

A Fast Segmentation Method Based on Centroidal Voronoi Tessellation and Its Applications: Intensity Inhomogeneity and Texture Segmentation

Jun Liu, Xuecheng Tai, Haiyang Huang, Zhongdan Huan

Abstract

In this paper, we proposed a new constrained optimization energy for image segmentation based on centroidal Voronoi tessellation (CVT). We show the characteristic functions of the Voronoi regions are the exact minimizers (may not unique) of the proposed energy at each iteration. Generally speaking, the CVT-based segmentation is always faster than the popularly used level set method since it don't need to solve a PDE 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 via incorporating the image global and local information into the energy. Compared with other methods, the experimental results in this paper have shown that our approach greatly improves the calculation efficiency without losing segmentation accuracy.

Key words: Image Segmentation, 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 the image segmentation is to find a partition of an image into its constituent parts.

There are many methods were proposed for image segmentation in the past several decades. These approaches can be classified in terms of different criterions. For example, region based and edge based segmentation(e.g.[1, 15, 12]), soft-threshold and hard-threshold segmentation (e.g.[2, 3, 4, 12, 15]), parametric and nonparametric (e.g. [12, 15, 5]) segmentation, supervised and unsupervised segmentation (e.g. [6, 7]), etc..

Level set method can be regraded as 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 [14] had been successfully extended to a wide range of occasions. One of the simplified variant of Mumford-Shah model is Chan-Vese model [15], which is well suited for segmenting an image with nearly constant intensity. In order to hand n phases segmentation, in [23, 8] the authors use $\log_2(n)$ level set functions to represent the n regions. However, such many unknown level set functions in the model would affect the efficiency of the implementation. To overcome this flaw, in [16], a piecewise constant level set method was proposed to deal with multi-class classification by representing any number of phases with one single label function, and their approximate model can be implemented by graph cut (see [9]). Still, to deal with n clusters, it needs to add the 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.

Centroidal Voronoi Tessellation (CVT) for image segmentation was proposed in [12]. In its simplest form, the basic CVT algorithm is known as the k-means clustering, which is widely used in many applications due to its high computational efficiency. 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 [14], Chan-Vese model [15], piecewise constant level set method [16], and

so on. All of these methods are based on the fact that the total variation (TV) of characteristic function on a region is equal to the length of the region boundary, but the TV regularization term in the cost functional makes this minimization problem time-consuming. In [13], the authors 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 TV. Compared with the level set method, the CVT based segmentation has certain benefits, 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 includes the piecewise smooth Mumford-Shah model [23] and local binary fitting (LBF) method [24, 25]. In these models, the active contour is driven by local information of the image and they can produce some impressive results. Another method is to add the local information to the segmentation functional by estimating the brightness of the background, such as [3, 26]. There are many unknown variables in these methods and finding them is a very time-consuming process.

In this paper, we propose a method for image segmentation by minimizing the segmentation energy with some constrain conditions. It is developed by the fact that this constraint optimization problem has an explicit solver 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. We show the smoothness term in the energy is invalid in the inner of a connected cluster region during the iteration, thus we only need to update the label function to obtain a fine result at the boundaries of a given initial clusters. This means that it can speed up the computational process and we propose a algorithm to accelerate the implementation. We extend the CVT based image segmentation in the presence of intensity inhomogeneities by combining local histogram equalization. In addition, as an application of the proposed method, we give a texture segmentation model based on CVT. Experimental results and comparisons with other methods have shown the high efficiency of the proposed approach.

The rest of the paper is organized as following: in section 2, the CVT and EWCVT for image segmentation is briefly introduced; The proposed energy, algorithm, 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 $\mathbb{L} = \{1, 2, \dots, M\}$. 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. Now, let us recall some concepts of centroidal Voronoi tessellation [10, 11, 12] 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

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

where $k = 1, 2, \dots, M$, 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

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

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) [10] 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 [10, 12], the authors define a energy for Voronoi tessellation, and they show that the CVT can be constructed by finding a minimizer of the following energy (Prop. 3.1, [10])

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

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 [12], an iteration algorithm (Lloyds Method) is employed to solve the minimization problem for image segmentation:

Algorithm 1 (CVT clustering, [12]). *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 [13] 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 [13], the authors defined a discrete energy for image segmentation which is robust to noise. We can rewrite this clustering energy with the continuous form as follows:

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

Here $\lambda > 0$ is a regularization parameter which 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 . $\chi : \Omega \times \Omega \mapsto \{0, 1\}$ is a characteristic function which has an expression

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

It has been shown in [13] 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

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

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

Then similar to [10, 12], an edge-weighted Voronoi region can be given as following

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

The EWCVT algorithm can be described as follows (more details about this method, please refer to [13]):

Algorithm 2 (EWCVT clustering, [13]). *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 under noise since there is a smooth constraint on Voronoi region Ω_k in E_2 . However, the EWCVT do not consider the image edges, and the edges in the image usually can provide some useful information for segmentation, we will show this flaw in the later experiments.

3. The Proposed Method

3.1. A constrained optimization for CVT and EWCVT

In this section, we give a constrained optimization energy for CVT-based image segmentation. In accordance with this new representation, the CVT-based image segmentation can be more easily generalized.

Let us begin with the following proposition:

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

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

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 $\hat{\mathbf{u}}$ of $E(\mathbf{u})$. Here, $|\cdot|$ is the cardinality of the set \mathbb{K}_x .

c) The binary function $\hat{\mathbf{u}} = (\hat{u}_1, \hat{u}_2, \dots, \hat{u}_M)$ with component function

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

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

Proof:

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

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

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

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

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

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

It contradicts the fact that $\underline{\mathbf{u}}$ is a minimizer of E , which completes the proof.

Now, let us introduce a vector-valued characteristic function $\mathbf{u} = (u_1, u_2, \dots, u_M)^T$ 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 rewrote as follows:

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

In the last equation, $\mathbf{D} = (D_1, D_2, \dots, D_M)^T$ with the k -th component function

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

Considering the following constrained optimization problem

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

We note the function $D(x)$ in here is different from the one in the proposition 1 since it is depended on the unknown function \mathbf{u} , thus the proposition 1 can not directly solve this minimization problem. Now, let us construct an iteration scheme to solve this problem: starting from an initial guess $\mathbf{u}^0, \mathbf{c}^0$; we compute a series of minimizers

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

such that

$$\mathbf{u}^{\nu+1} = \arg \min_{\mathbf{u}} E_2(\mathbf{u}, \mathbf{c}^{\nu}; \mathbf{u}^{\nu}), \quad \text{s.t.} \quad \sum_{k=1}^M u_k(x) = 1, u_k(x) \geq 0, \tag{8}$$

$$\mathbf{c}^{\nu+1} = \arg \min_{\mathbf{c}} E_2(\mathbf{u}^{\nu+1}, \mathbf{c}; \mathbf{u}^{\nu+1}), \tag{9}$$

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

Here

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

and ν is the iteration number. Now, please note the proposition 1 can be applied to the first minimization problem since \mathbf{D} is independent of the unknown function \mathbf{u} for any given $\mathbf{u}^\nu, \mathbf{c}^\nu$. Thus according to proposition 1, one can get $\mathbf{u}^{\nu+1}$ with the component function

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

is a solver for the first problem. Here

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

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

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

Now, comparing (11), (12) with (4) and (5), clearly, u_k in equation (11) actually is a characteristic function of Voronoi region Ω_k determined by (5) since the factor 2λ in (4) and λ in (12) can be regarded as two different regularization parameters. Therefore, minimizing (6) with (8), (9) is equivalent to EWCVT clustering algorithm 2. In the section 3.4, we will show the iteration scheme (8), (9) satisfies

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

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, although the spatial smoothness of the clusters is considered in the EWCVT, it still lacks image edge information and always fails to find some details of the object in the image. We introduce the following energy which combines the image intensity, the spatial smoothness of each cluster, and the image edge information together:

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

Here g is an edge detection function and is often taken as $g = \frac{1}{1+|\nabla f|}$ in many literatures. 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 objects have more accurate edges. For convenience, we refer this model as IEWCVT hereafter.

3.2.2. The intensity inhomogeneous image segmentation with CVT

Intensity inhomogeneity always appears in nature images such as MR images. It is more difficult to extract the objects accurately from the images with intensity inhomogeneity since an object in such images may not share the similar intensity. However it can 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. [23, 24, 25, 26]) to handle the intensity inhomogeneity problem, and their models are mainly based on the level set method and can provide an impressive result. However, the level set based approaches are time-consuming because they always need to solve a PDE during each outer iteration. As reported in [24, 25, 26], dividing an image with small size such as 100×100 into two parts would always cost several seconds or more. We will propose a very fast approach to address this issue in this section.

In fact, the brightness of an object in an image with intensity inhomogeneity is varied. The contrast between the target and the background in the bright areas is always greater than that in the dark areas, thus the intensity inhomogeneity problem can be handled by enhancing the local contrast of the images. An efficient tool for sharpening the contrast is the histogram equalization. Contrast-limited adaptive histogram equalization (CLAHE) [27] 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 to preprocess the images.

Let us denote \tilde{f} is the output of CLAHE. Figure1 shows an output of CLAHE. As can be seen from this figure, the contrast of the original image in the top half has been improved through CLAHE, however, the brightness of the circle in the bottom is still higher than that in the top. Generally speaking, segmenting \tilde{f} directly would still produce undesirable result (please see 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 the preprocessed image h . A result for segmenting h is illustrated in Figure1(f). From this figure, we can find that the semi-circular with low contrast in the original image can be well segmented, but the other half fails.

Based on the above observing, we propose the following energy E_4 to handle the intensity inhomogeneity segmentation problem.

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

where β is a parameter which controls the balance between f and h . In order to unify the scales, 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). We will give more experimental results and comparisons with other methods in section 4 to show the high computational efficiency of the proposed method.

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 [17], structure tensor [18], Beltrami framework [19, 20, 21] and so on. For simpleness and high efficiency, we choose the semi-local region texture descriptor which is proposed in [21] 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 $x = (x_1, x_2)^T$.) $\mathcal{S} : (x_1, x_2) \mapsto (x_1, x_2, \mathcal{P}_x(\mathbf{f}))^T$, where \mathcal{P}_x is a $(2r+1) \times (2r+1)$ square neighborhood centered at $x = (x_1, x_2)$,

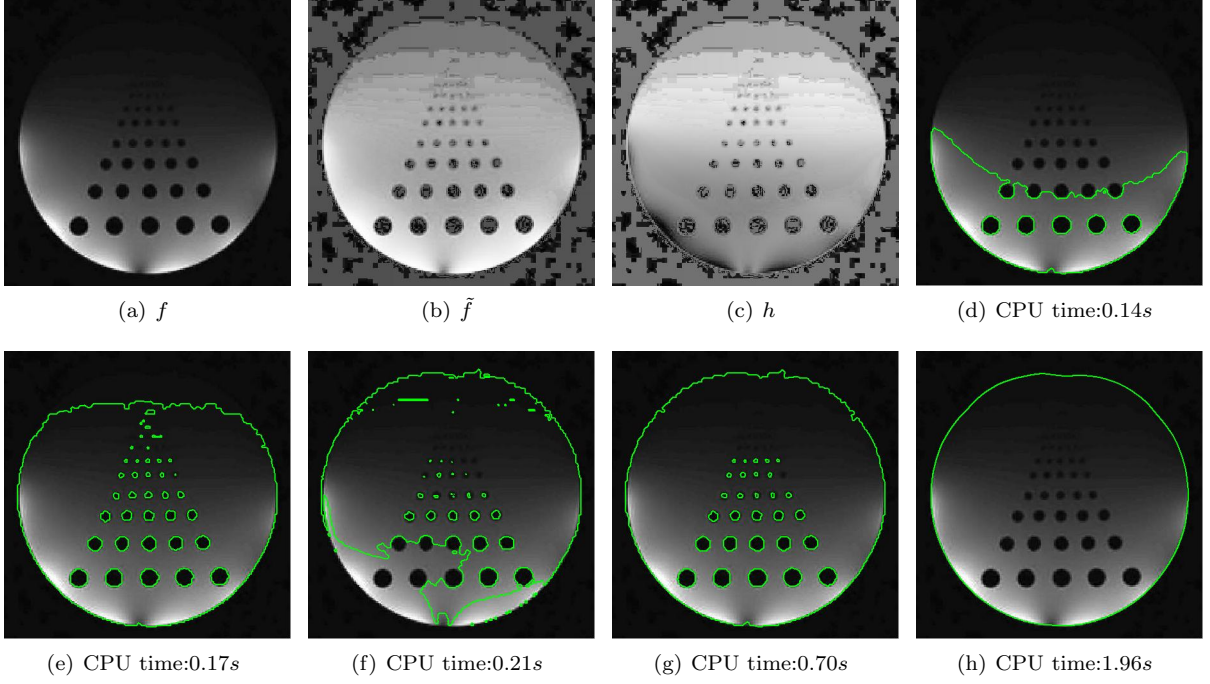


Figure 1: The effects of CLAHE and different segmentation results with IEWCVT (image size: 447×447 , 2 clusters). (a) the original image f ; (b) the output of CLAHE, i.e. $\tilde{f} = \text{adapthisteq}(f, \text{'NumTiles'}, [2 \ 2], \text{'ClipLimit'}, 0.2)$; (c) $h = \tilde{f} - f$; (d)-(f) segmentation 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), both of the two case, $\beta = 1, \lambda = 1.5$.

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

$$I = \begin{pmatrix} dx_1 \\ dx_2 \end{pmatrix} \begin{pmatrix} E & F \\ F & G \end{pmatrix} \begin{pmatrix} dx_1 & dx_2 \end{pmatrix},$$

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

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

The texture descriptor T in [21] is defined as

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

Here we choose

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

as the texture descriptor. Clearly, if $r = 0$, then $EG - F^2 = 1 + |\nabla f|^2$ and T is the often used edge detector function $g = \frac{1}{1 + |\nabla f|^2}$. 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 borders. 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. [22]),

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

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 $\int W(x, y)dy = 1$.

Then, the nonlocal edge detector function can be defined as

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

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

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

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

3.2.4. Block-Based CVT Image Segmentation

The previously discussed segmentation methods are point-based and the energy can be naturally extended to block-based one:

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

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 $\bar{f}(x) = \int_{\Omega} k(x - y) f(y) dy$, please note $\int k(y) dy = 1$, then

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

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

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

Please note \bar{f} is a denoised image with smooth kernel k , thus minimizing the energy (17) is equivalent to segment the denoised image, which is more robust to noise than the standard CVT clustering.

Another interesting thing for block-based segmentation is that when the center c_k is a function, namely

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

then it can produce a segmentation result which can connect the broken fringes in a regular fringe image. Please see Figure2 for the experimental results.

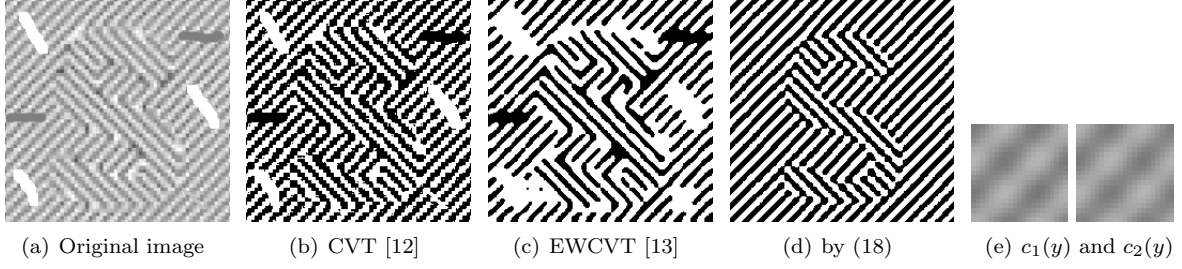


Figure 2: A comparison of the results obtained by CVT, EWCVT, and minimizing the energy (18) (2 clusters). (a) the original image; (b) segmentation result with CVT; (c) segmentation result with EWCVT, $\lambda = 0.02, \omega = 5$; (d) segmentation result by minimizing (18), $k = \frac{1}{|B(\omega)|}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

3.3. A Fast Narrow Banding Algorithm

In this section, we give a fast algorithm for minimizing the proposed energy.

Considering the following constraint optimization problems

$$\begin{aligned} (\mathbf{u}^*, \mathbf{c}^*) &= \arg \min_{\mathbf{u}, \mathbf{c}} \{E_i(\mathbf{u}, \mathbf{c}) = \int_{\Omega} \mathbf{D}(x, \mathbf{u}, \mathbf{c}(x)) \cdot \mathbf{u}(x) dx\} \quad \text{s.t.} \quad \sum_{k=1}^M u_k = 1, u_k \geq 0, \\ i &= 1, 2, \dots, 6, \end{aligned}$$

as the previous discussion, we can obtain a iteration scheme

$$\mathbf{u}^{\nu+1} = \arg \min_{\mathbf{u}} E_i(\mathbf{u}, \mathbf{c}^{\nu}; \mathbf{u}^{\nu}), \quad \text{s.t.} \quad \sum_{k=1}^M u_k(x) = 1, u_k(x) \geq 0, \quad (19)$$

$$\mathbf{c}^{\nu+1} = \arg \min_{\mathbf{c}} E_i(\mathbf{u}^{\nu+1}, \mathbf{c}; \mathbf{u}^{\nu+1}), \quad (20)$$

$$\nu = 0, 1, 2, \dots, \quad (21)$$

where

$$E_i(\mathbf{u}, \mathbf{c}; \mathbf{u}^{\nu}) = \int_{\Omega} \mathbf{D}(x, \mathbf{u}^{\nu}, \mathbf{c}(x)) \cdot \mathbf{u}(x) dx.$$

then applying the proposition 1, we obtain an uniform algorithm for all the models:

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 several standard CVT iteration:

Step 1, update l according c_k . $\forall k = 1, 2, \dots, M$, compute $D_k(x)$ for each model, 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 let $l(x) = k_1$. Mathematically, $l(x) = \min\{k_1 : k_1 \in \mathbb{K}_x\}$, where $\mathbb{K}_x = \{k : D_k(x) \leq D_i(x), \forall i \in \mathbb{L}\}$.

Step 2, update c_k according l and equation (20).

Step 3 if convergence criterion is reached, end the algorithm; else, go to step 1.

Let us do some further analysis for this algorithm. When a pixel $f(x)$ 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 $d^2(f(x), c_j) \leq d^2(f(x), c_i), i \in \mathbb{L} \setminus \{j\}$. 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 all the regularization parameter $\lambda > 0$, thus according algorithm 3, $f(x)$ will be still assigned to the j -th cluster no matter what the value of the parameter λ is. This fact means that

we do not need to calculate D_k and reassign the label function l for such pixels, therefore the algorithm can be efficiently implemented in a narrow band of the clusters boundaries.

We use the following method to quickly find the narrow band computational domain: let $\tilde{l}(x) = |(G_\sigma * l)(x) - l(x)|$, where G_σ 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. In this paper, unless otherwise specified, we set the parameter $\sigma = 3.0, \varepsilon = 0.01$.

In conclusion, we get a fast narrow banding algorithm for all the proposed models as follows:

Algorithm 4 (Narrow Banding 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 several standard 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 let $l(x) = \min\{k_1 : k_1 \in \mathbb{K}_x\}$, where $\mathbb{K}_x = \{k : D_k(x) \leq D_i(x), \forall i \in \mathbb{L}\}$, set $E_{new} = E_{new} + D_{l(x)}(x)$; else, nothing is done.

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.

In practice, this algorithm is easily coded and very fast. We will show its high efficiency in the experimental section.

3.4. Some Analytical Results for the Algorithm and Model

We have the following proposition on the algorithm and model.

Proposition 2 (Descent of the energy). *The sequence $(\mathbf{u}^\nu, \mathbf{c}^\nu)$ generated by (19), (20) satisfies $E_i(\mathbf{u}^\nu, \mathbf{c}^\nu; \mathbf{u}^\nu) \geq E_i(\mathbf{u}^{\nu+1}, \mathbf{c}^{\nu+1}; \mathbf{u}^{\nu+1})$, where $\nu = 0, 1, 2, \dots$.*

Proof:

Proposition 3 (The existence of the binary solution). *if $E_i(\mathbf{u}^*, \mathbf{c}^*)$ is the global minimization of $E_i(\mathbf{u}, \mathbf{c})$, then there must be a binary $\hat{\mathbf{u}}$ such that $E_i(\hat{\mathbf{u}}, \mathbf{c}^*)$ is also a global minimization of $E_i(\mathbf{u}, \mathbf{c})$.*

Proof:

4. Experimental Results

In this section, we propose four experiments to illustrate the high efficiency of the proposed models and algorithms. The implementation of all the algorithms described in this paper is made in Matlab (mex files) on a laptop with Intel 2.40G CPU.

In all the following 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. Experiment I: Comparison between EWCVT and IEWCVT

The main difference between EWCVT and IEWCVT (13) is the regularization term. Thus we pay more attention to observe what will happen when increasing the smoothness 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 $[0, 1]$.

Figure3 shows a comparison of segmentation results with EWCVT and IEWCVT. In this experiment, the synthetic image will be separated 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 in the boundaries, especially in Figure3(b). Naturally, one can expect to 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

Table 1: The CPU time in experiment I (Figure3).

Models	ω		
	3	9	15
EWCVT (Alg. 3)	0.22s	1.64s	13.18s
IEWCVT (Alg. 4)	0.25s	0.66s	1.95s

the first case, there is a bifurcation appeared in the center of Figure3(c), even although 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 (please see Figure3(d)). On the other hand, the IEWCVT can produce the precise clusters by increasing ω . In the next paragraph, we will give a theoretical interpretation about this phenomenon caused by EWCVT in the experiment.

First of all, the regularization term $E_L = \sum_{k=1}^M \int_{\Omega} u_k(x) \int_{B(x;\omega)} \sum_{j=1, j \neq k}^M u_j(y) dy dx \propto \text{length}(\Gamma)$, where Γ

is a boundary curve of the clusters (more details about this fact, please refer [13].), and 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 proof the fact:

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

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

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

Please note the angles formed by boundaries of the clusters in Figure3(c) and Figure3(d) are all nearly 120° , and the above conclusion is also confirmed by this experiment.

In addition, we list the respective CPU time for EWCVT and IEWCVT in the Table1. Clearly, the computational complexity of EWCVT is lower than IEWCVT's since the former does not need to calculate the edge detector function g . In order to illustrate the efficiency of the algorithm 4, we use algorithm 3 to solve EWCVT and algorithm 4 to solve IEWCVT. From this table, we can see algorithm 4 is much faster than algorithm 3, especially when the radius ω is large.

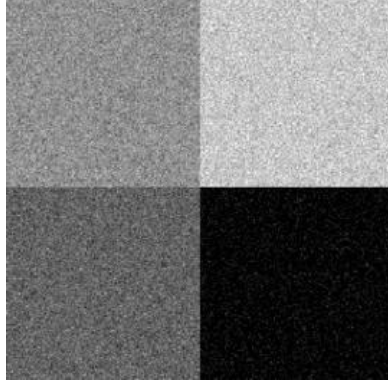
4.2. Experiment II: image segmentation with intensity inhomogeneous and comparison with other methods

In this experiment, we test the model described in section 3.2.2 and make some comparisons with other approaches. In [24, 25], 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.

There are several parameters in the proposed method. Let us first discuss how to choose them. Two of the most important parameters in the Contrast-limited adaptive histogram equalization (Matlab function "adapthisteq") to control the contrast enhancement 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 information about how to choose these parameters, one can refer to the help documents of matlab. The parameter ω is determined by noise and the scale of the objects which would be eliminated, and it should always be larger than the radius of the undesired targets. β and λ are two parameters to control the weights of cluster difference and smoothness, respectively. In most cases, we choose $\beta = 1.0, \lambda = 1.0$. In addition, we give all the parameters values which is used in this experiment in Table 2.

In Figure5, we display the outcomes obtained by EWCVT [13], LBF [25], and our method for segmenting images with intensity inhomogeneous into 2 clusters. Obviously, EWCVT can not provide the desired results in this case. Both of LBF in [25] and our method can produce similarly good results, however, looking at Table3, which records the CPU time for each method, one may find our method is much faster than LBF.

Unlike the popular level set based method which is not easily generalized to deal with the cases with more than two clusters (such as LBF [24, 25]), another superiority of CVT-based approach is that it can well



(a) Synthetic image, size 268×269 .



(b) EWCVT [13], $\omega = 3$



(c) EWCVT [13], $\omega = 9$



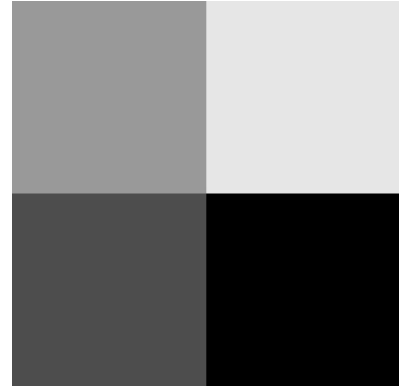
(d) EWCVT [13], $\omega = 15$



(e) IEWCVT, $\omega = 3$



(f) IEWCVT, $\omega = 9$



(g) IEWCVT, $\omega = 15$

Figure 3: Comparison between EWCVT [13] and the proposed IEWCVT. (a) Original image; (b)-(d) the clustering results by EWCVT; (e)-(g) the clustering results by IEWCVT.

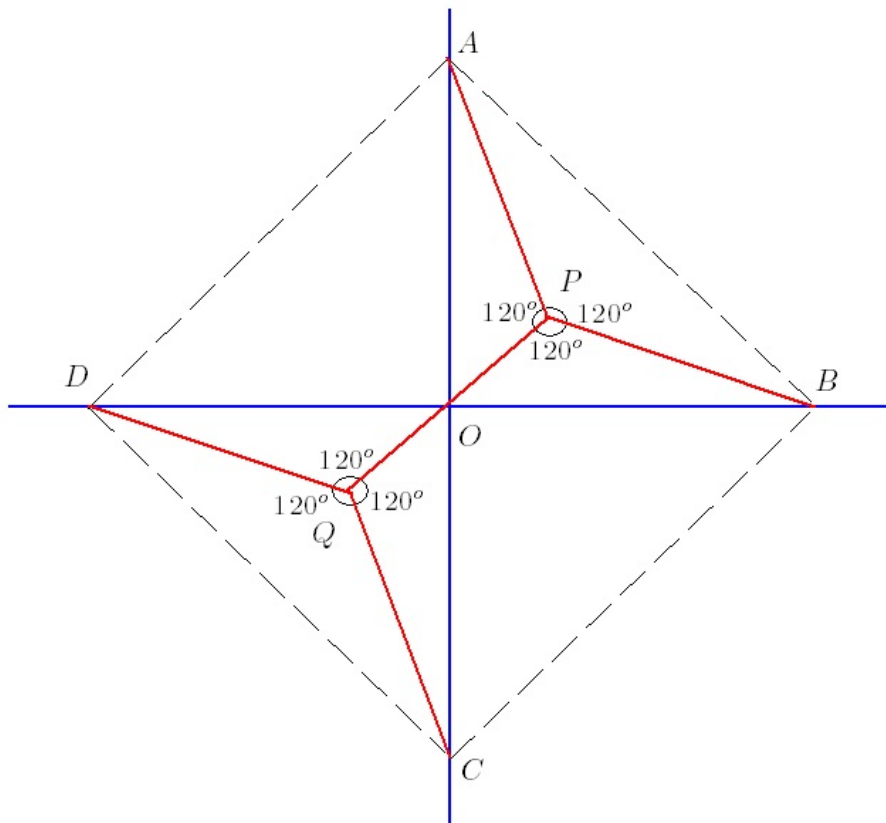


Figure 4: A diagram for explaining the results in Figure3(c) and Figure3(d)

Table 2: The parameters values used in experiment II (Figure5 and Figure6).

Figure	CLAHE		β	Regularization	
	“NumTiles”	“ClipLimit”		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

Table 3: A comparison of the the CPU time in experiment II (Figure5 and Figure6).

Figure	image size	EWCVT[13]	proposed			LBF[25]
			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	-

handle multi-clusters situations. In Figure6, we give a 3-clusters result for brain MR image with intensity inhomogeneous. As before, the CPU time for this segmentation are listed in Table3.

4.3. Experiment III: texture image segmentation

We mention the nonlocal edge detector function (16) is slightly better than the often used one when the images have numbers of repeatable structures and contaminated by noise, however, it is time-consuming to calculate the nonlocal weighting function. For the comparison between the nonlocal regularization and local regularization for image segmentation, please refer to [28]. In this experiment, we still use the conventional edge function in order to improve the computational efficiency.

Recent years, an efficient segmentation method named globally convex segmentation (GCS) is proposed in [29, 30]. In their models, the regularization term is a total variation norm, which is different from the one employed in this paper. The GCS model can be implemented by many algorithms such as [31, 32]. In [21], the authors extended the GCS framework to texture segmentation. We make some comparisons between our algorithm and [21].

Figure7 illustrates some results for texture segmentation. Comparing with the results which provided by ours and [21]’s carefully, one can find the segmentation results produced by the method in [21] are short of accuracy in the borders (See Figure8 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 adding the edge information of the original image into the energy and partly avoiding this phenomenon.

Finally, the values of parameters in our model and the CPU time are all displayed in Table4. We see again the CPU time of the proposed segmentation method is much less than [21]’s. Let us mention the segmentation model in [21] is based on probability density function, and it is often more time-consuming than Chan-Vese model [15], so we will make a comparison between split Bregman iteration for Chan-Vese model [31] and our algorithm in the next experiment.

4.4. A comparison between Chan-Vese model with split Bregman method and the proposed algorithm 4

Split Bregman methods is an efficient and fast algorithm for TV minimization. Let us mention again the regularization term between the Chan-Vese model [15] and ours is different. Though the TV term $\int_{\Omega} |\nabla u| dx$ can be rewrote formality as $\int_{\Omega} \nabla \cdot \left(\frac{\nabla u}{|\nabla u|} \right) u dx \triangleq \int_{\Omega} D \cdot u dx$ according to the divergence theorem, here D is still depended on u and it does not satisfy all the conditions of proposition 1, thus the proposed algorithm can not directly implement the TV-based segmentation models.

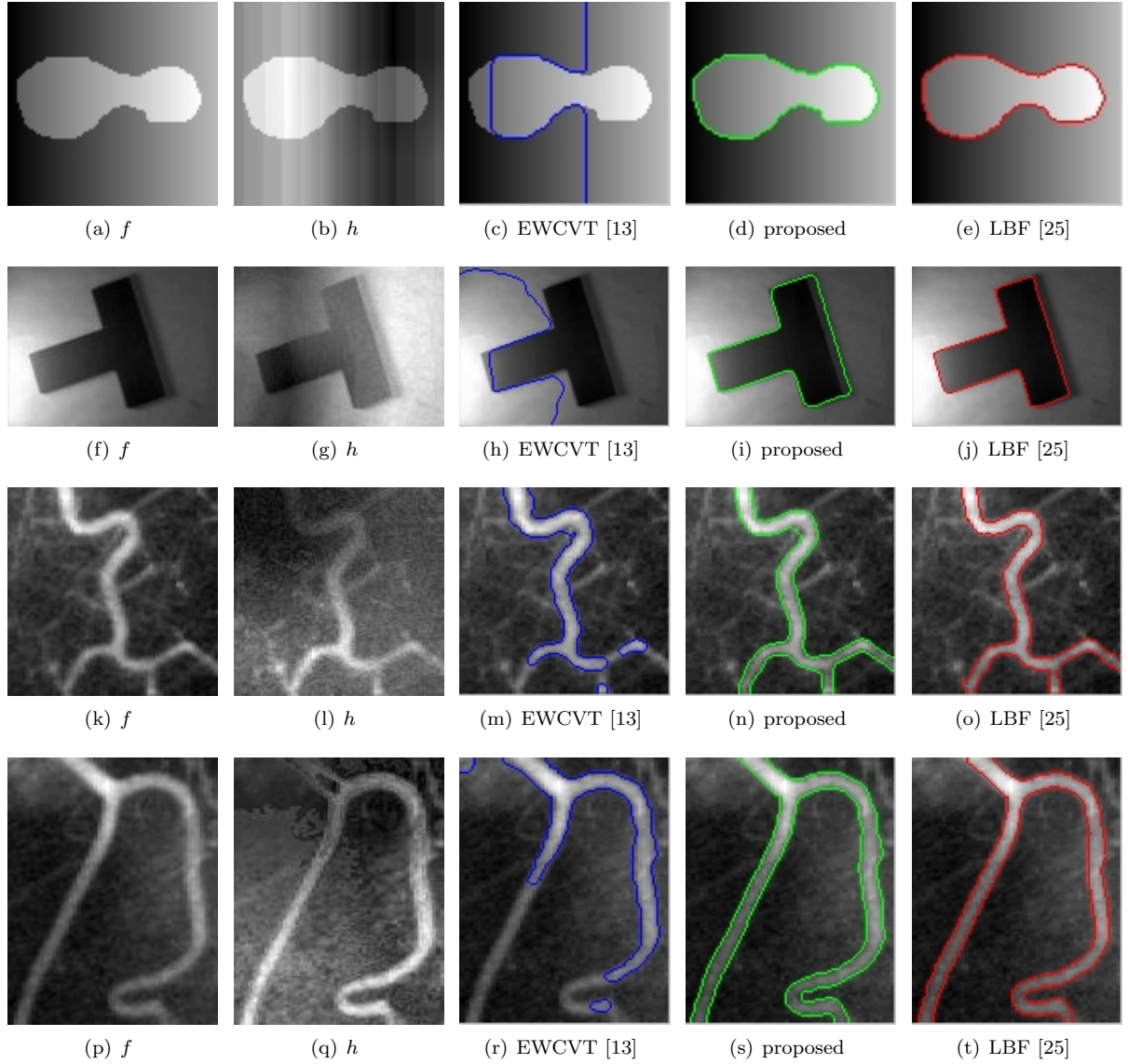


Figure 5: The results of intensity inhomogeneous image segmentation (2 clusters) obtained the proposed model (14) and other connected methods. Form left to right: first column, the original images f ; second column, the preprocessed images h ; third column, the results with EWCVT [13]; fourth column, the results obtained by the proposed method in section 3.2.2; fifth column, the results by LBF [25].

Table 4: The CPU time and parameters values in experiment III (Figure7, image size 321×481).

Figure	parameters			CPU time		
	r	ω	λ	calculate T	proposed	[21]
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

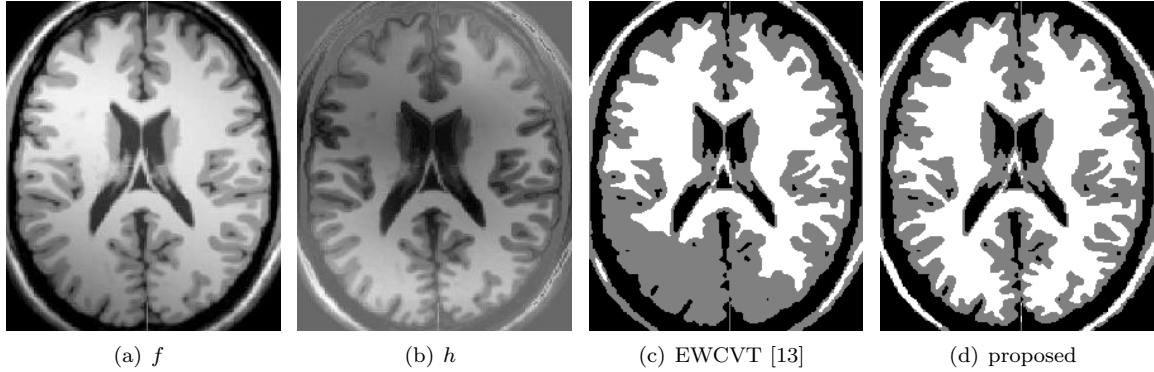


Figure 6: An example of 3-clusters segmentation with intensity inhomogeneous. (a), the original image; (b) the preprocessed image h ; (c) clusters obtained by EWCVT; (d) clusters provided by the proposed method.

Table 5: Comparisons between Alg. 4 and split Bregman iteration [31] (Figure9).

Fig. No. (from top to bottom)	image size	parameters		CPU time	
		ω	λ	Alg.4	[31]
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

All the comparisons between algorithm 4 and split Bregman method for GCS with Chan-Vese model [31], including the segmentation results and CPU time, are displayed in Figure9 and Table5.

Comparing with other methods, as it is seen in these figures and tables, our algorithm is highly competitive.

5. Conclusion

In this paper, we proposed a new method to segment images with intensity inhomogeneous and texture. These approaches are all based on minimizing a constrained optimization energy. With this new energy, we extended the CVT-base image segmentation: we added the edge information to the regularization term for avoiding to destroy the image edges; to deal with the intensity inhomogeneous problem, the energy is modified by fusing the original image and the enhanced image with Contrast-limited adaptive histogram equalization (CLAHE) together; For texture segmentation, we employed the texture descriptor introduced in [21] to describe the texture feature and integrated it into the CVT framework; The block-based energy is proposed and then we illustrated it is actually equivalent to segment the denoising images. All The proposed models can be implemented by an unified scheme. In addition, we introduced a narrow banding algorithm to accelerate the calculation. Compared with the often used level set method, experimental results had shown that our approach is much faster than others such as the level set method without loss of the impressive results.

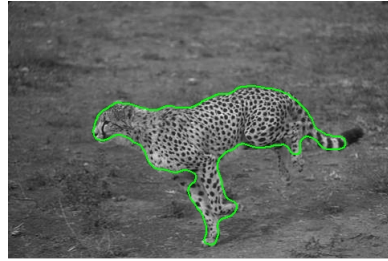
We mention that many of the level set based models can be easily extended to multiphase clustering and effectively implemented with the proposed framework. As future work, we would like to improve the denoising method in [33] with CVT framework. In addition, nonparametric segmentation with the proposed functional is also our future research content.

References

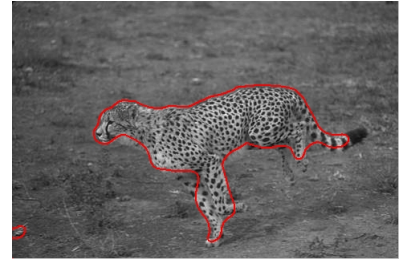
- [1] N. Paragios, R. Deriche, Geodesic active regions: a new framework to deal with frame partition problems in computer vision. J. Vis. Commun. Image R. 13 (2002) 249-268.



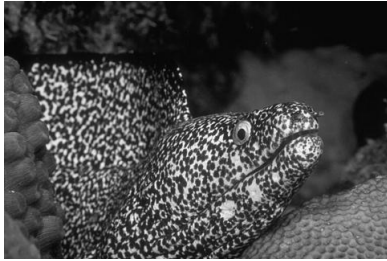
(a) f



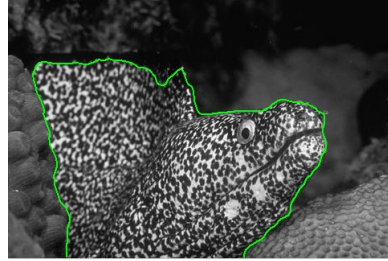
(b) proposed



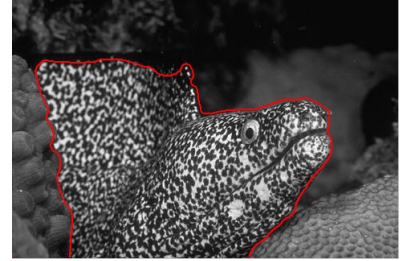
(c) [21]



(d) f



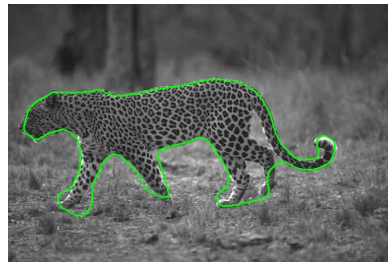
(e) proposed



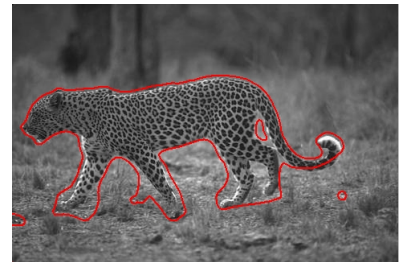
(f) [21]



(g) f



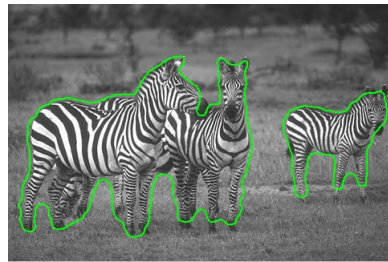
(h) proposed



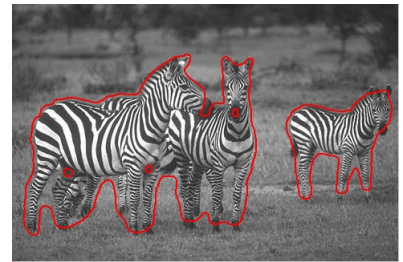
(i) [21]



(j) f



(k) proposed



(l) [21]

Figure 7: Comparisons between the proposed method and the model in [21] on texture segmentation.

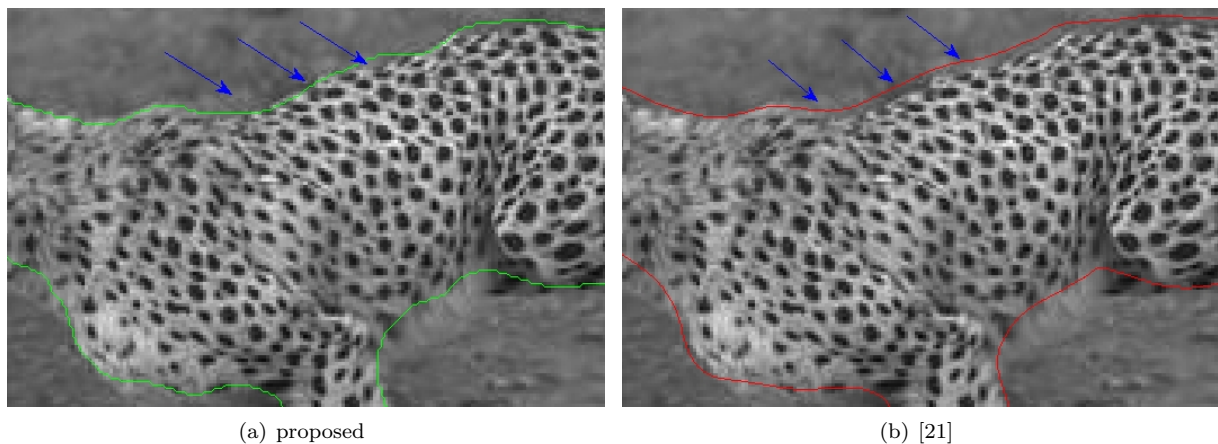
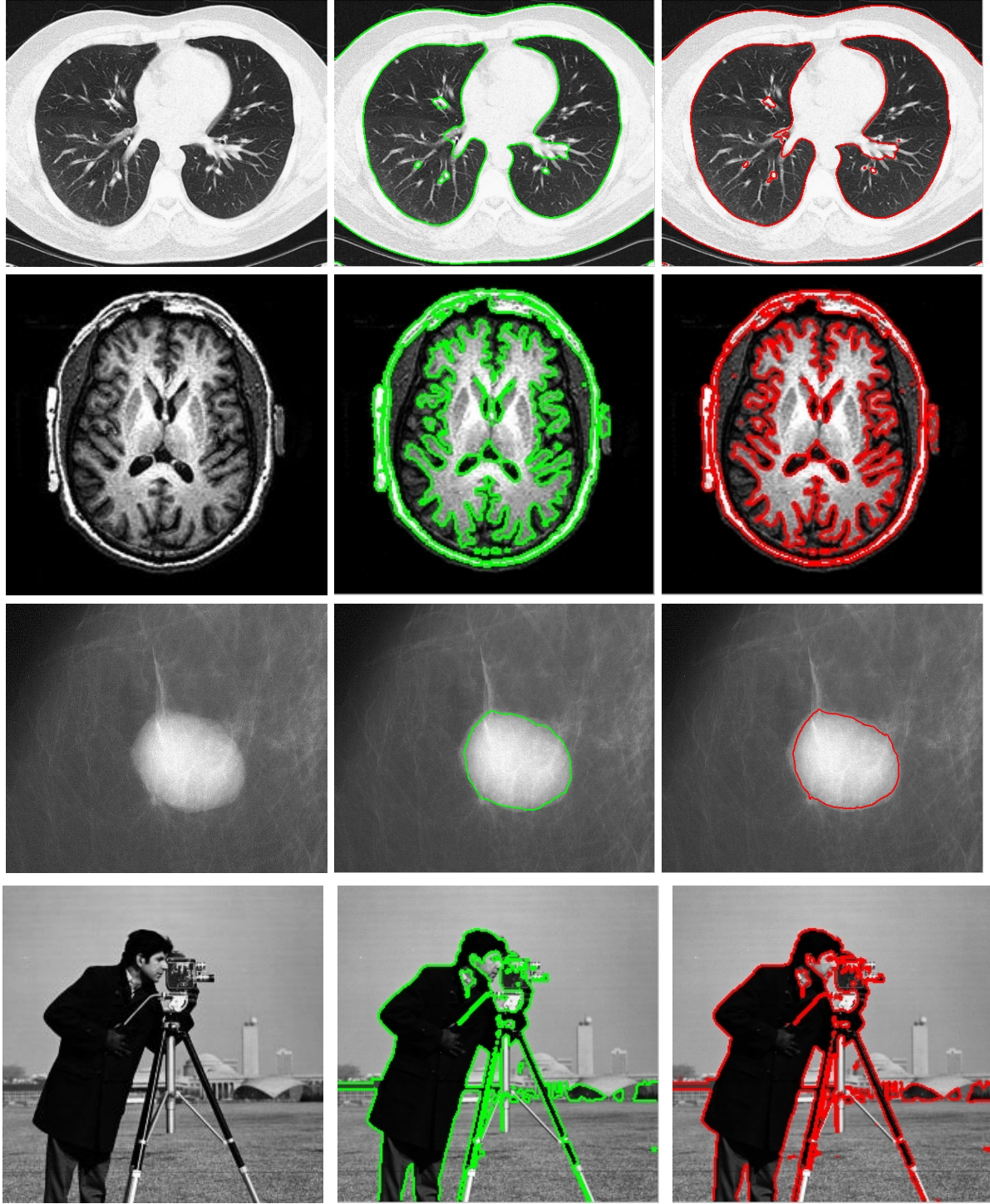


Figure 8: Partial enlarged views of Figure 7(b) and Figure 7(c) respectively.

- [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 Methods in Computer Vision*, 2009, pp. 1-13.
- [10] Q. Du, V. Faber, M. Gunzburger, Centroidal voronoi tessellations: applications and algorithms, *SIAM Rev.* 41 (4)(1999) 637-676.
- [11] Q. Du, M. Gunzburger, L.L. Zhu, Advances in studies and applications of centroidal voronoi tessellations, *Numer. Math. Theor. Meth. Appl.* 3 (2)(2010) 119-142.
- [12] 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.
- [13] 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.
- [14] D. Mumford, J. Shah, Optimal approximation by piecewise smooth functions and associated variational problems, *Comm. Pure Appl. Math.* 42 (1989) 577-685.
- [15] T.F. Chan, L.A. Vese, Active contours without edges, *IEEE Trans. Image Process.* 10 (2)(2001) 266-277.
- [16] 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.
- [17] S. Marcelja, Mathematical description of the response of simple cortical cells, *J. Optical Soc. Am.* 70 (1980) 1297-1300.
- [18] T. Brox, J. Weickert, B. Burgeth, P. Mrzek, Nonlinear structure tensors, *Image Vision Comput.* 24 (1)(2006) 41-55.
- [19] C. Sagiv, N. Sochen, Y.Y. Zeevi, Integrated active contours for texture segmentation, *IEEE Trans. Image Process.* 15 (6)(2006) 1633-1646.
- [20] N. Sochen, R. Kimmel, R. Malladi, A general framework for low level vision, *IEEE Trans. Image Process.* 7 (3)(1998) 310-318.
- [21] 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.
- [22] 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.
- [23] 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.
- [24] 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, pp. 1-7.
- [25] C.M. Li, C.Y. Kao, J. C. Gore, Z.H. Ding, Minimization of region-scalable fitting energy for image segmentation, *IEEE*



(a) Original image

(b) proposed Alg. 4

(c) Chan-Vese [31]

Figure 9: Results obtained by the proposed algorithm and the Chan-Vese model with split Bregman iteration [31].

- Trans. Image Process. 17 (10)(2008) 1940-1949.
- [26] X.F. Wang, D.S. Huang, H. Xu, An efficient local Chan-Vese model for image segmentation, Pattern. Recogn. 43 (3)(2010) 603-618.
 - [27] Z. Karel, Contrast limited adaptive histogram equalization, Graphic Gems IV, Academic Press Professional, Inc., San Diego, CA, USA, 1994, pp. 474-485.
 - [28] X. Bresson, T.F. Chan, Non-local unsupervised variational image segmentation models, CAM Report 07-23, 2007.
 - [29] 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.
 - [30] 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.
 - [31] T. Goldstein, X. Bresson, S. Osher, Geometric applications of the split Bregman method: segmentation and surface reconstruction, UCLA CAM Report 09-06, 2009.
 - [32] X. Tai, C. Wu, Augmented Lagrangian method, dual methods and split Bregman iteration for ROF model, UCLA CAM Report 09-05, 2009.
 - [33] 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. (2010) DOI: 10.1007/s11263-010-0351-9.