number of distinct regions. The original noisy or clean images are utilized in the segmentation procedure with the number and the clusters obtained from the AGMC procedure.

Now, we compare our method with the multilevel thresholding method and the phase balancing model. In the multilevel thresholding method [18], the cost function of Kittler and Illingworth was particularly selected to find the best number of classes of an image histogram. The assumption is that a histogram is normally distributed with distinct means and variances. The minimization problem is described as

$$\underset{1\leqslant l_1<\cdots<l_{m-1}<L}{\arg\min}\sum_{m=1}^{M} f(l_{m-1},l_m],$$
$$f(a,b]=\omega(a,b]\log\left(\frac{\sigma(a,b]}{\omega(a,b]}\right).$$

Since the objective function signifies the overlap between the Gaussian models, the optimal solution is the set of thresholds that minimizes the overlap.

We draw a comparison between our method and this multilevel thresholding method over the following three images: Figs. 11(a), 12(a), and 13(a). In Fig. 11, the image histogram is normally distributed, and for this case, we can clearly find the correct number of classes, i.e., three classes, in Fig. 11(e). If we apply the method [17] with the cost function of Kittler and Illingworth and the number of classes, $M = 3$, we have three classes, as shown in Fig. 11(f). Fig. 11(b) shows the result segmented by the thresholds in Fig. 11(f), and Fig. 11(c) is the segmentation result of our method. Both segmentation results are almost the same and the most effective ones.

On the other hand, in Fig. 12, the distribution Fig. 12(b) of the image histogram is not normal. It is composed of four peaks; one is prominently large and the other ones are relatively very small. In this experiment, we failed to obtain the correct number of classes, $M = 4$, and could not get the desired segmentation result although we set $M = 4$. We investigate the values of (9) and (10) at each stage $s = 1, \ldots, M$ to explain this phenomenon.

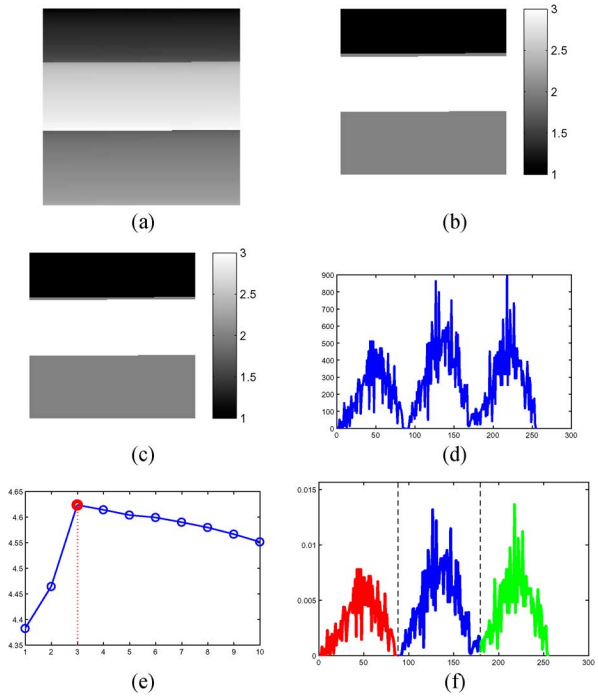

Fig. 11. (a) Original image. (b) Segmentation result from the multilevel thresholding method. (c) Segmentation result from our method. (d) Image histogram. (e) $CFF(s)$ for $1 \leq s \leq 10$ in the multilevel thresholding method. (f) Thresholds and classes of the histogram in the multilevel thresholding method. Our result (c) is almost the same as the correct result (b) from the multilevel thresholding method.

At the first stage, $s = 1$, we compute $E_{\text{val}}(1,t) = f(0,t]$ for all $1 \leqslant t \leqslant 253$. Then, at the second stage, $s = 2$, we compute $E_{\text{val}}(2,t) = \min_{1\leqslant l<t}\{E_{\text{val}}(1,l) + f(l,t]\}$ and $E_{\text{pos}}(2,t) = \arg\min_{1\leqslant l<t}\{E_{\text{val}}(1,l) + f(l,t]\}$ for $2 \leqslant t \leqslant 254$. However, since $-2 \leqslant |\, f(\cdot,\cdot]\,| \leqslant 4.3$ and $E_{\text{val}}(1,1)$ is around -26, $E_{\text{val}}(2,t)$ and $E_{\text{pos}}(2,t)$ are dominantly determined by $E_{\text{val}}(1,1)$. This implies that the method cannot identify any other peaks of the histogram at the second stage and cannot detect another peak for some stages. In Fig. 12, the other peaks were recognized after three stages. It directly caused the wrong number of classes and failed to decompose the histogram into proper classes. As shown in Fig. 12(c), the variations of E_{val} at the first and second stages are too big for the values of f to affect the evaluations in (9) and (10). If the given image histogram has an extravagantly large peak, one of the values E_{val} of the objective function of Kittler and Illingworth is extremely small compared with the other values. Then, the variations of E_{val} are quite big during some stages and it obstructs the exploration of proper classes as well as the correct number of classes. The multilevel thresholding method failed to separate the second and third peaks of the histogram in Fig. 12(d), but our method could find the correct number of regions through the AGMC procedure and could segment the image properly in Fig. 12(g).

We emphasize the need of our segmentation procedure in Fig. 13. The histogram in Fig. 13(b) does not have a clear threshold, and the segmentation result is very sensitive to the threshold. Fig. 13(c) and (f) are the results from the multilevel thresholding method, and Fig. 13(d), (e), (g), and (h) are obtained from our method. Fig. 13(d) shows two clusters acquired from the AGMC procedure, and each thick solid line represents the distribution of the histogram over each cluster. We note that

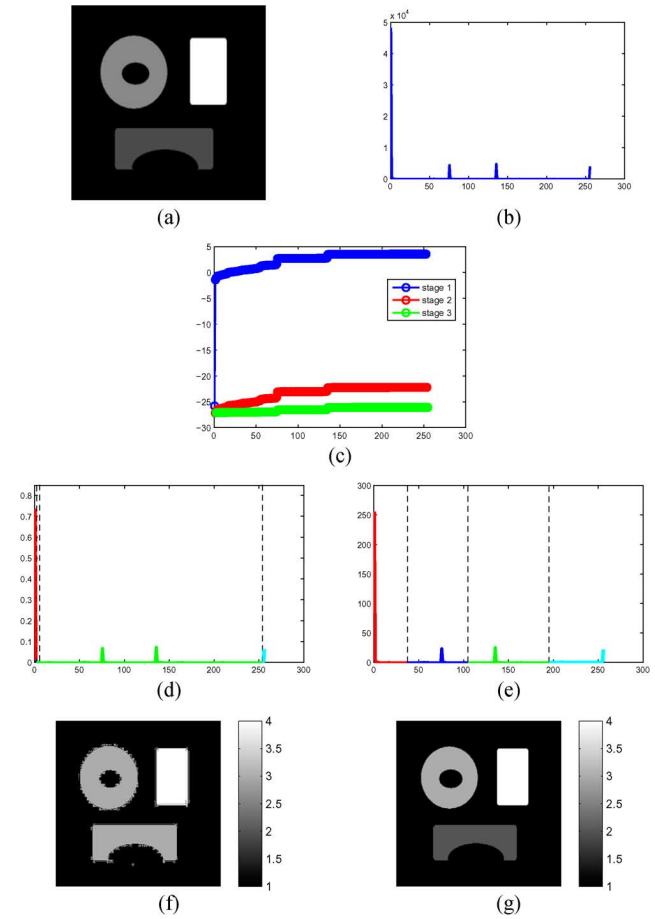

Fig. 12. (a) Original image. (b) Image histogram. (c) Values of $E_{\mathrm{val}}(\cdot,\cdot)$ at three stages for the multilevel thresholding method. (d) Thresholds and classes of the histogram in the multilevel thresholding method. (e) Thresholds corresponding to our segmentation result (g). (f) Segmentation result from the multilevel thresholding method. (g) Segmentation result from our method. According to the instructions of methods, we used the normalized histogram in (d) and the scaled histogram from 1 to 256 in (e). While our method gives a good result as shown in (g), the multilevel thresholding method cannot distinguish between the circle object and bottom object in (f).

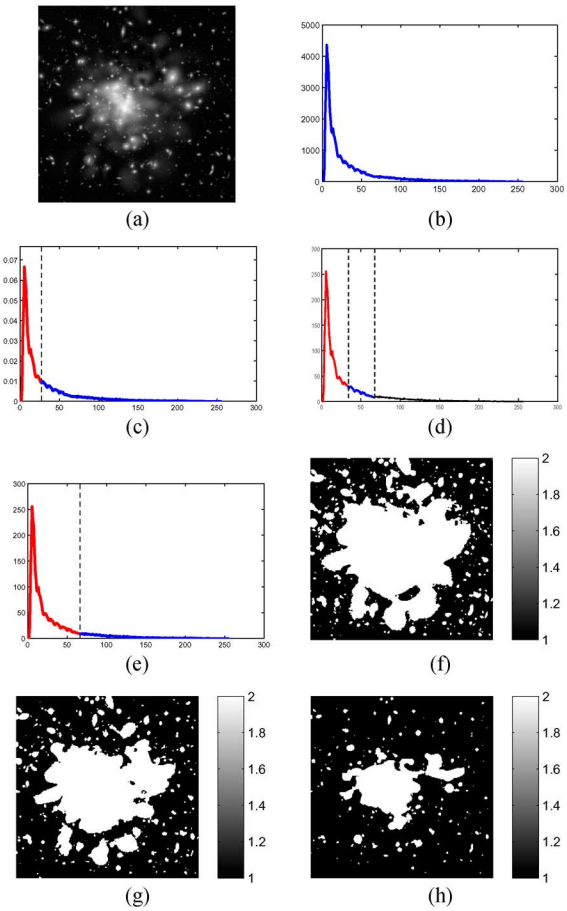

Fig. 13. (a) Original image. (b) Image histogram. (c) Thresholds and classes of the hqistogram in the multilevel thresholding method. (d) Two clusters, [1, 33] and [33, 68], from the AGMC method. (e) Changed clusters after our segmentation procedure. (f) Segmentation results corresponding to (c). (g) Initial level set ϕ from (d). (h) Final segmentation results for our model. The segmentation procedure by using the level set is helpful to get the reasonable result.

a cluster is not a set of histogram values but its domain. By using clusters in Fig. 13(d), we define an initial level set ϕ for the segmentation procedure by (14), as shown in Fig. 13(g). The exact border between clusters of the histogram is not required in the AGMC process because the border can be adjusted through the segmentation procedure. Fig. 13(h) shows the final segmentation result, and Fig. 13(e) describes the shifted border between the clusters after segmentation. By comparison with the original image Fig. 13(a), we see that Fig. 13(h) gives more reasonable segmentation result than Fig. 13(f).

Now, we compare our model with the phase model in Figs. 14 and 15. In Fig. 14, our method classified the original image in Fig. 14(a) into five different regions in Fig. 14(b) with $\sigma = 20$ and $\omega = 0.5$. In addition, the phase balancing model also gave a similar result with $\mu = 1.1$. In the phase balancing model [16], the authors mentioned that, when (x, y) is in the pth phase, a new phase is created only if there is a significantly large difference between the intensity $u_0(x, y)$ of the image and the average of intensities over the pth phase. The original image in Fig. 15(a) consists of a dark gray region (a rectangular object) whose intensities are similar to another black region (background) and

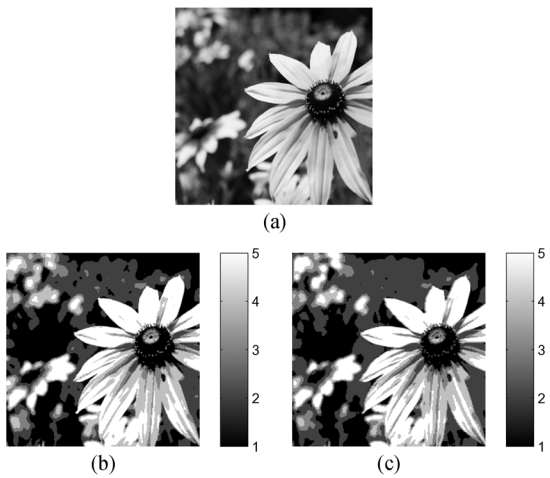

Fig. 14. (a) Original image. (b) Our method with $\sigma = 20$ and $\omega = 0.5$. (c) Phase balancing model with $\mu = 1.1$. Two reasonable results are similar.

different from the other white region (a circular object). The phase balancing model failed to distinguish between rectangular dark gray objects and the black background, as shown in Fig. 15(c), because the difference is too small to create a new phase. On the other hand, we obtained the desired segmentation

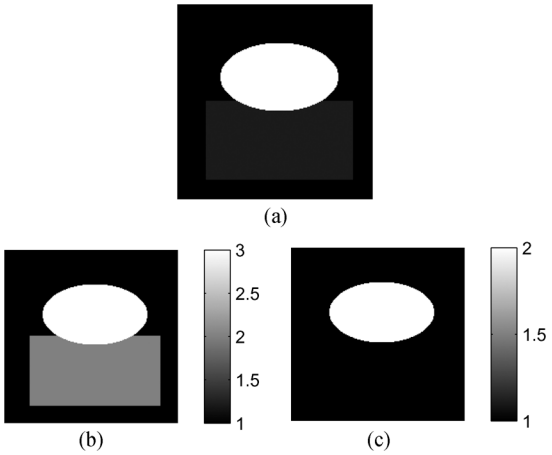

Fig. 15. (a) Original image. (b) Our method with $\sigma = 15$ and $\omega = 2$. (c) Phase balancing model with $\mu = 1$. Our method splits the image into the correct three regions, but the phase balancing model failed to distinguish between the black and the dark gray regions.

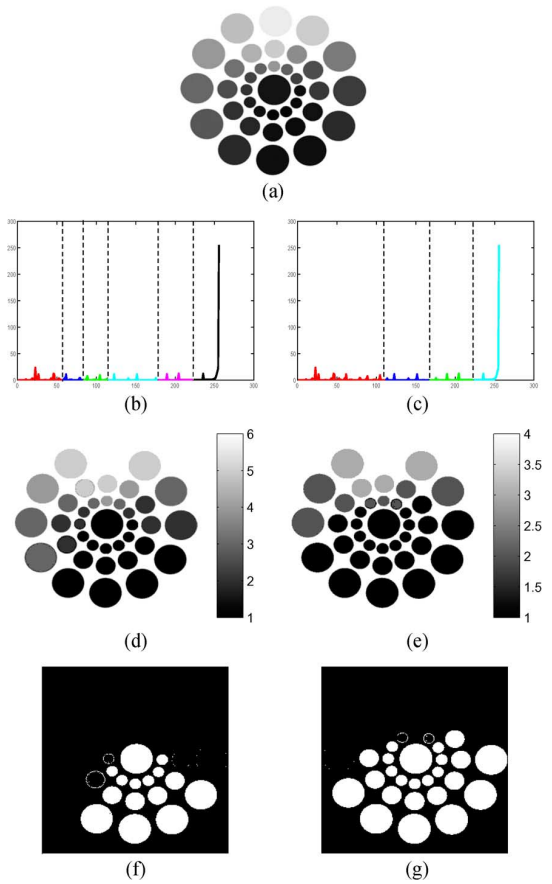

Fig. 16. Role of σ. (a) Original image. (b) Six clusters from the AGMC process with $\sigma = 30$ and $\omega = 3$. (c) Four clusters from the AGMC process with $\sigma = 40$ and $\omega = 3$. (d) Segmentation result of (b). (e) Segmentation result of (c). (f) First region in (d). (g) First region in (e). Both the first regions (f) and (g) in two experiments are different.

result Fig. 15(b) by adjusting σ to be quite small. σ is a parameter that controls the least difference in the gray levels of distinct regions and thus has an effect on the number of regions.

Fig. 16 explains the role of σ. In our experiments, we normally used σ between 15 to 40 at five intervals, for example, 15, 20, . . ., 40. Fig. 16(b), (d), and (f) are the results when we used $\sigma = 30$ and $\omega = 3$, and Fig. 16(c), (e), and (g) are the results when we used $\sigma = 40$ and $\omega = 3$. Fig. 16(f) and (g) shows

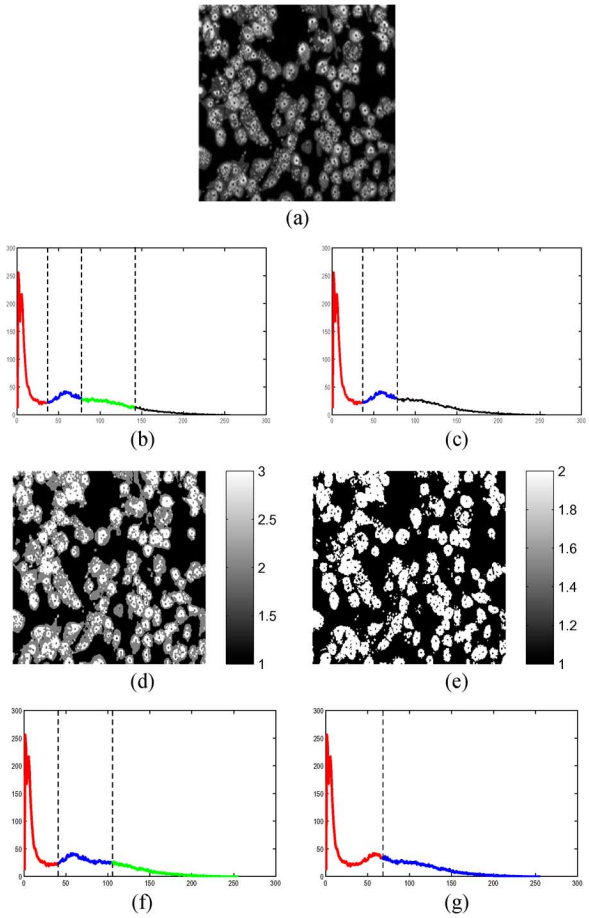

Fig. 17. Role of ω. (a) Original image. (b) Three clusters (thick lines) from the AGMC process with $\sigma = 30$ and $\omega = 1$. (c) Two clusters (thick lines) from the AGMC process with $\sigma = 30$ and $\omega = 1.5$. (d) Segmentation result of (b). (e) Segmentation result of (c). (f) Changed clusters after our segmentation procedure for case (b). (g) Changed clusters after our segmentation procedure for case (c). If we use a smaller σ, we get a larger number of regions.

the first regions of two cases. By using the smaller σ, we have the larger number of distinct regions. Fig. 16(b) and (c) signifies clusters of adaptive global maxima obtained from the AGMC procedure, and Fig. 16(d) and (e) are the segmented results with the numbers of distinct regions and clusters in Fig. 16(b) and (c).

A histogram contains a large number of local maxima; because of the definition of local maximum, meaning we might detect even small local maxima that are usually not meaningful. By controlling ω, we can exclude a possibility of the detection of such small maxima. However, there are some cases in which such local maxima are significant; thus, we need to select those maxima. We normally chose ω from 0.5 to 3 at 0.5 intervals. For the sake of our needs, we can draw the desired results by adjusting the value of ω, as shown in Fig. 17. By using the smaller ω, we can find the green cluster in Fig. 17(b) so that we can detect gray regions in Fig. 17(d).

Last, Fig. 18 shows the efficiency of the AGMC procedure for a noisy image, i.e., the same image in Fig. 9, and we acquired the correct number of distinct regions from the AGMC procedure for noisy image, as shown in Fig. 18(d). Simply to show more obvious regions, we applied the two-region segmentation model [5] to each noisy region in Fig. 18(c), and the results are displayed over in –(h).

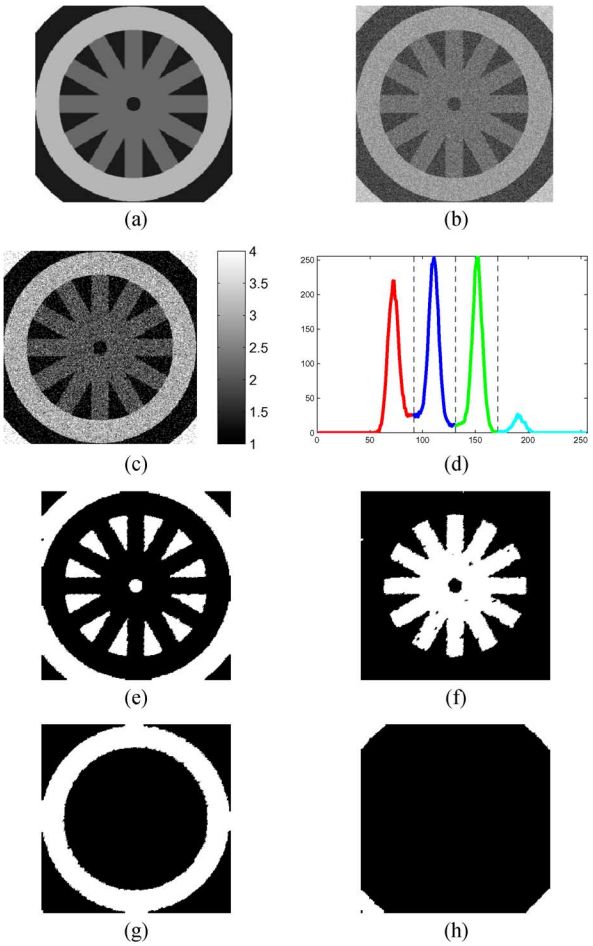

Fig. 18. SNR $= 6$. (a) Original clean image. (b) Noisy image. (c) Segmentation result. (d) Four clusters from the AGMC process with $\sigma = 30$ and $\omega = 0.5$. (e)–(h) Each region.

V. Conclusion

We have developed a new unsupervised multiple-region segmentation method by using the AGMC procedure. Our method is composed of two procedures, namely, the segmentation and the AGMC. In the AGMC procedure, we automatically obtain the number of distinct regions and clusters, which are subintervals of gray levels containing adaptive global maxima. We fix $k = 2$ and repeat the k-means clustering under a few rules in this procedure. In the segmentation procedure, we decompose the original image into the multiple regions by modifying the Mumford–Shah model according to the number and clusters obtained from the AGMC procedure. In the numerical experiments, we have presented the efficiency of our method through comparison with other methods.

References

[1] D. Mumford and J. Shah, "Optimal approximation by piecewise smooth functions and associated variational problems," *Commun. Pure Appl. Math.*, vol. 42, no. 5, pp. 577–685, Jul. 1989.
[2] T. Chan and L. A. Vese, "Active contours without edges," *IEEE Trans. Image Process.*, vol. 10, no. 2, pp. 266–277, Feb. 2001.
[3] S. Osher and J. A. Sethian, "Fronts propagating with curvature dependent speed : Algorithms based on Hamilton–Jacobi formulations," *J. Comput. Phys.*, vol. 79, no. 1, pp. 12–49, Nov. 1988.
[4] F. Gibou and R. Fedkiw, "Fast hybrid k-means level set algorithm for segmentation," in *Proc. 4th Annu. Hawaii Int. Conf. Stat. Math.*, 2002, pp. 281–291.
[5] B. Song and T. Chan, "A fast algorithm for level set based optimization," UCLA, Los Angeles, CA, UCLA CAM Rep. 02-68, 2002.
[6] T. F. Chan and L. A. Vese, "Image segmentation using level sets and the piecewise constant Mumford–Shah model," Dept. Math, UCLA, Los Angeles, CA, Tech. Rep. CAM 00-14, 2000.
[7] L. A. Vese and T. 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.
[8] J. Lie, M. Lysaker, and X.-C. Tai, "A variant of the level set method and applications to image segmentation," *Math. Comp.*, vol. 75, no. 255, pp. 1155–1174, Jul. 2006.
[9] X.-C. Tai and C. Yao, "Fast implementation of fast PCLSM with Newton updating algorithm," in *Image Processing Based on Partial Differential Equations*. Berlin, Germany: Springer-Verlag, 2006, pp. 249–264, ISBN 978-3-540-33266-4.
[10] X.-C. Tai and C.-H. Yao, "Image segmentation by piecewise constant Mumford–Shah model without estimating the constants," *J. Comput. Math.*, vol. 24, no. 3, pp. 435–443, 2006.
[11] E. Brown, T. F. Chan, and X. Bresson, "Convex formulation and exact global solutions for multi-phase piecewise constant Mumford–Shah image segmentation," UCLA, Los Angeles, CA, UCLA CAM Rep. 09–66, Aug. 2009.
[12] X.-C. Tai and T. F. Chan, "A survey on multiple level set methods with applications for identifying piecewise constant functions," *Int. J. Numer. Anal. Model.*, vol. 1, no. 1, pp. 25–47, 2004.
[13] J. Lie, M. Lysaker, and X.-C. Tai, "Piecewise constant level set methods and image segmentation," in *Scale Space and PDE Methods in Computer Vision : 5th International Conference, Scale-Space 2005*, R. Kimmel, N. Sochen, and J. Weickert, Eds. Heidelberg,, Germany: Springer–Verlag, 2005, vol. 3459, pp. 573–583.
[14] S. Osher and R. Fedkiw, "An overview and some recent results," *J. Comput. Phys.*, vol. 169, no. 2, pp. 463–502, May 2001.
[15] J. Lie, M. Lysaker, and X.-C. Tai, "A binary level set model and some applications for Mumford–Shah image segmentation," *IEEE Trans. Image Process.*, vol. 15, no. 5, pp. 1171–1181, May 2006.
[16] B. Sandberg, S. H. Kang, and T. Chan, "Unsupervised multiphase segmentation: A phase balancing model," *IEEE Trans. Image Process.*, vol. 19, no. 1, pp. 119–130, Jan. 2010.
[17] M. Luessi, M. Eichmann, G. Schuster, and A. Katsaggelos, "Framework for efficient optimal multilevel image thresholding," *J. Electron. Imag.*, vol. 18, p. 013004, Feb. 2009.
[18] M. Eichmann and M. Lussi, "Efficient multi level image thresholding," M.S. thesis, Science in Engineering, Hochschule fur Technik Rapperswil, Rapperswil, Switzerland, Dec. 2005.
[19] J. Kittler and J. Illingworth, "Minimum error thresholding," *Pattern Recognit.*, vol. 19, no. 1, pp. 41–47, 1986.
[20] T. Kurita, N. Otsu, and N. N. Abdelmalek, "Maximum likelihood thresholding based on population mixture models," *Pattern Recognit.*, vol. 25, no. 10, pp. 1231–1240, Oct. 1992.

Sunhee Kim received the B.S. degree in mathematics from Konkuk University, Seoul, Korea, in 2004, and the M.S. degree in mathematics from Seoul National University, Seoul, Korea, in 2006, where she is currently working toward the Ph.D. degree in mathematics.

Her research interests are in mathematical image processing such as segmentation, denoising, and applications in medical fields.

Myungjoo Kang received the B.S. degree in mathematics from Seoul National University, Seoul, Korea, in 1986, and the Ph.D. degree in mathematics from the University of California at Los Angeles, Los Angeles, in 1996.

He was with the Department of Electrical and Computer Engineering, University of California, San Diego, La Jolla, as a Postdoctoral Researcher from 1996 to 2000. From 2003 to 2007, he was an Assistant Professor with the Department of Mathematical Sciences, Seoul National University, where he has been an Associate Professor since 2008. His research interests are in mathematical image processing, as well as numerical schemes and computational fluid dynamics.