A Fast Segmentation Method Based on Constraint Optimization and Its Applications: Intensity Inhomogeneity and Texture Segmentation

Jun Liu[a], Xue-cheng Tai[b,c], Haiyang Huang[a], Zhongdan Huan[a]

[a] *School of Mathematical Sciences, Laboratory of Mathematics and Complex Systems, Beijing Normal University, Beijing 100875, P.R. China.*
[b] *Division of Mathematical Sciences, School of Physical and Mathematical Sciences, Nanyang Technological University, 637616, Singapore.*
[c] *Department of Mathematics, University of Bergen, Johannes Brunsgate 12, N-5008 Bergen, Norway.*

Abstract

We propose a new constraint optimization energy and an iteration scheme for image segmentation which is connected to edge-weighted centroidal Voronoi tessellation (EWCVT). We show that the characteristic functions of the edge-weighted Voronoi regions are the minimizers (may not unique) of the proposed energy at each iteration. We propose a narrow banding algorithm to accelerate the implementation, which makes the proposed method very fast. We generalize the CVT segmentation to hand intensity inhomogeneous and texture segmentation by incorporating the global and local image information into the energy functional. Compared with other approaches such as level set method, the experimental results in this paper have shown that our approach greatly improves the calculation efficiency without losing segmentation accuracy.

Key words: Image Segmentation, Constraint Optimization, Centroidal Voronoi Tessellation, Texture Segmentation, Intensity Inhomogeneous, Fast Algorithm

1. Introduction

Image segmentation plays a very important role in many practical applications, such as computer vision, artificial intelligence, medical images analysis and so on. The aim of image segmentation is to find a partition of an image into its constituent parts.

Many methods have been proposed for image segmentation. They can be classified in different groups including 1) region based and edge based segmentation (e.g.[1, 20, 17]),

2) soft-threshold and hard-threshold segmentation (e.g.[2, 3, 4, 17, 20]), 3) parametric and nonparametric segmentation (e.g. [17, 20, 5]), 4) supervised and unsupervised segmentation (e.g. [6, 7]), etc..

Level set method is one of the most successful segmentation method due to its flexibility and a well-grounded theory. The well-known Mumford-Shah model for image segmentation [19] had been successfully extended to a wide range of applications. One of the simplified variant of Mumford-Shah model is Chan-Vese model [20], which is well suited for segmenting an image with nearly constant intensity. However, the level set method (e.g. Chan-Vese model) needs to update the level set function by solving some nonlinear partial differential equations (PDE). In order to improve its efficiency and also overcome some of the difficulties with non-convexity and non-differentiability, some piecewise constant level set method (PCLSM) was proposed in [21, 22, 23, 24], to deal with multi-class classification by representing any number of phases with one single label function. In addition, numerical scheme based on graph cut method can be used to get fast implementations (see [9, 10]). In the graph cut approach of [9], to deal with n clusters, it needs to add extra $n-1$ layers vertices to the graph. For large images such as 3D CT images, this will consume large amounts of memory and the implementation is also time-consuming. Domain decomposition methods have been used to reduce the memory cost and also computing time [11, 12]. Recently, primal-dual approaches have been used to find global minimizers for for some related models in [44, 13, 14]. Compared with [21, 22, 23], we are using the characteristic functions to identify the sub-regions. The idea of using characteristic functions has aslo been used in [44, 13, 14, 25, 47, 26, 27] for multi-labelling problems and some other earlier work related to phase field.

Centroidal Voronoi Tessellation (CVT) for image segmentation was proposed in [17]. In its simplest form, the basic CVT algorithm is known as the k-means clustering, which is widely used in many applications. The standard CVT clustering is sensitive to noise and may fail to provide the accurate segmentation results when the image is contaminated by noise. To address this issue, a natural alternative is to add a length term (regularization term) in the energy, such as the length term of Mumford-Shah model [19], Chan-Vese model [20], piecewise constant level set method [21, 22, 23]. However, the regularization term makes this minimization problem difficult to solve. In [18], the au-

thors extend CVT clustering to Edge-weighted Centroidal Voronoi tessellation (EWCVT) clustering by incorporating the image intensity information together with the length of cluster boundaries. In EWCVT, pixels with its neighbors belong to different clusters are penalized and the approximate length regularization term is the same as that in [4]. Relatively speaking, this smoothness term can be more efficiently implement than Total-variation regularization. Compared with the level set method, the CVT based segmentation has certain advantages, such as the exact minimizer of the energy can be found without solving a partial differential equation (PDE). However, the EWCVT lacks of image edge information and may fail to provide the exact segmentation in some cases. Moreover, it can not handle the intensity inhomogeneous problem since it lacks of the local information in the data term.

Intensity inhomogeneous is common in natural images. Existing methods to tackle this problem include the piecewise smooth Mumford-Shah model [35] and local binary fitting (LBF) method [36, 37]. In these models, the active contour is driven by local information of the image. Another alternative is to add the local information to the segmentation functional by estimating the brightness of the background, such as [3, 38]. There are many unknown variables in these methods and it is a very time-consuming to solve them.

In this paper, we propose a method for image segmentation by minimizing the segmentation energy with some constraint conditions. This constraint optimization problem has an explicit solution which is connected with the Voronoi tessellation. To be precise, the characteristic function of the Voronoi region is the exact minimizer of the proposed energy. Even more, we can use narrow-band type of technique to speed up the computing. We aslo extend the CVT based image segmentation to treat images with intensity inhomogeneities by combining it with local histogram equalization. In addition, as an application of the proposed method, a texture segmentation model is also proposed. We theoretically show that the proposed algorithm is energy descent, and thus it is at least locally converged. Experimental results and comparisons with other methods have shown the high efficiency of the proposed approach. Especially, we want to emphasis that our proposed algorithm has the following advantages:

- Due to special linearization, there is no need to solve any partial differential equa-

tion in our algorithm. The iterative schemes is fast and stable.

- The proposed model can treat the length regularization term properly and accurately. It is known that "curve length" and "surface area" is crucial for a number of interface problem beside image segmentation. The algorithm proposed here offers a fast approach for the computation of these problems.

The rest of the paper is organized as follows: in section 2, the CVT and EWCVT for image segmentation is briefly introduced; The proposed energy, algorithm, some analytical results for the algorithm and model, and its applications are presented in section 3; section 4 contains the experimental results and some comparisons with other approaches; finally, some conclusion remarks are drawn in section 5.

2. Previous Work

2.1. CVT

Throughout this paper, we denote the label set $\mathbb{L} = \{1, 2, \cdots, M\}$. Suppose $f : \Omega \subseteq \mathbb{R}^2 \mapsto \mathbb{R}$ denotes a gray-valued image which is defined on a bound domain Ω. Then the image segmentation corresponds to divide the image domain Ω into several non-overlapping subregions Ω_k, such that the image data in each Ω_k share some similarities, such as the similar intensity values, textures, or structures. First, let us recall some concepts of centroidal Voronoi tessellation [15, 16, 17] for image segmentation.

Given a set of distinct image intensity values $\{c_k\}_{k=1}^M \in \mathbb{R}$, then the Voronoi region Ω_k in Ω corresponding to each c_k, is defined by

$$\Omega_k = \{x \in \Omega : d(f(x), c_k) \leq d(f(x), c_i), \forall i \in \mathbb{L}\}, \tag{1}$$

where $k = 1, 2, \cdots, M$, d is a distance measure, and $\{c_k\}_{k=1}^M$ are referred as Voronoi generators. On the other hand, for any given Voronoi tessellation denoted by $\{\Omega_k\}_{k=1}^M$, one can usually define the centroid $\tilde{c}_k$ of each region Ω_k by solving the following minimization problem

$$\tilde{c}_k = \arg\min_{c_k} \int_{\Omega_k} \rho(x) d^2(f(x), c_k)\, \mathrm{d}x, \tag{2}$$

where $\rho : \Omega \mapsto \mathbb{R}^+$ is a given density function.

Generally speaking, the Voronoi generators $\{c_k\}_{k=1}^M$ which generate the Voronoi tessellation $\{\Omega_k\}_{k=1}^M$ are not the centroids $\{\tilde{c}_k\}_{k=1}^M$. Centroidal Voronoi tessellation (CVT) [15] is a special Voronoi tessellation whose generators satisfy

$$c_k = \tilde{c}_k, \forall k \in \mathbb{L},$$

which means the generators of the Voronoi regions $\{\Omega_k\}_{k=1}^K$ coincide with the corresponding centroids $\{\tilde{c}_k\}_{k=1}^M$.

In [15, 17], the authors define an energy for Voronoi tessellation, and they show that the CVT can be constructed by finding a minimizer of the following energy (Prop. 3.1, [15])

$$E_1(\{(\Omega_k, c_k)\}_{k=1}^M) = \sum_{k=1}^{M} \int_{\Omega_k} \rho(x) d^2(f(x), c_k)\, \mathrm{d}x.$$

In image segmentation, ρ is often chosen as a probability density function of the uniform distribution, i.e. $\rho = \frac{1}{|\Omega|}$, where $|\cdot|$ represents the area. Then in [17], an iterative algorithm (Lloyds Method) is employed to solve the minimization problem for image segmentation:

Algorithm 1 (CVT clustering, [17]). *Given the number of classes M, choosing an initial set $\{c_k\}_{k=1}^M$,*

step 1. Find the Voronoi tessellation $\{\Omega_k\}_{k=1}^M$ by (1).

step 2. Determine the centroids $\{\tilde{c}_k\}_{k=1}^M$ through (2).

step 3. $\forall k \in \mathbb{L}$, if $\tilde{c}_k = c_k$, end the algorithm; else let $c_k = \tilde{c}_k$ and go to the step 1.

As mentioned earlier, the CVT clustering is sensitive to noise. EWCVT [18] is a robust CVT and it can provide good segmentation result under noise. We will give a brief introduction on EWCVT in the next subsection.

2.2. EWCVT

In [18], the authors defined a discrete energy for image segmentation which is robust to noise. We can rewrite this clustering energy in the continuous form as follows:

$$\begin{aligned} E_2(\{(\Omega_k, c_k)\}_{k=1}^M) = &\quad E_1(\{(\Omega_k, c_k)\}_{k=1}^M) + \lambda E_L \\ = &\quad \sum_{k=1}^{M} \int_{\Omega_k} d^2(f(x), c_k)\, \mathrm{d}x + \lambda \int_{\Omega} \int_{B(x;\omega)} \chi_k(x, y)\, \mathrm{d}y\, \mathrm{d}x. \end{aligned} \tag{3}$$

Here $\lambda > 0$ is a regularization parameter that controls the tradeoff between the CVT clustering energy and the length of the cluster boundaries. $B(x;\omega)$ stands for a neighborhood centered at x with parameter ω, e.g. $B(x;\omega)$ can be a disk centered at x with radius ω or a $\omega \times \omega$ square centered at x. The characteristic function $\chi_k : \Omega \times \Omega \mapsto \{0, 1\}$ for Ω_k is defined as:

$$\chi_k(x, y) = \begin{cases} 0, & x \in \Omega_k \ \text{ and } \ y \in \Omega_k, \\ 1, & \text{else.} \end{cases}$$

It was shown in [18] that

$$E_L \propto \text{length}(\Gamma)$$

in the case of two clusters. Here Γ is the boundary curve of a partition $\{\Omega_k\}_{k=1}^2$.

By considering the variation of the energy E_2 when one transfers a pixel located at x from its current Voronoi region Ω_{k_1} to another Voronoi region Ω_{k_2}, the authors proposed an algorithm based on edge-weighted Voronoi region to find the minimizer of the above energy. Firstly, they defined an edge-weighted distance $\tilde{d}$ from a pixel located at x to a generator c_k as

$$\tilde{d}(f(x), c_k) = \sqrt{d^2(f(x), c_k) + 2\lambda \tilde{n}_k(x)}, \tag{4}$$

where $\tilde{n}_k(x) = |B(x;\omega)| - |B(x;\omega) \cap \Omega_k|$.

Then similar to [15, 17], an edge-weighted Voronoi region can be given as:

$$\Omega_k = \{x \in \Omega : \tilde{d}(f(x), c_k) \leqslant \tilde{d}(f(x), c_i), \forall i \in \mathbb{L}.\}. \tag{5}$$

The EWCVT algorithm can be described as follows. For more details, please refer to [18].

Algorithm 2 (EWCVT clustering, [18]). *Given the number of classes M, choosing an initial set $\{c_k\}_{k=1}^M$,*

step 1. Find the edge-weighted Voronoi tessellation $\{\Omega_k\}_{k=1}^M$ by (5).

step 2. Determine the centroids $\{\tilde{c}_k\}_{k=1}^M$ through (2).

step 3. $\forall k \in \mathbb{L}$, if $\tilde{c}_k = c_k$, end the algorithm; else let $c_k = \tilde{c}_k$ and go to the step 1.

The only difference between the CVT clustering algorithm and the EWCVT clustering algorithm is the different distance functions d and $\tilde{d}$ in the Voronoi tessellation. EWCVT

can provide a better result than the standard CVT for noisy images. However, like many other models, the EWCVT does not consider image edge information. Edges in images usually can provide useful information for segmentation.

3. The Proposed Method

3.1. A constrained optimization for image segmentation

In this section, we present a constrained energy minimization problem for CVT-based image segmentation. Let us begin with the following proposition:

Proposition 1. *Given two vector functions* $\mathbf{D}, \mathbf{u} : \Omega \subset \mathbb{R}^2 \mapsto \mathbb{R}^M$, $\mathbf{D}(x) = (D_1(x), D_2(x), \cdots, D_M(x))^{\mathrm{T}}$, $\mathbf{u}(x) = (u_1(x), u_2(x), \cdots, u_M(x))^{\mathrm{T}}$. *We assume that* $\mathbf{u}(x)$ *satisfies* $\sum_{k=1}^{M} u_k(x) = 1$, *and* $u_k(x) \geqslant 0$. *At each* x, *we denote* $\mathbb{K}_x = \{k \in \mathbb{L} : D_k(x) \leqslant D_i(x), \forall i \in \mathbb{L}\}$. *Let* $E(\mathbf{u}) \triangleq \int_\Omega \mathbf{D}(x) \cdot \mathbf{u}(x)\, \mathrm{d}x$, *then the following conclusions hold:*
a) $\hat{\mathbf{u}} = (\hat{u}_1, \hat{u}_2, \cdots, \hat{u}_M)$ *with component function*

$$\hat{u}_k(x) = \begin{cases} 0, & k \notin \mathbb{K}_x, \\ \hat{u}_k(x), & k \in \mathbb{K}_x, \quad s.t. \quad \sum_{k \in \mathbb{K}_x} \hat{u}_k(x) = 1, \end{cases}$$

where $k \in \mathbb{L}$, *is a minimizer of* $E(\mathbf{u})$.
b) Specially, if $\forall x, |\mathbb{K}_x| = 1$, *then we have*

$$\hat{u}_k(x) = \begin{cases} 0, & k \notin \mathbb{K}_x, \\ 1, & k \in \mathbb{K}_x, \end{cases}$$

where $k \in \mathbb{L}$, *is the unique minimizer of* $E(\mathbf{u})$. *Here,* $|\cdot|$ *is the cardinality of the set* $\mathbb{K}_x$.
c) The binary function $\hat{\mathbf{u}} = (\hat{u}_1, \hat{u}_2, \cdots, \hat{u}_M)$ *with component function*

$$\hat{u}_k(x) = \begin{cases} 1, & k = \min\{k_1 : k_1 \in \mathbb{K}_x\}, \\ 0, & else, \end{cases}$$

where $k \in \mathbb{L}$, *is a minimizer of* $E(\mathbf{u})$.

The proof of this proposition is given in Appendix A. The conclusion of this observation is also the key ingredient in finding global minimizers for the model proposed in [44].

Similar as in [44, 47], let us introduce a vector-valued characteristic function $\mathbf{u} = (u_1, u_2, \cdots, u_M)^{\mathrm{T}}$ and each $u_k : \Omega \mapsto \{0, 1\}$ is defined as

$$u_k(x) = \begin{cases} 0, & x \in \Omega_k, \\ 1, & x \notin \Omega_k, \end{cases}$$

then (3) can be rewritten as follows:

$$\begin{aligned} & E_2(\mathbf{u}, \mathbf{c}) \\ &= \sum_{k=1}^{M} \int_{\Omega} d^2(f(x), c_k) u_k(x)\, \mathrm{d}x + \lambda \sum_{k=1}^{M} \int_{\Omega} u_k(x) \int_{B(x;\omega)} \sum_{j=1, j\neq k}^{M} u_j(y)\, \mathrm{d}y\, \mathrm{d}x, \\ &= \int_{\Omega} \sum_{k=1}^{M} \left\{ \left[d^2(f(x), c_k) + \lambda \int_{B(x;\omega)} \sum_{j=1, j\neq k}^{M} u_j(y)\, \mathrm{d}y \right] u_k(x) \right\} \mathrm{d}x, \\ &\triangleq \int_{\Omega} \mathbf{D}(x) \cdot \mathbf{u}(x)\, \mathrm{d}x. \end{aligned} \tag{6}$$

In the last equation, the k-th component of the vector function $\mathbf{D} = (D_1, D_2, \cdots, D_M)^{\mathrm{T}}$ is given by:

$$D_k(x) = d^2(f(x), c_k) + \lambda \int_{B(x;\omega)} \sum_{j=1, j\neq k}^{M} u_j(y)\, \mathrm{d}y. \tag{7}$$

We shall consider the following constrained optimization problem

$$(\mathbf{u}^*, \mathbf{c}^*) = \arg\min_{\mathbf{u}, \mathbf{c}} \quad \left\{ E_2 \triangleq \int_{\Omega} \mathbf{D}(x) \cdot \mathbf{u}(x)\, \mathrm{d}x \right\} \quad \text{s.t.} \quad \sum_{k=1}^{M} u_k(x) = 1, u_k(x) \geqslant 0.$$

Note that the function $D(x)$ is different from the one in the proposition 1. It depends on function $\mathbf{u}$. Thus the conclusion of proposition 1 can not directly used to solve this minimization problem. Instead, we shall use an iterative scheme to solve this problem. Starting from an initial guess $\mathbf{u}^0, \mathbf{c}^0$, we compute a series of minimizers

$$\mathbf{u}^1, \mathbf{c}^1, \mathbf{u}^2, \mathbf{c}^2, \cdots, \mathbf{u}^{\nu+1}, \mathbf{c}^{\nu+1}, \cdots$$

such that

$$\mathbf{u}^{\nu+1} = \arg\min_{\mathbf{u}} \quad \tilde{E}_2(\mathbf{u}, \mathbf{c}^{\nu}; \mathbf{u}^{\nu}), \quad \text{s.t.} \sum_{k=1}^{M} u_k(x) = 1, u_k(x) \geqslant 0, \tag{8}$$

$$\mathbf{c}^{\nu+1} = \arg\min_{\mathbf{c}} \quad \tilde{E}_2(\mathbf{u}^{\nu+1}, \mathbf{c}; \mathbf{u}^{\nu+1}), \tag{9}$$

$$\nu = 0, 1, 2, \cdots \tag{10}$$

Here

$$\begin{aligned}\tilde{E}_2(\mathbf{u},\mathbf{c};\mathbf{u}^\nu) = & \int_\Omega \sum_{k=1}^{M} \left\{ \left[d^2(f(x),c_k) + \lambda \int_{B(x;\omega)} \sum_{j=1,j\neq k}^{M} u_j^\nu(y)\,\mathrm{d}y \right] u_k(x) \right\} \mathrm{d}x, \\ = & \int_\Omega \mathbf{D}(x,\mathbf{c},\mathbf{u}^\nu(x)) \cdot \mathbf{u}(x)\,\mathrm{d}x,\end{aligned}$$

and ν is the iteration number. Due to the linearization, one can use proposition 1 to get $\mathbf{u}^{\nu+1}$, i.e. the function given by:

$$u_k^{\nu+1}(x) = \begin{cases} 1, & k = \min\{k_1 : k_1 \in \mathbb{K}_x\}, \\ 0, & else. \end{cases} \tag{11}$$

is a minimizer of (8). Here

$$\begin{aligned}\mathbb{K}_x &= \{k : D_k(x,\mathbf{c}^\nu,\mathbf{u}^\nu) \leqslant D_i(x,\mathbf{c}^\nu,\mathbf{u}^\nu), \forall i \in \mathbb{L}\} \\ &= \left\{k : \sqrt{D_k(x,\mathbf{c}^\nu,\mathbf{u}^\nu)} \leqslant \sqrt{D_i(x,\mathbf{c}^\nu,\mathbf{u}^\nu)}, \forall i \in \mathbb{L}\right\} \\ &= \left\{k : \sqrt{d^2(f(x),c_k^\nu) + \lambda\tilde{n}_k(x)} \leqslant \sqrt{d^2(f(x),c_i^\nu) + \lambda\tilde{n}_i(x)}, \forall i \in \mathbb{L}\right\}.\end{aligned} \tag{12}$$

The last equation in (12) is derived from (7)

$$\begin{aligned}D_k(x,\mathbf{c}^\nu,\mathbf{u}^\nu) &= d^2(f(x),c_k^\nu) + \lambda \int_{B(x;\omega)} \sum_{j=1,j\neq k}^{M} u_j^\nu(y)\,\mathrm{d}y \\ &= d^2(f(x),c_k^\nu) + \lambda \int_{B(x;\omega)} (1 - u_k^\nu(y))\,\mathrm{d}y \\ &= d^2(f(x),c_k^\nu) + \lambda(|B(x;\omega)| - |B(x;\omega)\cap\Omega_k|).\end{aligned}$$

Comparing (11)–(12) with (4)–(5), it is clear that the function u_k in equation (11) is the characteristic function of the Voronoi region Ω_k determined by (5). Therefore, the iterative solutions from (8)–(9) is equivalent to the one from EWCVT clustering algorithm 2. In section 3.4, we will show that the iteration scheme (8)–(9) has the property:

$$E_2(\mathbf{u}^\nu,\mathbf{c}^\nu) \geqslant E_2(\mathbf{u}^{\nu+1},\mathbf{c}^{\nu+1}),$$

which implies the energy E_2 does not increase during this iteration.

3.2. The Proposed Models

In this section, we generalize CVT-based method in several ways.

3.2.1. Improved EWCVT

As mentioned earlier, even though the spatial smoothness of the clusters is considered in EWCVT, it still lacks image edge information. We introduce the following energy which combines the image intensity, the spatial smoothness of each cluster, and the image edge information together:

$$E_3(\mathbf{u},\mathbf{c}) = \int_\Omega \sum_{k=1}^{M} \left\{ \left[d^2(f(x),c_k) + \lambda \int_{B(x;\omega)} g(y)(1-u_k(y))\,\mathrm{d}y \right] u_k(x) \right\} \mathrm{d}x. \qquad (13)$$

Here g is an edge detection function and is often taken as $g = \frac{1}{1+|\nabla f|}$. We will introduce a nonlocal version of the edge function later. Compared with EWCVT, minimizing the above energy can improve the segmentation results such that the segmented objets have more accurate edges. For convenience, we refer this model as IEWCVT hereafter.

3.2.2. The intensity inhomogeneous image segmentation with Constraint Optimization

Intensity inhomogeneity always appears in nature images. It is difficult to extract the objects accurately from the images with intensity inhomogeneity. However it can often still be recognized by human eyes. The reason is that the pixels in the object are still different from these in their neighborhoods. Thus local information may help us to overcome this difficulty. There are some methods (e.g. [35, 36, 37, 38, 42, 44]) to handle the intensity inhomogeneity problem. The method we will propose is very fast and can handle egde information as well.

For real images, the brightness of an object is often varied. The contrast between the target and the background is always greater than the contrast inside the bright or the dark areas. Thus it is advantageous to enhance the local contrast of the images with intensity inhomogeneity. An efficient tool for sharpening the contrast is the histogram equalization. Contrast-limited adaptive histogram equalization (CLAHE) [39] is a variant of histogram equalization which operates on small regions in the image, called tiles, rather than the entire image. Each tile's contrast is enhanced, so that the histogram of the output region approximately matches the histogram specified by the given distribution (usually is the uniform distribution). Compared with the standard histogram equalization, the sharpened contrast in the CLAHE, especially in homogeneous areas, can be limited to avoid amplifying any noise that might be present in the image. There is a function called

"adapthisteq" for CLAHE implementation in the image processing toolbox of Matlab. We will use this function.

Let us denote $\tilde{f}$ as the output of CLAHE. In Figure 1, we display one test example by CLAHE. As can be seen from this test, the contrast of the original image in the top half has been improved through CLAHE. However, the brightness of the circles in the lower part of the image is still higher than that in the top. Generally speaking, segmenting $\tilde{f}$ directly would still produce undesirable results (c.f. Figure1(e)). On the hand, $h = \tilde{f} - f$ is an image which has the contrary contrast relative to f. This means that the areas with high contrast in the original image would have low contrast in h. A result for segmenting h is illustrated in Figure1(f). We can see that the semi-circles with low contrast in the original image can be well segmented, but the others have failed.

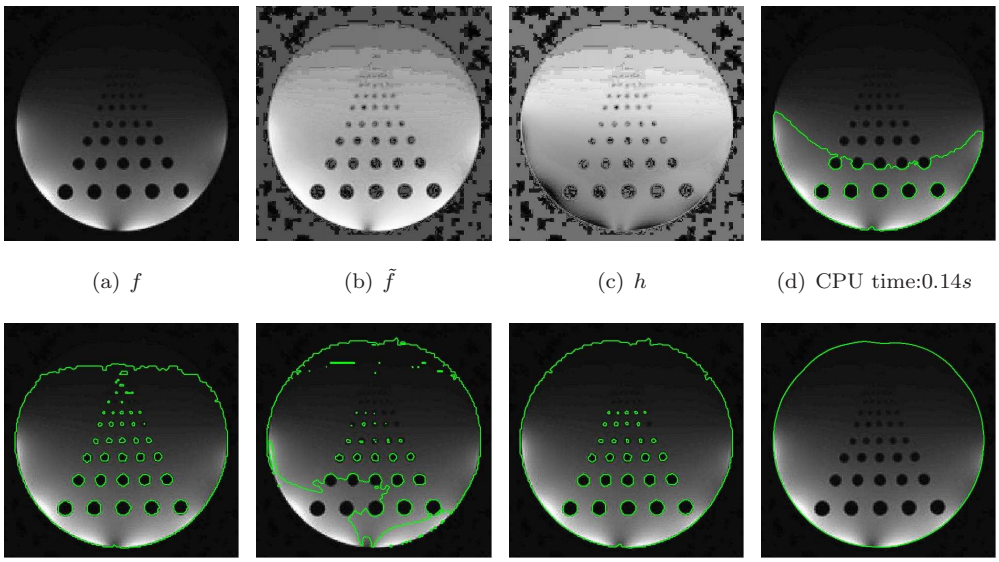
Based on the above observation, we suggest to use the following energy E_4 to segment images with intensity inhomogeneity:

$$\begin{aligned} &E_4(\mathbf{u}, \mathbf{c}, \tilde{\mathbf{c}}) \\ &= \int_\Omega \sum_{k=1}^{M} \left\{ \left[d^2(f(x), c_k) + \beta d^2(h(x), \tilde{c}_k) + \lambda \int_{B(x;\omega)} g(y)(1 - u_k(y))\, \mathrm{d}y \right] u_k(x) \right\} \mathrm{d}x, \end{aligned} \tag{14}$$

where β is a parameter which controls the balance between f and h. In order to balance the different scales equally, the images f and h are usually both normalized in $[0, 1]$. The segmentation results by minimizing the proposed energy with different regularization parameters are displayed in Figure1(g) and Figure1(h). More experimental results and comparisons with other methods will be given in section 4.

3.2.3. Texture Segmentation with CVT

For texture image segmentation, a key step is to define a proper texture descriptor. There are many different texture feature description methods in the literatures, including Gabor filters [28], structure tensor [29, 30], Beltrami framework [31, 32, 33]. For simplicity and efficiency, we choose the semi-local region texture descriptor which is proposed in [33] to describe the image texture feature in this paper. The semi-local region texture descriptor, says T, is defined as follows: an image $\mathbf{f} : \Omega \subset \mathbb{R} \mapsto \mathbb{R}^d$ can be viewed as a hypersurface in $\mathbb{R}^{d+2}$ with a parametric representation (the parameter is $x = (x_1, x_2)^{\mathrm{T}}$.) $\mathcal{S} : (x_1, x_2) \mapsto (x_1, x_2, \mathcal{P}_x(\mathbf{f}))^{\mathrm{T}}$, where $\mathcal{P}_x$ is a $(2r + 1) \times (2r + 1)$ square neighborhood



(a) f (b) $\tilde{f}$ (c) h (d) CPU time:0.14s

(e) CPU time:0.17s (f) CPU time:0.21s (g) CPU time:0.70s (h) CPU time:1.96s

Figure 1: The effects of CLAHE and different segmentation results with IEWCVT (image size: 447×447, 2 clusters). (a) the original image f ; (b) the output of CLAHE, i.e. $\tilde{f}$ =adapthisteq(f,'NumTiles',[2 2],'ClipLimit',0.2); (c) $h = \tilde{f} - f$; (d)-(f) segmentation results for f, $\tilde{f}$, and h respectively. $\omega = 3, \lambda = 1.5$; (g)-(h) segmentation results with the proposed energy (14) but with different parameters. $\omega = 3$ for (g) and $\omega = 19$ for (h), for both of the two cases, we have used $\beta = 1, \lambda = 1.5$.

centered at $x = (x_1, x_2)$, i.e. $\mathcal{P}_x(\mathbf{f}) = \{\mathbf{f}(x_1 + \tau, x_2 + \tau) : \tau \in [-r, r]\}$. Then the first fundamental form of this hypersurface is given by

$$I = \begin{pmatrix} \mathrm{d}x_1 \\ \mathrm{d}x_2 \end{pmatrix} \begin{pmatrix} E & F \\ F & G \end{pmatrix} \begin{pmatrix} \mathrm{d}x_1, & \mathrm{d}x_2 \end{pmatrix},$$

where

$$E = \mathcal{S}_{x_1} \cdot \mathcal{S}_{x_1}, \quad F = \mathcal{S}_{x_1} \cdot \mathcal{S}_{x_2}, \quad G = \mathcal{S}_{x_2} \cdot \mathcal{S}_{x_2},$$

and

$$\mathcal{S}_{x_1} = (1, 0, \partial_{x_1}\mathcal{P}_x)^{\mathrm{T}}, \quad \mathcal{S}_{x_2} = (0, 1, \partial_{x_2}\mathcal{P}_x)^{\mathrm{T}}.$$

The texture descriptor T in [33] is defined as

$$T = \exp\{-\frac{EG - F^2}{\sigma^2}\}.$$

Here we choose

$$T = \frac{1}{EG - F^2} = \frac{1}{1 + \sum_{\tau=-r}^{r} |\nabla f(x + \tau)|^2} \tag{15}$$

as the texture descriptor. Clearly, if $r = 0$, then $EG - F^2 = 1 + |\nabla \mathbf{f}|^2$ and T is the often used edge detector function $g = \frac{1}{1+|\nabla \mathbf{f}|^2}$. In fact, the texture descriptor T is equivalent to smooth the gradient image with the $(2r+1) \times (2r+1)$ square kernel, thus for larger r, there are more discriminative information between the texture object and the background. However a larger patch would cause a loss of accuracy in the segmentation. We will propose a CVT-based model to improve the accuracy of the segmentation result by adding the edge information of the original texture image. Let us first introduce a nonlocal edge detector function. Inspired by the definition of T, we choose $\mathcal{P}_x$ as the nonlocal version

$$\mathcal{P}_x(\mathbf{f}) = \{W(x, y)\mathbf{f}(y) : y \in N_x\},$$

where N_x stands for a searching window centered at $x = (x_1, x_2)$ with size $n_1 \times n_2$, and the weighting function $W(x, y)$ is known and given by (see e.g. [34]),

$$W(x, y) = c_0 \exp\left\{\frac{-||P_x - P_y||^2}{\sigma_2^2}\right\}.$$

Here $||\cdot||$ denotes the norm between two $n_2 \times n_2$ image patches $P(x)$ and $P(y)$, which are centered at $x = (x_1, x_2)$ and $y = (y_1, y_2)$ respectively, σ_2 is a parameter, and c_0 is a normalizing factor such that $\int W(x, y)\mathrm{d}y = 1$.

Then, the nonlocal edge detector function can be defined as

$$g(x) = \frac{1}{EG - F^2} = \frac{1}{1 + \sum_y W^2(x,y)|\nabla_x f(y)|^2}. \tag{16}$$

Combining the texture feature descriptor T, we propose the following energy for texture segmentation:

$$E_5(\mathbf{u}, \mathbf{c}) = \int_\Omega \sum_{k=1}^{M} \left\{ \left[d^2(T(x), c_k) + \lambda \int_{B(x;\omega)} g(y)(1 - u_k(y))\,\mathrm{d}y \right] u_k(x) \right\} \mathrm{d}x,$$

where T, g are defined by (15) and (16) respectively.

3.2.4. Block-Based CVT Image Segmentation

The segmentation methods we have just discussed are point-based. The energy can be naturally extended to block-based one with:

$$E_6(\mathbf{u}, \mathbf{c}) = \sum_{k=1}^{M} \int_\Omega \int_\Omega k(x-y) d^2(f(y), c_k) \mathrm{d}y u_k(x)\,\mathrm{d}x, \tag{17}$$

where k is always a Gaussian kernel or a compactly supported function such as $k(x) = \frac{1}{|B(x;\omega)|}\chi_{B(x;\omega)}$.

Let us analysis this energy. Suppose d is the often used L^2 norm, i.e $d^2(f(y), c_k) = |f(y) - c_k|^2$. Denote $\bar{f}(x) = \int_\Omega k(x-y) f(y)\,\mathrm{d}y$. We know that $\int k(y)\mathrm{d}y = 1$. Thus

$$\begin{aligned} \int_\Omega k(x-y) d^2(f(y), c_k)\mathrm{d}y &= \int_\Omega k(x-y)|f(y) - \bar{f}(x) + \bar{f}(x) - c_k|^2 \mathrm{d}y \\ &= \int_\Omega k(x-y)|f(y) - \bar{f}(x)|^2 \mathrm{d}y + |\bar{f}(x) - c_k|^2. \end{aligned}$$

The value of the first term in the above equation is often very small, and if we ignore this term in the energy, then (17) becomes

$$E_6(\mathbf{u}, \mathbf{c}) = \sum_{k=1}^{M} \int_\Omega |\bar{f}(x) - c_k|^2 u_k(x)\,\mathrm{d}x.$$

As $\bar{f}$ is a denoised image with a smoothing kernel k, the minimization of the energy (17) is equivalent to segment the denoised image. Thus, it is natural that it is more robust to noise than the standard CVT clustering.

An interesting observation has been observed for this block-based segmentation model. When the center c_k is a function, namely

$$E_6(\mathbf{u}, \mathbf{c}) = \sum_{k=1}^{M} \int_\Omega \int_\Omega k(x-y) d^2(f(y), c_k(y)) \mathrm{d}y u_k(x)\,\mathrm{d}x, \tag{18}$$

(a) Original image (b) CVT [17] (c) EWCVT [18] (d) by (18) (e) $c_1(y)$ and $c_2(y)$

Figure 2: A comparison of the results obtained by CVT, EWCVT, and minimizing the energy (18) (2 clusters). (a) the original image; (b) segmentation result with CVT; (c) segmentation result with EWCVT, $\lambda = 0.02, \omega = 5$; (d) segmentation result by minimizing (18), $k = \frac{1}{|B(\omega)|}\chi_{B(\omega)}$ and $B(\omega)$ is token as a 21×21 square; (e) the finally estimated $c_1(y)$ and $c_2(y)$ respectively (size 21×21).

the model can produce segmentation results that can connect broken fringes in a regular fringe image. See Figure2 for the experimental results.

3.3. A Fast Narrow Banding Algorithm

In this section, we explain some details about a fast algorithm for minimizing the proposed energy.

Consider the following constraint optimization problem:

$$(\mathbf{u}^*, \mathbf{c}^*) = \arg\min_{\mathbf{u},\mathbf{c}} \left\{ E(\mathbf{u}, \mathbf{c}) = \int_\Omega \mathbf{D}(x, \mathbf{u}(x), \mathbf{c}) \cdot \mathbf{u}(x) \mathrm{d}x \right\} \quad \text{s.t.} \sum_{k=1}^{M} u_k = 1, u_k \geqslant 0. \tag{19}$$

As we have discussed, we can use the following iterative scheme to solve it:

$$\mathbf{u}^{\nu+1} = \arg\min_{\mathbf{u}} \quad \tilde{E}(\mathbf{u}, \mathbf{c}^\nu; \mathbf{u}^\nu), \quad \text{s.t.} \sum_{k=1}^{M} u_k(x) = 1, u_k(x) \geqslant 0, \tag{20}$$

$$\mathbf{c}^{\nu+1} = \arg\min_{\mathbf{c}} \quad \tilde{E}(\mathbf{u}^{\nu+1}, \mathbf{c}; \mathbf{u}^{\nu+1}), \tag{21}$$

$$\nu = 0, 1, 2, \cdots, \tag{22}$$

where

$$\tilde{E}(\mathbf{u}, \mathbf{c}; \mathbf{u}^\nu) = \int_\Omega \mathbf{D}(x, \mathbf{u}^\nu, \mathbf{c}(x)) \cdot \mathbf{u}(x) \mathrm{d}x.$$

The above two minimization problems are both easy to solve with close-form solutions. Applying Proposition 1, we obtain the following algorithm for this model:

Algorithm 3. *Given the number of classes M, choosing an initial cluster centers $\{c_k\}_{k=1}^M$, and obtain an initial label function $l(x)$ by a CVT scheme:*

Step 1, update $l(x)$ from (20) using the values of c_k: $\forall k = 1, 2, \cdots, M$, compute $D_k(x)$ and find the smallest $D_k(x)$ at each x; if the smallest $D_k(x)$ is not unique, choose the smallest index with respect to k among them; let us denote it as $D_{k_1}(x)$; then assign $l(x) = k_1$, i.e. $l(x) = \min\{k_1 : \quad k_1 \in \mathbb{K}_x\}$, where $\mathbb{K}_x = \{k : D_k(x) \leqslant D_i(x), \forall i \in \mathbb{L}\}$.
Step 2, update c_k according to $l(x)$ and equation (21).
Step 3 if converged, end the algorithm; else, go to step 1.

This algorithm has some special properties. When a pixel x and all of its neighborhood pixels in $B(x;\omega)$ can be initially assigned to the same j-th cluster by the standard CVT algorithm, then we have $d^2(f(x), c_j) \leqslant d^2(f(x), c_i), i \in \mathbb{L} \setminus \{j\}$. In this case, the smoothness term $\tilde{n}_j(x) = 0$ and $\tilde{n}_i(x) = |B(x;\omega)|$. Obviously,

$$d^2(f(x), c_j) + \lambda \tilde{n}_j(x) < d^2(f(x), c_i) + \lambda \tilde{n}_i(x),$$

i.e. $D_j(x) < D_i(x)$ holds for any $\lambda > 0$. Thus according to algorithm 3, x will be still assigned to the j-th cluster independent of the value of λ. This means that we do not need to calculate D_k and reassign the label function $l(x)$ for these pixels. Therefore the speed of the algorithm can be accelerated by using a narrow band technique.

We use the following method to detect the narrow band: let $\tilde{l}(x) = |(G_\sigma * l)(x) - l(x)|$, where $G_\sigma(x) = \frac{1}{2\pi\sigma^2} \exp\{-\frac{||x||^2}{2\sigma^2}\}$ is a Gaussian kernel with the standard deviation σ, if $\tilde{l}(x) > \varepsilon$, then x can be regarded as an boundary points near the boundaries of the clusters. The width of the narrow band is controlled by the parameters σ and ε. A larger σ will be used for the wider computational band. As for ε, a smaller value will give a wider band. In this paper, unless otherwise specified, we set the parameter $\sigma = 3.0, \varepsilon = 0.01$.

To summarize, the narrow band algorithm for the proposed models can be written as follows:

Algorithm 4 (Narrow Band Algorithm). *Given the number of classes M, choosing an initial cluster centers $\{c_k\}_{k=1}^M$, and obtain an initial label function $l(x)$ by any CVT iteration, calculate $\tilde{l}(x) = |(G_\sigma * l)(x) - l(x)|$. Let energy $E_{old} = +\infty$:*
Step 1, set energy $E_{new} = 0$. For all $x \in \Omega$, if $\tilde{l}(x) > \varepsilon$, then compute $D_k(x)$ and set $l(x) = \min\{k_1 : \quad k_1 \in \mathbb{K}_x\}$, where $\mathbb{K}_x = \{k : D_k(x) \leqslant D_i(x), \forall i \in \mathbb{L}\}$. Update $E_{new} = E_{new} + D_{l(x)}(x)$; Otherwise, do nothing.

Step 2, update c_k according $l(x)$.

Step 3, if $|E_{new} - E_{old}|^2 < 0.01E_{old}$, end the algorithm; else let $E_{old} = E_{new}$ and go to step 1.

We want to emphasis that this algorithm is very easy to implement and it is very fast. Its high efficiency will be demonstrated by numerical experiments given later.

3.4. Some Analytical Results for the Algorithm and Model

The proposed algorithm has some very nice properties. The analysis of these properties will be given in the appendixes. We state these properties in the following two propositions.

Proposition 2 (Energy descent). *The sequence $(\mathbf{u}^\nu, \mathbf{c}^\nu)$ generated by (20),(21) satisfies $E(\mathbf{u}^\nu, \mathbf{c}^\nu) \geqslant E(\mathbf{u}^{\nu+1}, \mathbf{c}^{\nu+1})$, where $\nu = 0, 1, 2, \cdots$.*

Corollary 1. *The algorithm 3 and the narrow band algorithm 4 are both convergent.*

Proposition 3 (The existence of the binary solution). *if $(\mathbf{u}^*, \mathbf{c}^*)$ is a global minimizer of (19), then there must exist a binary $\hat{\mathbf{u}}$ such that $E(\hat{\mathbf{u}}, \mathbf{c}^*) = E(\mathbf{u}^*, \mathbf{c}^*)$, i.e. $(\mathbf{u}^*, \mathbf{c}^*)$ is also a global minimizater of (19).*

Proofs of the propositions will be given in Appendix B and C. Similar conclusions also can be found in [18].

4. Experimental Results

In this section, we use several experiment example to show the properties and efficiency of the models and algorithms. The implementation of all the algorithms is done in Matlab (mex files) on a laptop with Intel 2.40G CPU.

In all the experiments, the original images f are all normalized in [0,1] and the neighborhood $B(x;\omega)$ in the regularization term is taken as a circle with radius ω.

4.1. Tuning of the Parameters

There are several parameters that need to be tuned. The parameter ω in the regularization term is determined by noise and the scale of the objects that need to be eliminated. It should always be larger than the radius of the undesired objects. The λ controls the ratio between the data driven term and smoothness term. For the choice of λ, one can refer to [18].

For intensity inhomogeneity model, two of the most important parameters in the "Contrast-limited adaptive histogram equalization" (Matlab function "adapthisteq") are the number of tiles "NumTiles" and the contrast enhancement limit parameter "ClipLimit". Generally speaking, larger parameter values of "NumTiles" and "ClipLimit" can be more accurate in finding details such as weak edges in the image. For more details about how to choose these parameters, one can refer to the help documents of matlab. β is a parameter to control the weights of the original image information and enhanced local information. In most cases, we choose β to be around 1.0. If the image has little intensity non-uniformity, then β should be decreased. r used in the texture model should always be bigger than the scale of the texture structure. For natural images used in this paper, it can be set to be 9 or 11.

4.2. Experiment I: Comparison between EWCVT and IEWCVT

The main difference between EWCVT and IEWCVT (13) is the regularization term. We shall show effect of this term with different regularization parameters. There are two parameters λ, ω in both models to control the smoothness of the clusters. We fix λ with a relatively large value $\lambda = 100$ in this experiment. In addition, the edge detector function g in IEWCVT is still selected as the traditional one $g = \frac{1}{1+|\nabla f|^2}$ to improve the computation efficiency. For the purpose of unifying the scale, we normalize g in $[0, 1]$.

In Figure3, we show some segmentation results with EWCVT and IEWCVT. In this experiment, the synthetic image need to be segmented into 4 groups. When ω is very small ($\omega = 3$), both of the models can provide good results (see Figure3(b) and Figure1(e)). However, there are still some burrs on the boundaries, especially in Figure3(b). One can choose a larger ω to remove these burrs. We show the segmentation results produced by both models with parameter $\omega = 9$ in the middle column. Compared

with the first choice, there is a bifurcation appeared in the center of Figure3(c), even though the boundaries of the clusters in Figure3(c) and Figure3(f) are more straighter than before. When we increase the neighborhood parameter to $\omega = 15$, this bifurcation would be enlarged (c.f. Figure3(d)). On the other hand, the IEWCVT can accurately find the clusters with increased ω. In the following, we will give a theoretical explanation about this phenomenon observed with EWCVT in this experiment.

First of all, the regularization term $E_L = \sum_{k=1}^{M} \int_{\Omega} u_k(x) \int_{B(x;\omega)} \sum_{j=1, j\neq k}^{M} u_j(y)\, \mathrm{d}y\, \mathrm{d}x \propto$ length(Γ) if Γ is the boundary curve of the clusters, c.f. [18]. Thus we only need to show that the length of the clusters boundaries produced by EWCVT in Figure3(d) is shorter than the length of the real boundaries. As shown in Figure4, we only need to prove that:

$$(AP + BP + OP) + (CQ + DQ + OQ) < AC + BD.$$

The above inequality holds when P, Q are the Fermat points of the $\triangle ABO$ and $\triangle CDO$ respectively, i.e. when

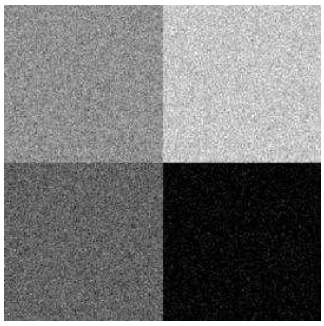
$$\angle APB = \angle BPO = \angle APO = 120^o, \quad \angle CQD = \angle CQO = \angle DQO = 120^o.$$

In fact the angles between boundaries of the clusters in Figure3(c) and Figure3(d) are all nearly 120^o. It is amazing to see that our algorithm is able to get the angles correctly. This is difficult for many algorithms designed in the literature.

4.3. Experiment II: image segmentation with intensity inhomogeneity

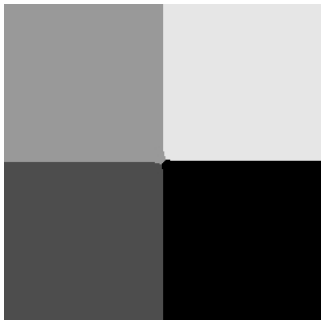
In this experiment, we test the model described in section 3.2.2 and also compare it with other approaches. In [36, 37], the authors proposed a level set method driven by local binary fitting (LBF) energy to hand the intensity inhomogeneous problem, we compare our method with theirs.

In Figure5, we display the segmentation results obtained by EWCVT [18], LBF [37], and our method for images with intensity inhomogeneities. EWCVT can not find the correct segmentation. Both of LBF in [37] and our method can produce good results with similar quality. From Table2, which records the CPU time for both method, one can see clarly that our method is much faster than LBF. In addition, all the parameters values used in this experiment are given in Table 1.



(a) Synthetic image, size 268 × 269.

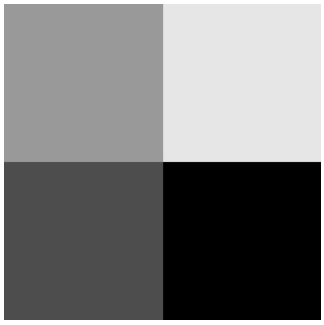
(b) EWCVT [18], $\omega = 3$



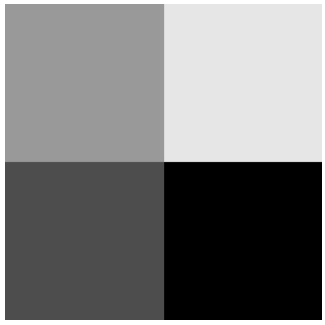
(c) EWCVT [18], $\omega = 9$

(d) EWCVT [18], $\omega = 15$

(e) IEWCVT, $\omega = 3$



(f) IEWCVT, $\omega = 9$



(g) IEWCVT, $\omega = 15$

Figure 3: Comparison between EWCVT [18] and the proposed IEWCVT. (a) Original image; (b)-(d) the clustering results by EWCVT; (e)-(g) the clustering results by IEWCVT.

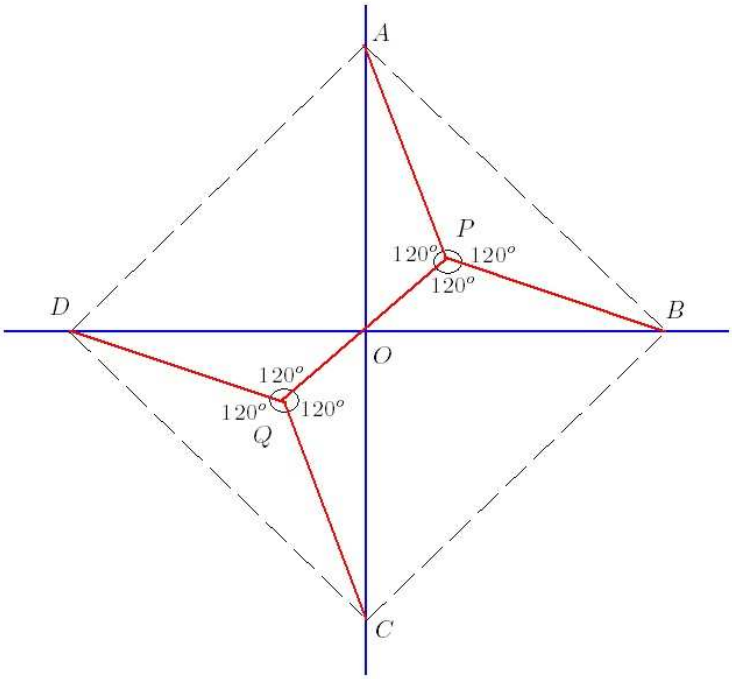


Figure 4: A diagram for explaining the results in Figure3(c) and Figure3(d)

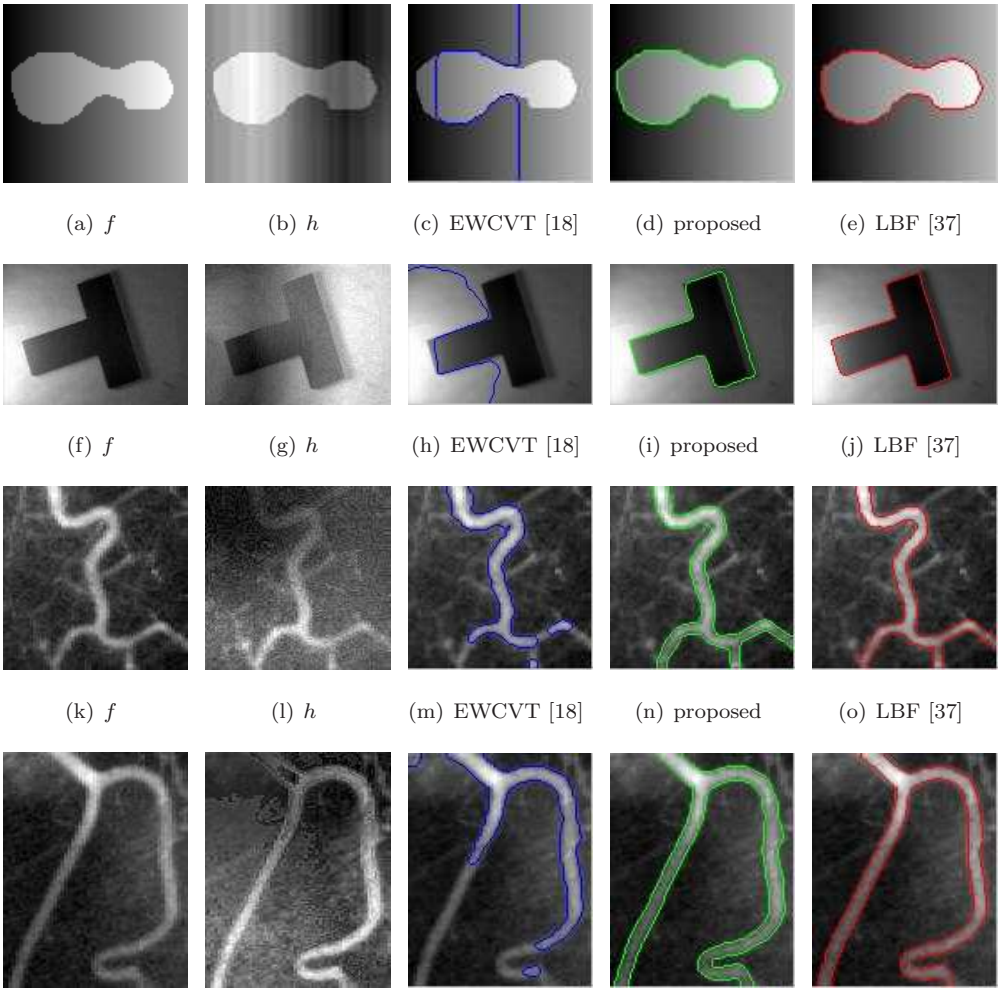
.

Unlike the popular level set based method which is cannot be easily generalized to cases with more than two clusters (such as LBF [36, 37]), another superiority of the CVT-based approach is that it can handle multi-clusters easily. In Figure6, we give a 3-clusters result for brain MR image with intensity inhomogeneity. The CPU time usage for this test is listed in Table2. One can see that our method is very efficient in term of CPU time usage and has great potential for real-time applications.

Our method can be easily applied to 3-D image segmentations. MR images are usually corrupted by a spatially varying bias field, which results in inhomogeneous intensity in the images. We test our algorithm on a 3-D data set downloaded from the BrainWeb (`http://mouldy.bic.mni.mcgill.ca/brainweb/`). Three slices of the original data with 40% intensity non-uniformity and 5% noise are displayed in the first column of Figure7. In this experiment, both the proposed model and EWCVT are used for brain segmentation. We use some skull striping technique to get the 1922592 (about $124 \times 124 \times 124$) voxels. Then we segment the brain into 3 clusters. The control neighborhood $B(x;\omega)$ is taken as a ball with radius 2. The other parameters values used in the experiment and the CPU time cost are listed in Table1, Table2. The segmentation results are shown in the last two columns of Figure7. As it can be seen, the results with model (14) are much better than EWCVT [18], especially in the dash line ellipse area. Since the ground truth data of this brain data is available in BrainWeb, then we can use a segmentation precision $p = \frac{n_{correct}}{n_{total}}$ to indicate the segmentation quality for these two methods. Here we only take white matter to evaluate and $n_{correct}$ is the number of white matter voxels which are segmented correctly and n_{total} corresponds to the number of the white matter in the ground truth. For EWCVT and the proposed methods, $p = 84.9\%, 94.2\%$ respectively. Finally, in order to show the algorithms performed on different regions, the results on multiple slices on one direction are displayed in Figure8.

4.4. Experiment III: texture image segmentation

The nonlocal edge detector function (16) is often slightly better than the conventional edge function when the images contaminated by noise and have repeatable structures. However, it is time-consuming to calculate the nonlocal weighting function. For the comparison between the nonlocal regularization and local regularization for image segmentation, we refer to [40]. In this experiment, we still use the conventional edge function



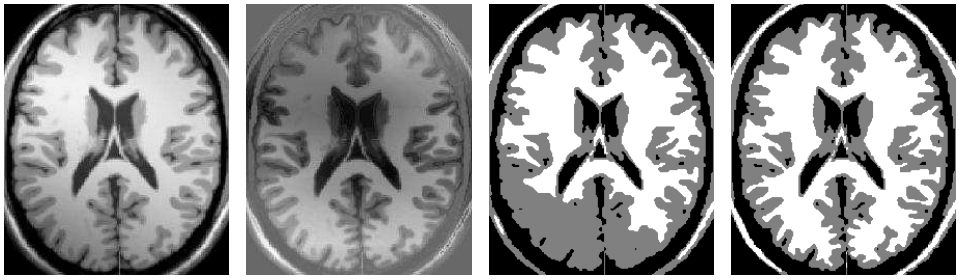
(a) f (b) h (c) EWCVT [18] (d) proposed (e) LBF [37]

(f) f (g) h (h) EWCVT [18] (i) proposed (j) LBF [37]

(k) f (l) h (m) EWCVT [18] (n) proposed (o) LBF [37]

(p) f (q) h (r) EWCVT [18] (s) proposed (t) LBF [37]

Figure 5: The results of intensity inhomogeneous image segmentation (2 clusters) obtained the proposed model (14) and other connected methods. Form left to right: first column, the original images f; second column, the preprocessed images h; third column, the results with EWCVT [18]; fourth column, the results obtained by the proposed method in section 3.2.2; fifth column, the results by LBF [37].



(a) f (b) h (c) EWCVT [18] (d) proposed

Figure 6: An example of 3-clusters segmentation with intensity inhomogeneous. (a), the original image; (b) the preprocessed image h; (c)clusters obtained by EWCVT; (d) clusters provided by the proposed method.

Table 1: The parameters values used in experiment II (Figure5,6,7).

Figure	CLAHE			Regularization	
	"NumTiles"	"ClipLimit"	β	λ	ω
Fig.5(a)	2×2	$1E-03$	1.0	1.0	3
Fig.5(f)	2×2	$5E-04$	0.7	1.0	5
Fig.5(k)	2×2	$5E-04$	1.0	1.0	5
Fig.5(p)	2×2	$5E-04$	1.0	1.0	5
Fig.6(a)	4×4	$1E-03$	1.1	0.01	3
Fig.7	4×4	$1E-02$	1.0	0.05	2

Table 2: A comparison of the the CPU time in experiment II (Figure5,6,7).

Figure	image size	EWCVT[18]	proposed			LBF[37]
			CLAHE	segmentation	total time	
Fig.5(a)	85×88	0.015s	0.006s	0.014s	0.020s	0.593s
Fig.5(f)	96×127	0.021s	0.005s	0.047s	0.052s	7.312s
Fig.5(k)	110×111	0.044s	0.005s	0.036s	0.041s	1.934s
Fig.5(p)	131×103	0.052s	0.006s	0.044s	0.050s	3.494s
Fig.6(a)	261×217	0.147s	0.018s	0.182s	0.200s	-
Fig.7	$181 \times 217 \times 181$	17.75s	20.02s	18.93s	38.95s	-

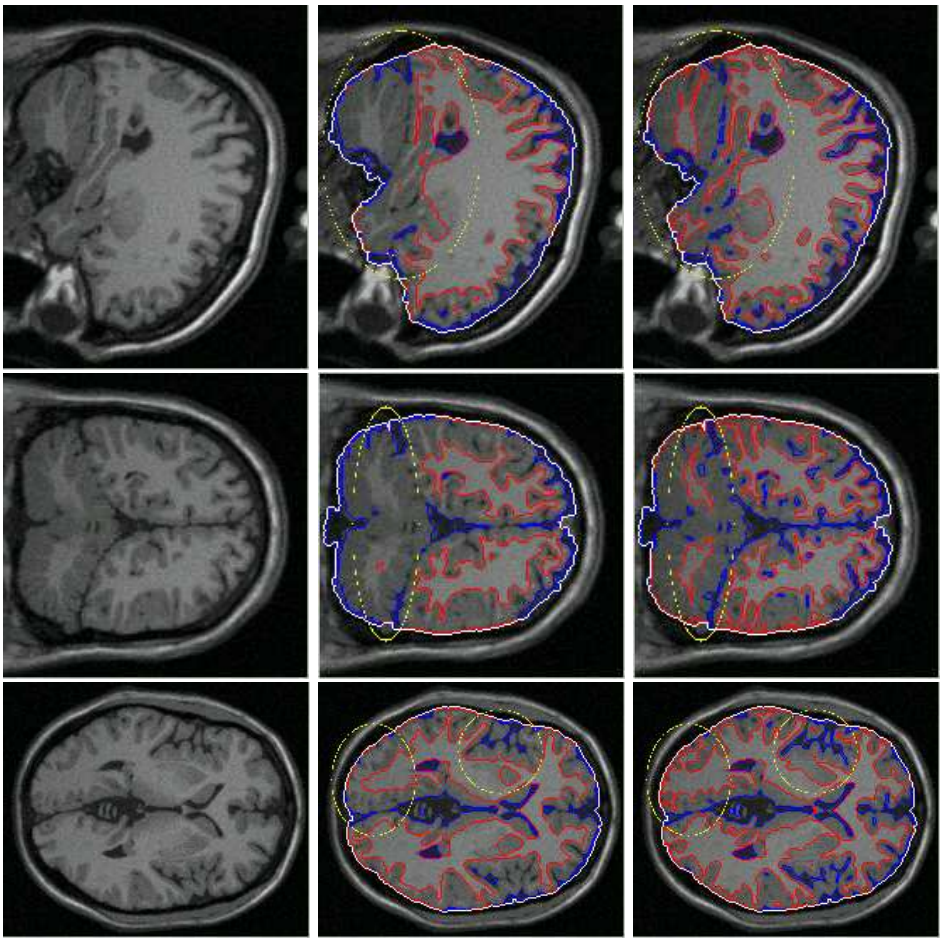


Figure 7: Intensity non-uniformity and noisy 3-D MR medical image segmentation with EWCVT and the proposed method (size:$181 \times 217 \times 181$, 3 clusters). First column, the slices of original image; Second column, segmentation results with EWCVT [18]; Third column, the results with ours. The white line represents the computation area, the blue and red lines are the boundaries of the segmented 3 clusters.

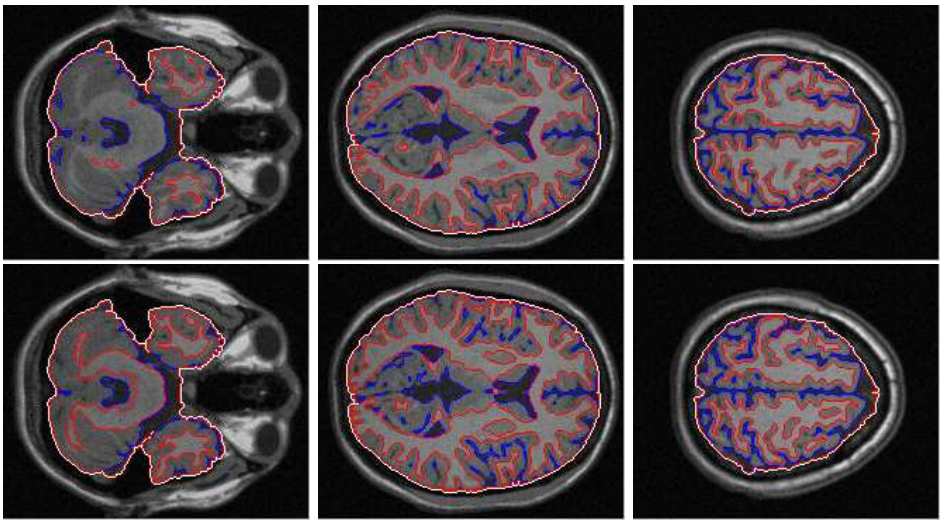


Figure 8: The results on multiple slices on 'z' direction for the 3D segmentation in Figure7. First row: the EWCVT results. Second row: the proposed results. From left to right, the images are x-y view with $z = 40, 80, 130$, respectively.

for the sake of computational efficiency.

To show the good properties of our algorithms, we shall compare it with some recently proposed efficient and global minimization techniques. Recently, some efficient and global minimization techniques has been proposed for segmentation [41, 42, 44]. In these models, the regularization term is a total variation of the minimization function, which is different from the one employed in this paper. These models can be minimized by many algorithms such as [43, 45]. In [33], the authors extended the models to texture segmentation. We shall compare our algorithm with that of [33].

Figure9 shows some results for texture segmentation. Comparing with the results between our method and the method of [33], one can see that the segmentation results obtained by the method of [33] are less accurate near the borders (See Figure10 for more details). It's easy to understand since the texture feature image T is dilated in the object boundaries relative to the original image when calculate T according (15). We add the edge information of the original image into the energy functional and thus could avoid this problem.

In addition, we plot the convergence of the energy functional in Figure11. This

Table 3: The CPU time and parameters values in experiment III (Figure9, image size 321×481).

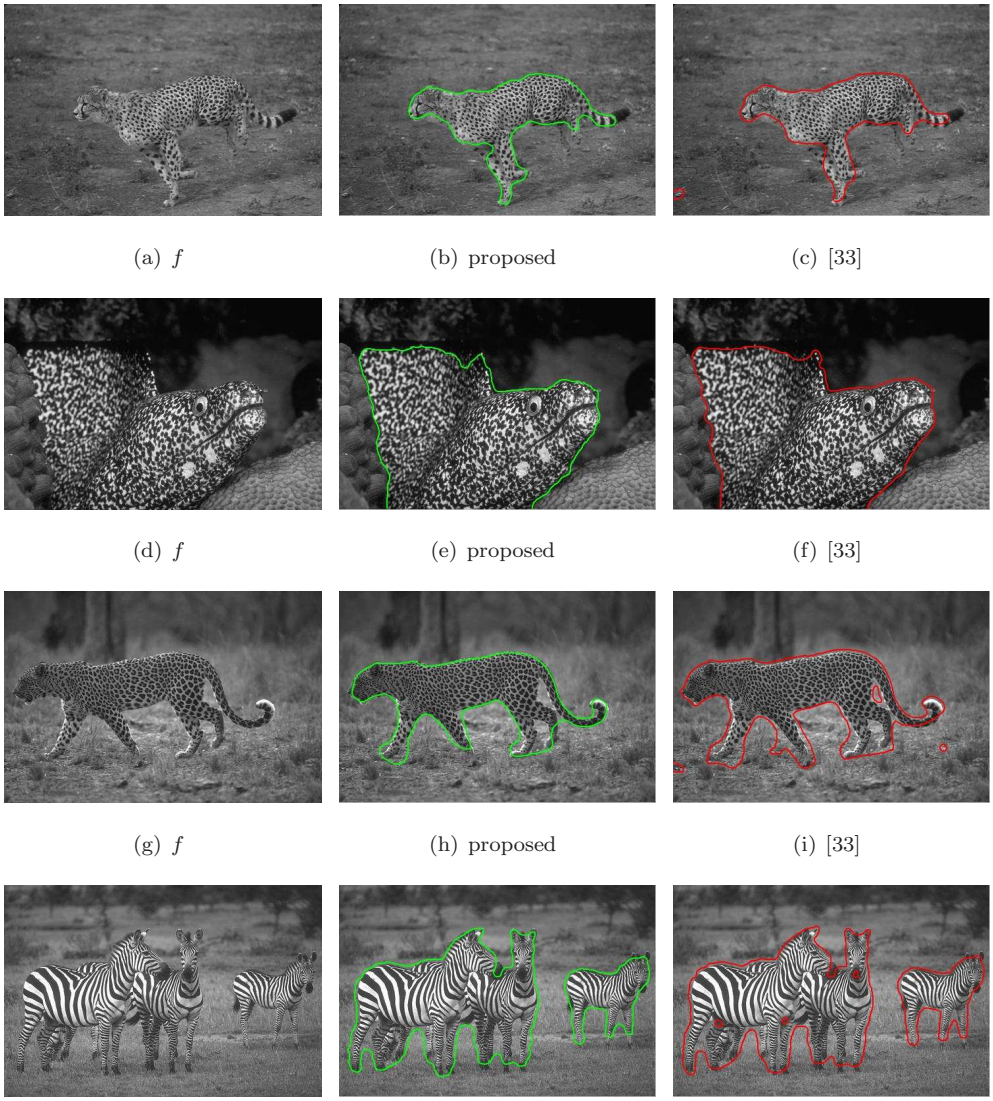
Figure	parameters			CPU time		
	r	ω	λ	calculate T	proposed	[33]
Fig.5(a)	11	7	0.4	0.06s	0.61s	2.03s
Fig.5(d)	11	7	0.5	0.06s	0.36s	2.37s
Fig.5(g)	9	9	0.133	0.05s	1.95s	5.44s
Fig.5(j)	11	7	0.15	0.06s	0.55s	2.15s

numerical verify that proposition 2 is correct. Finally, the values of parameters use this this experiment and the CPU time usage are displayed in Table3. Again, one can clearly see that the CPU time cost of our proposed segmentation method is much less than that of [33]. Let us mention the segmentation model in [33] is based on a probability density function, and it is often more time-consuming than Chan-Vese model [20]. Due to this, we will make some comparisons between the split Bregman iteration for Chan-Vese model [43] and our algorithm in the next experiment.

4.5. Experiment IV: a comparison between the split Bregman method for the Chan-Vese model and our proposed algorithm 4

The Split Bregman method of [43] is an efficient and fast algorithm for TV minimization. Let us mention again that the regularization term used in the Chan-Vese model [20] and our model is different. Though the TV term $\int_\Omega |\nabla u| \mathrm{d}x$ can be rewritten formality as $-\int_\Omega \nabla\cdot(\frac{\nabla u}{|\nabla u|})u\mathrm{d}x \triangleq \int_\Omega D\cdot u\mathrm{d}x$ according to the divergence theorem, the iteration scheme (20), (21) can not be applied to it since we can not guarantee that the energy descends with the iterations. Thus, proposition 2 will not be valid and the proposed algorithm can not used. For TV regularization, a good connections between the Gaussian mixture model and CVT model for image segmentation has been observed in [44] for multi-phase clustering.

All the comparisons between algorithm 4 and the split Bregman method for the Chan-Vese model [43], including the segmentation results and CPU time cost, are shown in Figure 12 and Table 4. Again, the accuracy and efficiency of our proposed algorithm is easy to see.



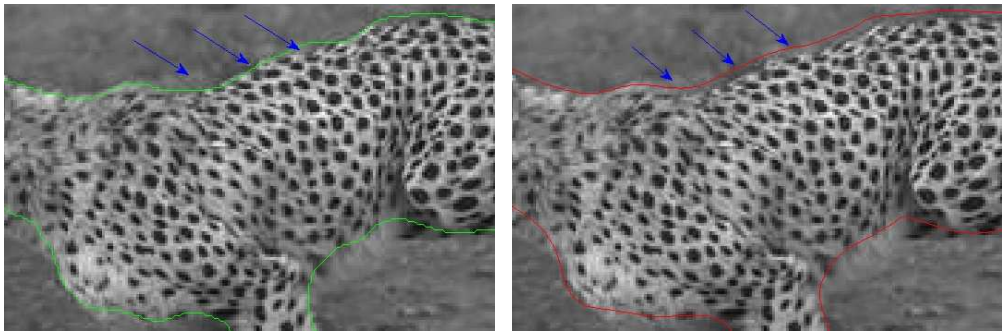
(a) f (b) proposed (c) [33]

(d) f (e) proposed (f) [33]

(g) f (h) proposed (i) [33]

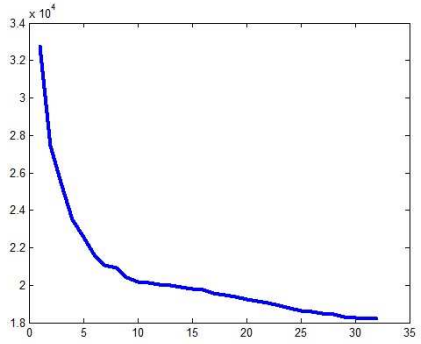
(j) f (k) proposed (l) [33]

Figure 9: Comparisons between the proposed method and the model in [33] on texture segmentation.

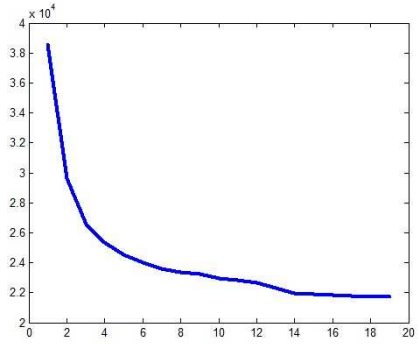


(a) proposed (b) [33]

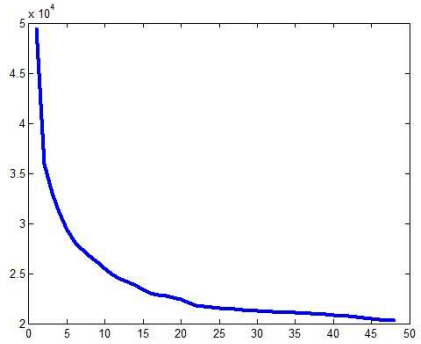
Figure 10: Partial enlarged views of Figure9(b) and Figure9(c) respectively.



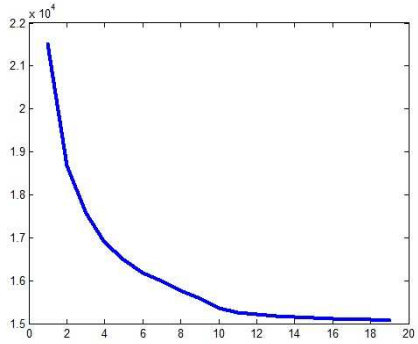
(a) The energy changes of Figure9(b)



(b) The energy changes of Figure9(e)

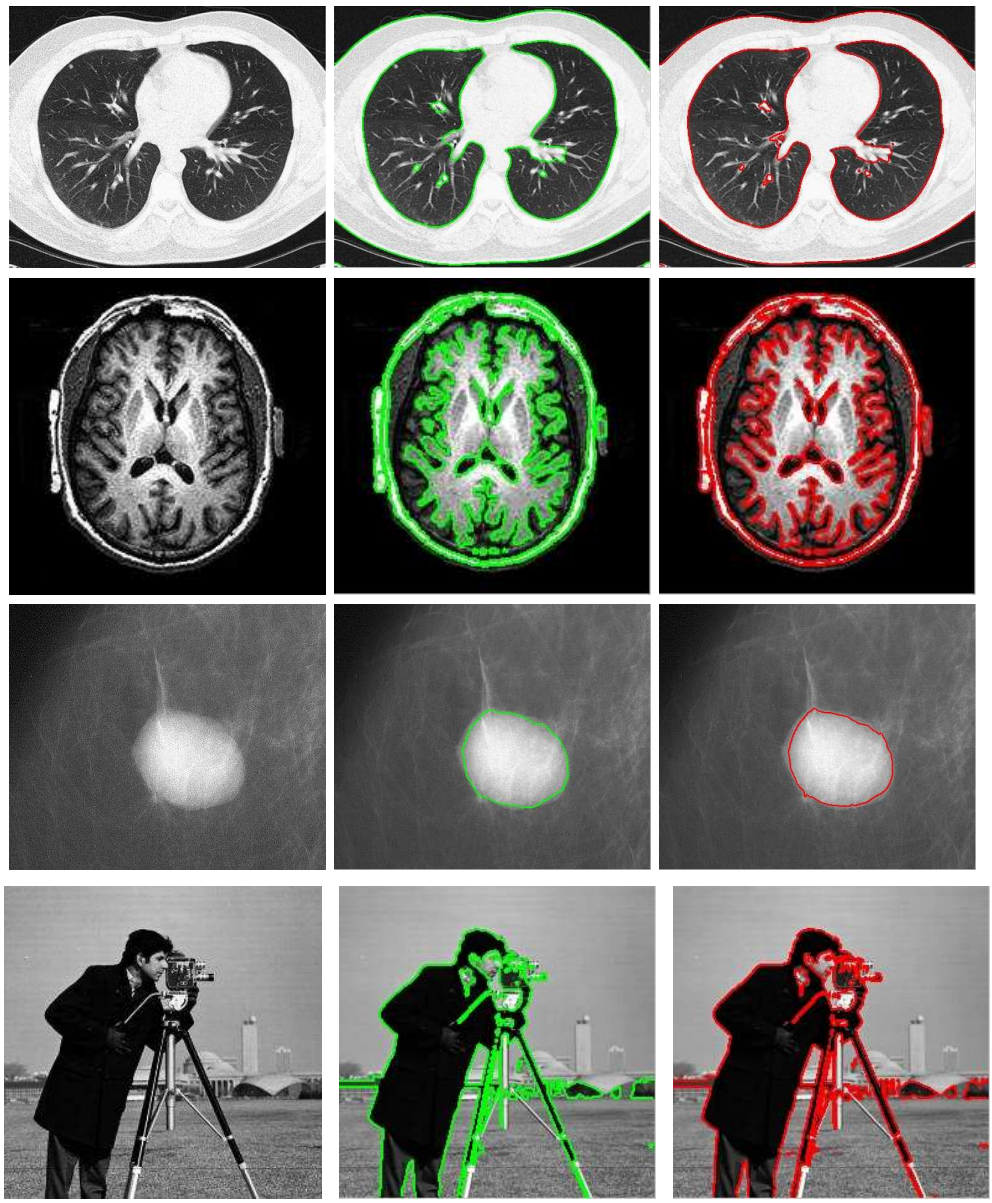


(c) The energy changes of Figure9(h)



(d) The energy changes of Figure9(k)

Figure 11: The variations of the energies during the iteration. x-axis: number of iterations; y-axis: the energy E.



(a) Original image (b) proposed Alg. 4 (c) Chan-Vese [43]

Figure 12: Results obtained by the proposed algorithm and the Chan-Vese model with split Bregman iteration [43].

Table 4: Comparisons between Alg. 4 and split Bregman iteration [43] (Figure12).

Fig. No.	image size	parameters		CPU	time
(from top to bottom)		ω	λ	Alg.4	[43]
1	337×407	7	0.15	0.421s	0.905s
2	210×210	3	0.1	0.031s	0.032s
3	403×481	7	0.1	0.140s	2.402s
4	253×256	3	0.1	0.047s	0.062s

4.6. Experiment V: Triple junctions

One essential advantage of our method is that it is efficient and at the same it can regularize the segmentation boundaries. Many methods have been proposed in the literature for interface problems. To show that a method can regularize the the boundaries according to their lengths, it is a tradition to test them for triple junction problems. In these tests, one takes away the fidelity term from (13) and only minimizes the regularization term with some proper boundary conditions for the label functions u_k. If a method can treat the length term correctly, then internal junctions must be triple junctions and must have 120^o between the boundaries at junctions [44, 46, 47]. This is a very challenging problem and it has been analysed intensively in the literature. It is closely related to the so-called "Steiner tree problem". It has many applications where one needs to find interfaces to minimize surface tensions. There are not so many interface methods that can solve this problem accurately and efficiently [47, 44, 46].

In all the experiments, the total number M of clusters is given. As in [46], we fix the values of $u_k, k = 1, 2, \cdots M$ on $\partial\Omega$ and minimize E_3 with $g = 1$ and without the data fidelity term, i.e. we minimize

$$E_L = \int_\Omega \sum_{k=1}^{M} u_k(x) \int_{B(x;\omega)} (1 - u_k(y))\, \mathrm{d}y\, \mathrm{d}x \tag{23}$$

with given boundary values for $u_k, k = 1, 2, \cdots M$. As u_k is a label function, its values on $\partial\Omega$ can only be 0 or 1.

In Figure 13, some numerical results with different phase number M are shown. The boundary condition used for the u_k's is clear from the plots in the figure. It is important to note that all the junctions are triple junctions and the angles around the junctions are

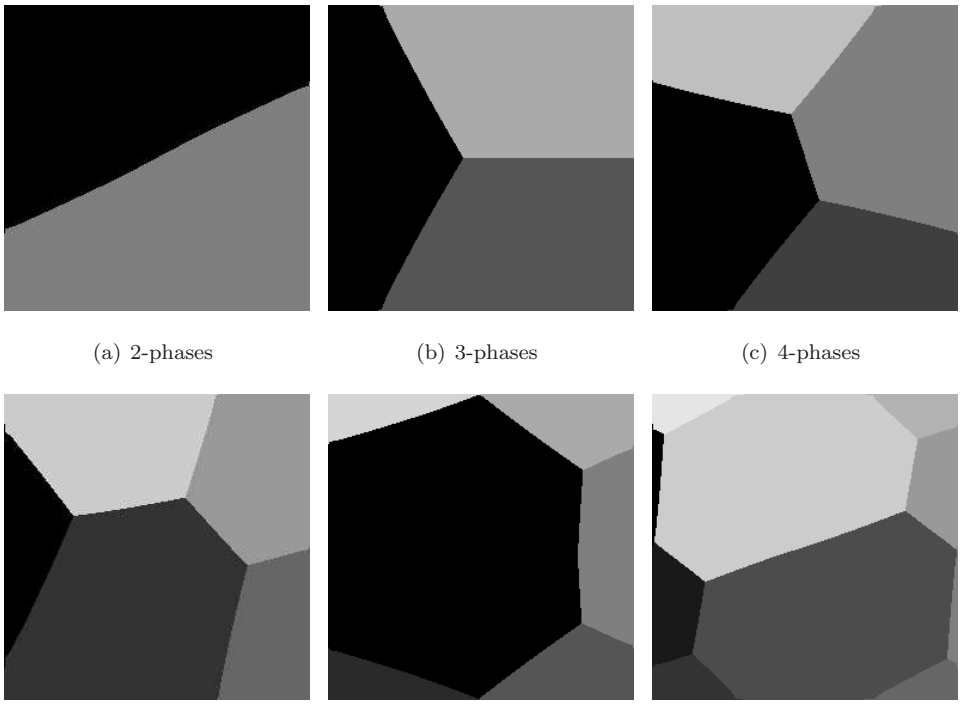
120^o. All the boundaries between the clusters are straight lines. We want to emphasis that the solution is not unique and the energy functional has many local minimums for $M > 3$. There are not so many known methods that can solve this problem efficiently for multiphase problems. Our method can get the correct solution for rather big phase number M. As the solution is not unique, our final solution could differ with different initial values.

To further valid our method, we test in on a junction problem with $M = 4$. This problem has two solutions that are analytically known, see Figure 14.(e)-(f). In this test, we assume that label functions $u_k, k = 1, 2, 3, 4$ are known outside the circle shown in Figure 14.(a). We need to find the values of u_k inside the circle that minimizes the length term E_3. This is equivalent to take Ω to be the disk in Figure 14.(a) and minimize the length term with fixed boundary values of u_k on the circle. With different initial values, our algorithm is alway able to converge to one of the solutions, see Figure 14.(b)-(g). This is remarkable. The algorithms proposed in [44, 47] are able to find triple junctions when $M = 3$ as in Figure 13.(b), but cannot find true solutions for this test example due non-uniqueness of the solution and local minimums for the energy functional.

5. Conclusion

In this paper, we proposed a new method to segment images with inhomogeneous intensity values with texture. These approaches are all based on minimizing a constrained optimization energy. An iterative scheme is used. It is guaranteed that the energy is decreasing. Moreover, all the subproblem during the iteration have closed form solutions. We can regularize the minimization problem with the length of the interface boundaries, yet no partial differential equation need to be solven during the iterations. A narrow band technique is derived for the iterations. The algorithm is very fast. It is remarkable that the algorithm can find global minimizers for triple junction problems with large phase number M.

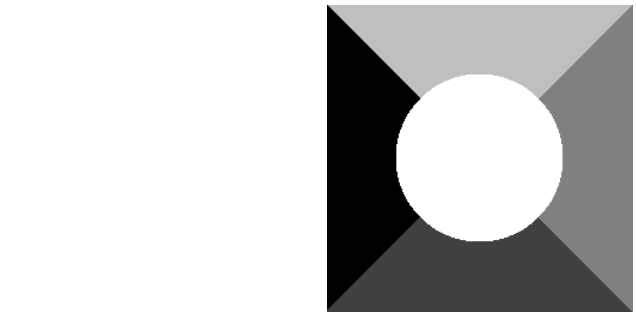
We mention that many of the level set based models can be easily extended to multiphase clustering and effectively implemented with the proposed framework. As future work, we would like to improve the denoising method in [48] with CVT framework. In addition, nonparametric segmentation with the proposed functional is also our future



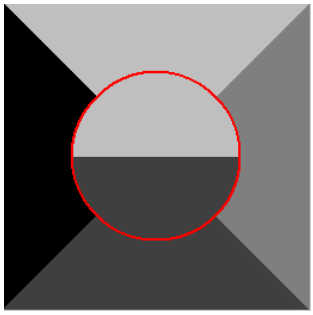
(a) 2-phases (b) 3-phases (c) 4-phases

(d) 5-phases (e) 6-phases (f) 10-phases

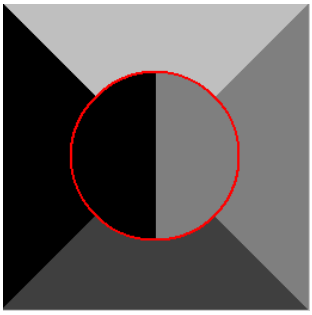
Figure 13: Multi-phase junctions obtained by our method.



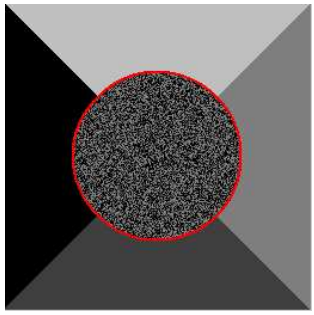
(a) original



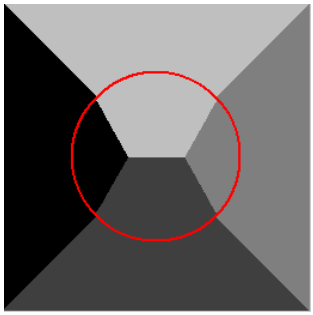
(b) initial labels (case 1)



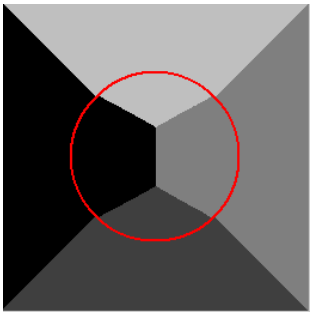
(c) initial labels (case 2)



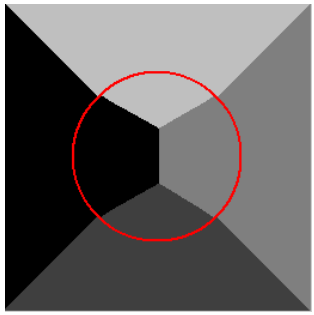
(d) initial labels (case 3)



(e) result 1



(f) result 2



(g) result 3

Figure 14: Complete the white circle: the solutions with different initial values. In this experiment, we let the parameter $\omega = 35$.

research content.

6. Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (No. 11071023), in part by the MOE (Ministry of Education) Tier II project T207N2202 and NRF 2007IDM-IDM 002-010.

A. The proof of proposition 1

For each x, let us denote $\bar{D}(x) = \min\{D_1(x), D_2(x), \cdots, D_M(x)\}$, then for all $\mathbf{u} \geqslant 0$ and $\sum_{k=1}^{M} u_k(x) = 1$

$$E(\mathbf{u}) = \int_\Omega \mathbf{D}(x) \cdot \mathbf{u}(x)\,\mathrm{d}x = \int_\Omega \sum_{k=1}^{M} D_k(x)u_k(x)\,\mathrm{d}x \geqslant \int_\Omega \bar{D}(x)\,\mathrm{d}x.$$

On the other hand, it is easy to check $\hat{\mathbf{u}}$ in a) or c) satisfies

$$E(\hat{\mathbf{u}}) = \int_\Omega \sum_{k=1}^{M} D_k(x)\hat{u}_k(x)\,\mathrm{d}x = \int_\Omega \bar{D}(x)\,\mathrm{d}x,$$

Thus $\hat{\mathbf{u}}$ is a minimizer of E, which implies a) and c) hold.
Moreover, suppose b) fails, that is to say there exists another minimizer $\underline{\mathbf{u}} \neq \hat{\mathbf{u}}$. Let us write $\tilde{\Omega} = \{x : \underline{\mathbf{u}}(x) \neq \hat{\mathbf{u}}(x)\}$, then

$$\begin{aligned} E(\underline{\mathbf{u}}) &= \int_\Omega \sum_{k=1}^{M} D_k(x)\underline{u}_k(x)\,\mathrm{d}x = \int_{\Omega-\tilde{\Omega}} \sum_{k=1}^{M} D_k(x)\hat{u}_k(x)\,\mathrm{d}x + \int_{\tilde{\Omega}} \sum_{k=1}^{M} D_k(x)\underline{u}_k(x)\,\mathrm{d}x \\ &> \int_{\Omega-\tilde{\Omega}} \bar{D}(x)\,\mathrm{d}x + \int_{\tilde{\Omega}} \bar{D}(x)\,\mathrm{d}x = \int_\Omega \bar{D}(x)\,\mathrm{d}x. \end{aligned}$$

It contradicts the fact that $\underline{\mathbf{u}}$ is a minimizer of E, which completes the proof. See aslo [44] for another proof for this proposition.

B. The proof of proposition 2

First, we have

$$\begin{aligned} \tilde{E}(\mathbf{u}^\nu, \mathbf{c}^\nu; \mathbf{u}^\nu) &\geqslant \tilde{E}(\mathbf{u}^{\nu+1}, \mathbf{c}^\nu; \mathbf{u}^\nu), \\ \tilde{E}(\mathbf{u}^{\nu+1}, \mathbf{c}^\nu; \mathbf{u}^{\nu+1}) &\geqslant \tilde{E}(\mathbf{u}^{\nu+1}, \mathbf{c}^{\nu+1}; \mathbf{u}^{\nu+1}), \end{aligned}$$

according to (20) and (21) respectively. In the next, we prove $\tilde{E}(\mathbf{u}^{\nu+1}, \mathbf{c}^{\nu}; \mathbf{u}^{\nu}) \geqslant \tilde{E}(\mathbf{u}^{\nu+1}, \mathbf{c}^{\nu}; \mathbf{u}^{\nu+1})$.

Let us recall

$$\tilde{E}(\mathbf{u}, \mathbf{c}; \mathbf{u}^{\nu}) = \int_{\Omega} \mathbf{D}(x, \mathbf{u}^{\nu}(x), \mathbf{c}) \cdot \mathbf{u}(x) \mathrm{d}x = \int_{\Omega} \sum_{k=1}^{M} \left\{ \left[d^2(f(x), c_k) + \lambda \int_{B(x;\omega)} g(y)(1 - u_k^{\nu}(y)) \, \mathrm{d}y \right] u_k(x) \right\} \mathrm{d}x.$$

Now, we consider the following minimizer of the energy

$$\tilde{E}(\mathbf{u}^{\nu+1}, \mathbf{c}^{\nu}; \mathbf{u}) = \int_{\Omega} \sum_{k=1}^{M} \left\{ \left[d^2(f(x), c_k^{\nu}) + \lambda \int_{B(x;\omega)} g(y)(1 - u_k(y)) \, \mathrm{d}y \right] u_k^{\nu+1}(x) \right\} \mathrm{d}x,$$

such that $u_k(x) \geqslant 0, \sum_{k=1}^{M} u_k(x) = 1$. Without loss of generality, we denote

$$\tilde{u}^{\nu+1}(x) = \max\{u_1^{\nu+1}(x), u_2^{\nu+1}(x), \cdots, u_M^{\nu+1}(x)\},$$

then we have $\forall \mathbf{u}$

$$\tilde{E}(\mathbf{u}^{\nu+1}, \mathbf{c}^{\nu}; \mathbf{u}) \geqslant \int_{\Omega} \sum_{k=1}^{M} \left\{ \left[d^2(f(x), c_k^{\nu}) + \lambda \int_{B(x;\omega)} g(y) \, \mathrm{d}y \right] u_k^{\nu+1}(x) \right\} \mathrm{d}x - \lambda \int_{\Omega} \tilde{u}^{\nu+1}(x) \int_{B(x,\omega)} g(y) \, \mathrm{d}y \, \mathrm{d}x,$$

please note $u_k^{\nu+1}(x)$ is binary and satisfies $\sum_{k=1}^{M} u_k^{\nu+1}(x) = 1$, thus it is not difficult to check $u^{\nu+1}$ is a minimizer of $\tilde{E}(\mathbf{u}^{\nu+1}, \mathbf{c}^{\nu}; \mathbf{u})$. Therefore one can get

$$\tilde{E}(\mathbf{u}^{\nu+1}, \mathbf{c}^{\nu}; \mathbf{u}^{\nu}) \geqslant \tilde{E}(\mathbf{u}^{\nu+1}, \mathbf{c}^{\nu}; \mathbf{u}^{\nu+1}).$$

In addition, we have

$$\begin{aligned} E(\mathbf{u}^{\nu}, \mathbf{c}^{\nu}) = &\quad \tilde{E}(\mathbf{u}^{\nu}, \mathbf{c}^{\nu}; \mathbf{u}^{\nu}), \\ E(\mathbf{u}^{\nu+1}, \mathbf{c}^{\nu+1}) = &\quad \tilde{E}(\mathbf{u}^{\nu+1}, \mathbf{c}^{\nu+1}; \mathbf{u}^{\nu+1}), \end{aligned}$$

and thus $E(\mathbf{u}^{\nu}, \mathbf{c}^{\nu}) \geqslant E(\mathbf{u}^{\nu+1}, \mathbf{c}^{\nu+1})$.

C. The proof of proposition 3

As before, we write

$$\tilde{E}(\mathbf{u}, \mathbf{c}^*; \mathbf{u}^*) = \int_{\Omega} \sum_{k=1}^{M} \left\{ \left[d^2(f(x), c_k^*) + \lambda \int_{B(x;\omega)} g(y)(1 - u_k^*(y)) \, \mathrm{d}y \right] u_k(x) \right\} \mathrm{d}x.$$

Let us consider the minimizer of $\tilde{E}(\mathbf{u}, \mathbf{c}^*; \mathbf{u}^*)$ with the constraint conditions $u_k(x) \geqslant 0, \sum_{k=1}^{M} u_k(x) = 1$. According to proposition (1), the binary $\hat{\mathbf{u}}$ with component functions

$$\hat{u}_k(x) = \begin{cases} 0 & k \in \mathbb{K}_x \\ 1 & k \notin \mathbb{K}_x \end{cases}$$

is a minimizer of $\tilde{E}(\mathbf{u}, \mathbf{c}^*; \mathbf{u}^*)$, and then one can get

$$\tilde{E}(\hat{\mathbf{u}}, \mathbf{c}^*; \mathbf{u}^*) \leqslant \tilde{E}(\mathbf{u}^*, \mathbf{c}^*; \mathbf{u}^*). \tag{24}$$

On the other hand, we may write

$$\tilde{E}(\hat{\mathbf{u}}, \mathbf{c}^*; \mathbf{u}) = \int_\Omega \sum_{k=1}^{M} \left\{ \left[d^2(f(x), c_k^*) + \lambda \int_{B(x;\omega)} g(y)(1 - u_k(y))\, \mathrm{d}y \right] \hat{u}_k(x) \right\} \mathrm{d}x.$$

As in the proof of proposition 2, one can get $\hat{\mathbf{u}}$ is a minimizer of $\tilde{E}(\hat{\mathbf{u}}, \mathbf{c}^*; \mathbf{u})$ with the constraint conditions, and thus

$$\tilde{E}(\hat{\mathbf{u}}, \mathbf{c}^*; \hat{\mathbf{u}}) \leqslant \tilde{E}(\hat{\mathbf{u}}, \mathbf{c}^*; \mathbf{u}^*). \tag{25}$$

Combining (24) and (25), we get

$$E(\hat{\mathbf{u}}, \mathbf{c}^*) = \tilde{E}(\hat{\mathbf{u}}, \mathbf{c}^*; \hat{\mathbf{u}}) \leqslant \tilde{E}(\mathbf{u}^*, \mathbf{c}^*; \mathbf{u}^*) = E(\mathbf{u}^*, \mathbf{c}^*).$$

Thus $E(\hat{\mathbf{u}}, \mathbf{c}^*)$ is a global minimization of (19).

References

[1] N. Paragios, R. Deriche, Geodesic active regions: a new framework to deal with frame partition problems in computer vision. J. Vis. Commun. Image R. 13 (2002) 249-268.

[2] C. Carson, S. Belongie, H. Greenspan, J. Malik, Blobworld: Image segmentation using Expectation-Maximization and its application to image querying, IEEE Trans. Pattern Anal. Machine Intell. 24 (1999) 1026-1038.

[3] D.L. Pham, J.L. Prince, An adaptive fuzzy c-means algorithm for image segmentation in the presence of intensity inhomogeneities, Pattern Recognit. Lett. 20 (1)(1999) 57-68.

[4] D.L. Pham, Spatial models for fuzzy clustering, Comput. Vis. Image Und. 84 (2001) 285-297.

[5] N. Joshi, M. Brady, Non-Parametric mixture model based evolution of level sets and application to medical images, Int. J. Comput. Vis. 88 (1)(2010) 52-68.

[6] I. Gath, A. B. Geva, Unsupervised optimal fuzzy clustering, IEEE Trans. Pattern Anal. Machine Intell. 11 (7)(1987) 773-781.

[7] C. Rother, V. Kolmogorov, A. Blake, GrabCut: Interactive foreground extraction using iterated graph cuts, ACM T. Graphic. 23 (2004) 309-314.

[8] L. Bertelli, S. Chandrasekaran, F. Gibou, B. S. Manjunath, On the length and area regularization for multiphase level set segmentation, Int. J. Comput. Vis. (2010) DOI: 10.1007/s11263-010-0348-4.

[9] E. Bae, X.C. Tai, Graph cut optimization for the piecewise constant level set method applied to multiphase image segmentation, in: Scale Space and Variational Mehotds in Computer Vision, (2009) 1-13.

[10] E. Bae, X.C. Tai, Efficient global minimization for the multiphase Chan-Vese model of image segmentation, EMMCVPR, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) , 5681 (2009) 28-41.

[11] X.C. Tai, Y. Duan, Domain decomposition methods with graph cuts algorithms for image segmentation. To appear in International J. Numer. Anal. Modelling.

[12] T. P. Gurholt, X.C. Tai, 3D multiphase piecewise constant level set method based on graph cut minimization. Numer. Math. Theor. Meth. Appl. 2 (2009) 403-420.

[13] J. Yuan, E. Bae, X.C. Tai. A study on continuous max-flow and min-cut approaches. In CVPR, USA, San Francisco, 2010.

[14] J. Yuan, E. Bae, X.C. Tai, Y. Boykov. A continuous max-flow approach to potts model. In ECCV, 2010.

[15] Q. Du, V. Faber, M. Gunzburger, Centroidal voronoi tessellations: applications and algorithms, SIAM Rev. 41 (4)(1999) 637-676.

[16] Q. Du, M. Gunzburger, L.L. Ju, Advances in studies and applications of centroidal voronoi tessellations, Numer. Math. Theor. Meth. Appl. 3 (2)(2010) 119-142.

[17] Q. Du, M. Gunzburger, L.L. Ju, X.Q. Wang, Centroidal voronoi tessellation algorithms for image compression, segmentation, and multichannel restoration, J. Math. Imaging. Vis. 24 (2)(2006) 177-194.

[18] J. Wang, L.L. Ju, X.Q. Wang, An edge-weighted centroidal voronoi tessellation model for image segmentation, IEEE Trans. Image Process. 18 (8)(2009) 1844-1858.

[19] D. Mumford, J. Shah, Optimal approximation by piecewise smooth functions and associated variational problems, Comm. Pure Appl. Math. 42 (1989) 577-685.

[20] T.F. Chan, L.A. Vese, Active contours without edges, IEEE Trans. Image Process. 10 (2)(2001) 266-277.

[21] J. Lie, M. Lysaker, X.C. Tai, A binary level set model and some applications to Mumford-Shah image segmentation, IEEE Trans. Image Process. 15 (5)(2006) 1171-1181.

[22] J. Lie, M. Lysaker, X.C. Tai, A variant of the level set method and applications to image segmentation. Math. Comp. 75 (255) (2006) 1155-1174.

[23] X.C. Tai, O. Christiansen, P. Lin, I. Skjaelaaen, Image segmentation using some piecewise constant level set methods with MBO type of project. Int. J. Comput. Vis., 73 (1)(2007) 61-76.

[24] Jung, Y.M. and Kang, S.H. and Shen, J., Multiphase image segmentation via Modica-Mortola phase transition, SIAM Journal on Applied Mathematics, vol. 67, no. 5, pp. 1213–1232, 2007.

[25] F. Li, M.K. Ng, T.Y. Zeng, C.L. Shen, A multiphase image segmentation method based on fuzzy region competition, SIAM J. Imaging Sciences 3 (3)(2010) 277-299.

[26] C. Zach and D. Gallup and J.-M. Frahm and M. Niethammer, Fast Global Labeling for Real-Time Stereo Using Multiple Plane Sweeps, in "Vision, Modeling and Visualization Workshop (VMV)", 2008.

[27] Lellmann, J. and Breitenreicher, D. and Schnörr, C., Fast and Exact Primal-Dual Iterations for Variational Problems in Computer Vision, European Conference on Computer Vision (ECCV),

2010.

[28] S. Marcelja, Mathematical description of the response of simple cortical cells, J. Optical Soc. Am. 70 (1980) 1297-1300.

[29] T. Brox, J. Weickert, B. Burgeth, P. Mrzek, Nonlinear structure tensors, Image Vision Comput. 24 (1)(2006) 41-55.

[30] S.D. Han, W.B. Tao, D.S. Wang, X.C. Tai, X.L. Wu, Image segmentation based on grabcut frame integrating multi-scale nonlinear structure tensor, IEEE Trans. Image Process. 18 (10)(2009) 2289-2302.

[31] C. Sagiv, N. Sochen, Y.Y. Zeevi, Integrated active contours for texture segmentation, IEEE Trans. Image Process. 15 (6)(2006) 1633-1646.

[32] N. Sochen, R. Kimmel, R. Malladi, A general framework for low level vision, IEEE Trans. Image Process. 7 (3)(1998) 310-318.

[33] N. Houhou, J.P. Thiran, X. Bresson, Fast texture segmentation based on semi-local region descriptor and active contour, Numer. Math. Theor. Meth. Appl. 2 (4)(2009) 445-468.

[34] A. Buades, B. Coll, J.M. Morel, A review of image denoising algorithms, with a new one, Multiscale Model. Sim. 4 (2)(2005) 490-530.

[35] L.A. Vese, T.F. Chan, A multiphase level set framework for image segmentation using the mumford and shah model. Int. J. Comput. Vis. 50 (3)(2002) 271-293.

[36] C.M. Li, C.Y. Kao, J.C. Gore, Z.H. Ding, Implicit active contours driven by local binary fitting energy, in:Proceedings of the CVPR'07. (2007) 1-7.

[37] C.M Li, C.Y Kao, J. C. Gore, Z.H Ding, Minimization of region-scalable fitting energy for image segmentation, IEEE Trans. Image Process. 17 (10)(2008) 1940-1949.

[38] X.F. Wang, D.S. Huang, H. Xu, An efficient local Chan-Vese model for image segmentation, Pattern. Recogn. 43 (3)(2010) 603-618.

[39] Z. Karel, Contrast limited adaptive histograph equalization, Graphic Gems IV, Academic Press Professional, Inc., San Diego, CA, USA, (1994) 474-485.

[40] X. Bresson, T.F. Chan, Non-local unsupervised variational image segmentation models, CAM Report 07-23, (2007).

[41] T.F. Chan, S. Esedoglu, M. Nikolova, Algorithms for finding global minimizers of image segmentation and denoising models, SIAM J. Appl. Math. 66 (5)(2006) 1632-1648.

[42] X. Bresson, S. Esedoglu, P. Vandergheynst, J.P. Thiran, S. Osher, Fast global minimization of the active contour/snake model, J. Math. Imaging. Vis. 28 (2)(2007) 151-167.

[43] T. Goldstein, X. Bresson, S. Osher, Geometric applications of the split Bregman method: segmentation and surface reconstruction, UCLA CAM Report 09-06, (2009).

[44] E. Bae, J. Yuan, X.C. Tai, Global minimization for continuous multiphase partitioning problems using a dual approach, UCLA CAM Report 09-75,(2009). To appear in Int. J. Comput. Vis. (online 2010).

[45] X.C. Tai, C.L. Wu, Augmented Lagrangian method, dual methods and split Bregman iteration for ROF model, UCLA CAM Report 09-05, (2009).

[46] H.W. Li, X.C. Tai, Piecewise constant level set method for multiphase motion, Int. J. Numer. Anal. Mod. 4 (2)(2007) 291-305.

[47] T. Pock, A. Chambolle, D. Cremers, H. Bischof, A convex relaxation approach for computing minimal partitions, IEEE CVPR2009, 810-817.

[48] J. Liu, H.Y. Huang, Z.D. Huan, H.L. Zhang, Adaptive variational method for restoring color images with high density impulse noise, Int. J. Comput. Vis. 90 (2)(2010) 131-149.