

# A Continuous Random Walk Model With Explicit Coherence Regularization for Image Segmentation

Mengfei Li<sup>1</sup>, Hong Gao, Feifei Zuo, and Hongwei Li<sup>2</sup>

**Abstract**—Random walk is a popular and efficient algorithm for image segmentation, especially for extracting regions of interest (ROIs). One difficulty with the random walk algorithm is the requirement for solving a huge sparse linear system when applied to large images. Another limitation is its sensitivity to seeds distribution, i.e., the segmentation result depends on the number of seeds as well as their placement, which puts a burden on users. In this paper, we first propose a continuous random walk model with explicit coherence regularization (CRWCR) for the extracted ROI, which helps to reduce the seeds sensitivity, so as to reduce the user interactions. Then, a very efficient algorithm to solve the CRWCR model will be developed, which helps to remove the difficulty of solving huge linear systems. Our algorithm consists of two stages: initialization by performing one-dimensional random walk sweeping based on user-provided seeds, followed by the alternating direction scheme, i.e., Peaceman–Rachford scheme for further correction. The first stage aims to provide a good initial guess for the ROI, and it is very fast since we just solve a limited number of one-dimensional random walk problems. Then, this initial guess is evolved to the ideal solution by applying the second stage, which should also be very efficient since it fits well for GPU computing, and 10 iterations are usually sufficient for convergence. Numerical experiments are provided to validate the proposed model as well as the efficiency of the two-stage algorithm.

**Index Terms**—Random walk, image segmentation, Peaceman-Rachford scheme.

## I. INTRODUCTION

IMAGE segmentation models, e.g. energy-based ones, could be labeled as continuous models or discrete models according to their formulations. The former includes the Mumford-shah model [1], Chan-ese [2] model, the Potts model [3] as well as the active contour (AC) and geodesic active contour (GAC) models [4], [5], while the latter mainly consists of markov random field-based graph models.

A vast segmentation algorithms to solve the above mentioned models or their various relaxations have been proposed in the literature. In the continuous setting, the technique of

convex relaxation has been extensively utilized to develop fast and robust approximation algorithms [6]–[10]. Level-set method [11] based curve evolution is another popular algorithm [12]–[16]. In the discrete setting, efficient algorithms include the maxflow [17] based graph-cut algorithms like  $\alpha$ – $\beta$  swap and  $\alpha$ –expansion [18]–[20], Grow-cut [21] etc., while for non-maxflow based graph algorithms, we have EWCVT [22], normalized cuts [23], random walk (RW) algorithm [24] etc. The high computational complexity and memory consumption of the graph-cut algorithms have limited their use to problems of moderated size. Graph-cut with GPU acceleration, developed in 2008 [25] and more recently in 2015 [26], has greatly reduced its computational complexity.

Continuous models are very flexible in the sense that they could be easily modified, adapted or relaxed to construct various approximate models. By this way, efficient and robust algorithms could be developed. On the other hand, Discrete models and their solving algorithms are more general in the sense that they are formulated on graphs which don't assume regular structures, i.e. the nodes are not required to be arranged in a lattice grid.

When working on images, which do possess regular lattice grids, the discrete models or algorithms could sometimes be reformulated in the continuous setting such that powerful algorithms were derived. For example, the maxflow algorithm (whether implemented by augmenting path or push-relabel) was originally designed for graph-cut. Appleton and Talbot [27] introduced a continuous maximal flow system which is described by a partial differential equation system. An efficient solver for flow simulation was also developed by the authors. Yuan *et al.* [28] introduced alternative continuous max-flow models and the technique of augmented Lagrangian was utilized to design efficient algorithms for graph-cut segmentation.

The basic idea of random walk was introduced by Karl Pearson in 1905, while the RW algorithm for image segmentation was proposed by Grady [24], [29]. Since then, RW has been extensively researched and widely used for image segmentation, especially for extracting regions of interest (ROI). The connections between RW, graph-cut, normalized-cuts and MRF have been investigated in [24], [29], and [30]. The connection between the RW algorithm and heat diffusion has also been investigated in the literature [31]–[35].

Although the RW algorithm has been successfully applied for image segmentation, it still suffers from some limitations. Firstly, huge sparse linear systems need to be solved when

Manuscript received January 29, 2018; revised July 14, 2018 and September 26, 2018; accepted November 11, 2018. Date of publication November 16, 2018; date of current version December 3, 2018. This work was supported by the National Natural Science Foundation of China under Grant 61771324 and Grant 61871275. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Mireille Boutin. (Corresponding author: Hongwei Li.)

M. Li, H. Gao, and H. Li are with the School of Mathematical Sciences, Capital Normal University, Beijing 100048, China, and also with the Beijing Advanced Innovation Center for Imaging Technology, Capital Normal University, Beijing 100048, China (e-mail: hongwei.li91@cnu.edu.cn).

F. Zuo is with LargeV Instrument Corporation Ltd., Beijing 100084, China. Digital Object Identifier 10.1109/TIP.2018.2881907

applying the RW algorithm on large images, which might still be difficult for modern linear system solvers. Secondly, the segmentation results are sensitive to the distribution of the seeded nodes, including the number of seeds as well as their placement. This places a burden on users. The sensitivity of seeds distribution could be understood as that the RW model lacks regularization and more prior information is needed.

The main contributions of this paper can be summarized as follows.

- A continuous model is formulated for the random walk algorithm, which approximates well the discrete energy defined for the RW model when working on images. This continuous formulation leads to a fast solver for RW based segmentation problems.
- A constraint term that explicitly measures the intensity coherence of the extracted ROI is introduced and integrated into the continuous RW model, which helps to reduce seeds sensitivity, so as user interactions.
- An efficient algorithm to solve the continuous model is proposed, which aims to remove the difficulty with solving huge linear systems when applying random walk based algorithms.

The remainder of this paper is organized as follows. In Section II, more recent development of RW based algorithms will be reviewed and analyzed. In Section III, we review the theory of traditional RW algorithm, and the RW model shall be reformulated in the continuous setting. In Section IV, the alternating direction scheme, i.e. the Peaceman-Rachford (PR) scheme for solving the huge linear systems is introduced. The coherence constraint term is proposed in Section V, followed by numerical experiments and analysis to verify its effectiveness. Then, a very efficient two-stage algorithm will be developed to solve the constrained model. Section VI dedicates to numerical experiments and analysis for validating the proposed model and algorithm. In Section VII, we conclude this paper with discussions.

## II. RELATED WORK

For dealing with the huge linear systems arising from applying the RW algorithm, various methods have been proposed in the literature. Grady *et al.* [24] have tested the GPU implementation of the congruent gradient solver. In [36]–[38], the multi-grid (MG) solver and GPU acceleration have been investigated. In [39] and [40], the technique of super-pixel and super-voxel were utilized to reduce the number of vertex nodes.

The preconditioned conjugate gradient (PCG) [41] method is usually a good choice for solving large sparse symmetric linear systems. The preconditioner could be the simple Jacobi preconditioner [42] or constructed from the incomplete Cholesky Factorization (ICF) [43]. In our experiments, while the PCG solver converges with Jacobi preconditioner for images size of  $128^2$ , it might fail to converge for images size of  $1024^2$  in a reasonable number of iterations. This is understandable since the condition number of the linear systems could be as large as  $O(N^2)$ , where  $N$  denotes the number of pixels, and the Jacobi preconditioner usually is not quite effective in reducing it. For the linear systems involved with

the RW algorithm, the condition number might be much worse since there are very small (close to zero) values corresponding to edge nodes, which could promote or lead to singularity. Convergence behavior can be much improved with an accurate enough ICF preconditioner. However, more accuracy means more computational complexity in construction and application, as well as more memory consumption, which could be prohibitive for very large matrix. The MG solver has been proven to be very efficient for solving linear systems obtained from discretization of elliptic partial differential equations. However, when solving the huge linear systems from applying the RW algorithm, difficulties might arise in devising the prolongation, restriction and coarsening operators [36], [37]. So, when the data set is very large, new linear system solvers specific for the problem at hand need to be developed.

The RW algorithm has also been extended in various ways to deal with more challenging segmentation problems. The lazy random walk algorithm (LRW) [44], [45], was utilized as a key ingredient for superpixel generation in [40]. The partially absorbing random walk algorithm (PARW) [46], was generalized and employed for video supervoxel segmentation in [47]. Please refer to [48]–[50] for more applications utilizing the idea of random walk for video object detection. The subMarkov random walk (SMRW), which was proposed in [51], tried to unify many other random walk based algorithms including RW, LRW and PARW. By adding auxiliary nodes, SMRW can encode additional information to its graph, so as to effectively segment objects with tubular structures.

The above mentioned methods, while successfully extending the RW algorithm to suit different scenarios, they don't take the computational complexity or seeds sensitivity into consideration. Indeed, the Markov nature and the linear transition probability relationship between adjacent nodes would lead to a sparse linear system to describe the stationary distribution. To reduce the seeds sensitivity, additional prior information is needed to regularize the RW model. The LRW and PARW algorithms, although assuming different transition probabilities, incorporate no additional prior information regarding to the image content. For the SMRW algorithm [51], prior regarding to the color distributions of foreground and background has been incorporated. However, this prior intends to provide shortcuts to nodes having difficulties to "communicate" to the seeded nodes, rather than to reduce seeds sensitivity. Similar idea (known as label prior) can be found in [52].

So, the two limitations, which seem internal to RW based algorithms, still exist in the state-of-the-art algorithms, which calls for further research for segmenting huge images by applying the RW based algorithms.

## III. THE CONTINUOUS MODEL FOR THE RANDOM WALK ALGORITHM

In this section, we will build a continuous energy model for the traditional RW algorithm, which is the basis for developing the fast alternating direction PR scheme.

The traditional RW algorithm was formulated in the discrete setting. The image is mapped into an undirected weighted graph  $\langle V, E \rangle$ , where  $V$  consists of vertices (nodes) and  $E$

denotes the set of edges  $\{e|e \in E \subseteq V \times V\}$ . The vertices on the graph represent pixels in the image. Denote  $e_{ij}$  as the edge connecting two adjacent nodes  $v_i$  and  $v_j$ . Each edge is assigned a value  $w_{ij}$  called weight, which is usually defined as

$$w_{ij} = \exp(-\beta(I_i - I_j)^2), \quad (1)$$

where  $\beta$  is a adjustable parameter, and  $I_i$  and  $I_j$  denote the gray values of the adjacent pixels corresponding to nodes  $v_i$  and  $v_j$ .

The combinatorial Laplace matrix [53] is defined as:

$$L_{ij} = \begin{cases} d_i & \text{if } i = j, \\ -w_{ij} & \text{if } v_i, v_j \text{ are adjacent nodes,} \\ 0 & \text{otherwise,} \end{cases}$$

where  $d_i = \sum_j w_{ij}$ . The matrix  $L$  is sparse, semi-definite and symmetric.

Let  $u$  denote the vector that defines the probability field on the vertices, then the traditional RW algorithm tries to find a minimizer of the following energy

$$D[u] = \frac{1}{2} u^T L u \quad (2)$$

s.t.

$$R u = b, \quad (3)$$

where  $b$  is a vector consisting of zero elements except on the specific locations corresponding to the foreground seeds where their values are set to 1, and  $R$  is a diagonal matrix with values of zero on the diagonal except on the specific locations corresponding to the foreground and background seeds, where their values are set to 1. So, the traditional RW algorithm tries to find the minimizer of the above discrete energy with the constraint that the values of the seeds are fixed to 0 (background) or 1 (foreground). We call (2) the traditional RW model. The model (2) is motivated by the classical Dirichlet integral defined as

$$D[v] = \frac{1}{2} \int_{\Omega} |\nabla v|^2 d\Omega, \quad (4)$$

for potential field  $v$  and region  $\Omega$ . To find a minimizer of (4), one could solve its Euler-Lagrangian equation

$$\nabla^2 v = 0, \quad (5)$$

with suitable boundary conditions. Grady [29] showed that with a properly defined metric regarding to the image contents, the Dirichlet integral could be “translated” to the traditional RW model.

It’s worth to point out that the energy of the traditional RW model is not equivalent to the Dirichlet integral since a direct discretization of (4) is not an approximation to (2), simply because the equation (4) is isotropic and has nothing to do with the image contents. In the following, we will formulate the traditional RW model in the continuous setting to establish a corresponding continuous energy.

Suppose a continuous image  $I(x, y)$  is defined on some rectangular domain  $\Omega$ , and users have marked some foreground seeds as well as background seeds. Let  $v(x, y)$  denote

the continuous potential field we are looking for. Define the following energy

$$E_1(v) = \frac{1}{2} \int_{\Omega} [g(I_x)v_x^2 + g(I_y)v_y^2] d\Omega, \quad (6)$$

s.t.

$$\left. \frac{\partial v}{\partial \bar{n}} \right|_{\partial\Omega} = 0, \quad (7)$$

$$\mathfrak{R}v = \mathfrak{b}, \quad (8)$$

where  $\bar{n}$  denotes the unit outward normal of  $\Omega$ ,  $\mathfrak{b}$  is a field consisting of zero elements except on the locations corresponding to the foreground seeds where their values are set to 1, and  $\mathfrak{R}$  denotes the projection operator

$$\mathfrak{R}v(x, y) = \begin{cases} v(x, y), & (x, y) \text{ is a seeded point;} \\ 0, & \text{otherwise.} \end{cases}$$

Note that  $\mathfrak{R}$  and  $\mathfrak{b}$  are the continuous analogs of  $R$  and  $b$  in (3), i.e.  $R$  and  $b$  should approximate  $\mathfrak{R}$  and  $\mathfrak{b}$  in the discrete setting, and  $(I_x, I_y)$  denotes the image gradient. The function  $g(\cdot)$  is defined as

$$g(\cdot) = \exp(-\beta|\cdot|^2).$$

We call (6) the continuous RW (CRW) model, and the traditional RW model could be thought of as being an approximation to (6). We acknowledge that the above continuous RW model has been proposed in the literature, e.g. see [32]. However, the following derivation is still valuable to serve our purpose. The Euler-Lagrangian equation of (6) reads

$$-\frac{\partial}{\partial x} \left( g(I_x) \frac{\partial v}{\partial x} \right) - \frac{\partial}{\partial y} \left( g(I_y) \frac{\partial v}{\partial y} \right) = 0. \quad (9)$$

To solve (9), we need to discretize both the domain  $\Omega$  as well as the equation. Define

$$\Delta_{L_x} v = -\frac{\partial}{\partial x} \left( g(I_x) \frac{\partial v}{\partial x} \right), \quad \Delta_{L_y} v = -\frac{\partial}{\partial y} \left( g(I_y) \frac{\partial v}{\partial y} \right),$$

and

$$\Delta_L = \Delta_{L_x} + \Delta_{L_y}.$$

Let the domain  $\Omega$  be uniformly discretized and approximated by  $\Omega_h = \{(x_i, y_j), x_i = i * h, y_j = j * h, i = 0, 1, \dots, M - 1, j = 0, 1, \dots, N - 1\}$ . Define  $\partial\Omega_h = \{(x_0, y_j) \cup (x_{M-1}, y_j) \cup (x_i, y_0) \cup (x_i, y_{N-1}), i = 0, 1, \dots, M - 1, j = 0, 1, \dots, N - 1\}$ . For a digital image of size  $M \times N$ , each pixel could be mapped to a grid node in  $\Omega_h$ . Let  $v_{ij}$  denote  $v(x_i, y_j)$ , and define

$$\delta_{L_x}^2 v_{ij} = -g_{i-\frac{1}{2},j} v_{i-1,j} + (g_{i-\frac{1}{2},j} + g_{i+\frac{1}{2},j}) v_{i,j} - g_{i+\frac{1}{2},j} v_{i+1,j},$$

$$\delta_{L_y}^2 v_{ij} = -g_{i,j-\frac{1}{2}} v_{i,j-1} + (g_{i,j-\frac{1}{2}} + g_{i,j+\frac{1}{2}}) v_{i,j} - g_{i,j+\frac{1}{2}} v_{i,j+1},$$

where  $g_{i-\frac{1}{2},j} = g((I_x)_{i-\frac{1}{2},j}) \approx \exp(-\beta(I_{i,j} - I_{i-1,j})^2)$ , and  $g_{i+\frac{1}{2},j}$ ,  $g_{i,j-\frac{1}{2}}$  and  $g_{i,j+\frac{1}{2}}$  could be defined and discretized in

a similar manner. Then a second order approximation to (9) on the grid  $(x_i, y_j)$  could be done as follows

$$\Delta_{L_x} v_{ij} = \frac{\delta_{L_x}^2 v_{ij}}{h^2} + O(h^2), \quad \Delta_{L_y} v_{ij} = \frac{\delta_{L_y}^2 v_{ij}}{h^2} + O(h^2),$$

so

$$\Delta_L v_{ij} = \frac{\delta_{L_x}^2 v_{ij}}{h^2} + \frac{\delta_{L_y}^2 v_{ij}}{h^2} + O(h^2) = 0. \quad (10)$$

Let  $u$  denote the discrete function that approximates  $v$  on grid points which satisfies

$$\frac{\delta_{L_x}^2 u_{ij}}{h^2} + \frac{\delta_{L_y}^2 u_{ij}}{h^2} = 0, \quad (11)$$

i.e.

$$(\delta_{L_x}^2 + \delta_{L_y}^2)u_{ij} = 0, \quad (12)$$

then (12) is a second order approximation to (9), which is actually obtained by omitting the  $O(h^2)$  term in (10). For boundary nodes, i.e.  $(x_i, y_j) \in \partial\Omega_h$ , to keep the second order accuracy, we need to introduce fictitious nodes and then eliminate them by the boundary condition described by (7). Moreover, one needs to define terms like  $g_{-\frac{1}{2},j}$  and  $g_{M-\frac{1}{2},j}$  etc. To coincide with the traditional RW algorithm, these terms shall be treated as zeros, which means that the points inside  $\Omega$  have no interactions with outside points. In matrix-vector form, (12) could be written as

$$Lu = 0, \quad (13)$$

where  $L$  is the matrix defined in (2) when the traditional RW algorithm is applied to 4-adjacent graphs mapped from images of size  $M \times N$ , and  $u_{ij}$  has been concatenated column by column into vector  $u$ .

Due to the Neumann boundary conditions, the matrix  $L$  is singular. This singularity shall be removed by imposing the constraint (8), which could be approximated well by (3). It's easy to see that equation (13) is the Euler-Lagrangian equation for equation (2). Since (13) is a second order approximation to (9) on the grid points, which is the Euler-Lagrangian equation for (6), we conclude that (2) and (6) are modeling the same potential field, such that one is formulated in the discrete setting while the other one is in the continuous setting.

For the continuous model, one still needs to solve a huge sparse linear system if we just discrete its Euler-Lagrangian equation directly. However, the continuous form of the model makes alternative solution methods possible. In the next section, an operator splitting approach will be proposed to solve the continuous model efficiently.

#### IV. THE PEACEMAN-RACHFORD (PR) SCHEME FOR SOLVING THE HUGE LINEAR SYSTEM

The equation (12) can be solved by the traditional RW algorithm described in [29]. For later reference, we review it briefly. Denote the seeded nodes as  $V_M$  and the unseeded nodes as  $V_U$ , which satisfy  $V_M \cup V_U = V$  and  $V_M \cap V_U = \emptyset$ . By rearranging the nodes, the matrix  $L$  can be decomposed as

$$\begin{bmatrix} L_M & B \\ B^T & L_U \end{bmatrix}. \quad (14)$$

The objective function is then decomposed as

$$\begin{aligned} D[x_U] &= \frac{1}{2} [x_M^T \quad x_U^T] \begin{bmatrix} L_M & B \\ B^T & L_U \end{bmatrix} \begin{bmatrix} x_M^T \\ x_U^T \end{bmatrix} \\ &= \frac{1}{2} (x_M^T L_M x_M + 2x_U^T B^T x_M + x_U^T L_U x_U), \end{aligned} \quad (15)$$

where  $x_M$  and  $x_U$  represent the potentials for seeded nodes and unseeded nodes respectively. To minimize (15), one just needs to solve the following linear equation

$$L_U x_U = -B x_M. \quad (16)$$

The traditional RW algorithm decomposes the potential field into  $x_U$  and  $x_M$ , and only  $x_U$  is solved in (16). In this way, the constraint  $Ru = b$  in (2) is directly absorbed and handled. The drawback of this approach is that it ruins the underlying structure of the potential field, i.e. the solution could not be mapped to a structured grid anymore. When considering some efficient solvers such as the Geometric multi-grid method, or parallel computing devices such as GPU, structured grid is usually required for better performance. So our first effort is to deal with the constraint in a different way, so as to preserve the grid structure.

#### A. Dealing With the Constraint—Penalty Method

To preserve the grid structure, we treat the whole potential field as unknown, and handle the seeded nodes by the penalty method. The traditional RW model (2) is a constrained minimization problem. By utilizing the classical penalty method, the constrained model could be transformed into an unconstrained one

$$D[u] = \frac{1}{2} u^T L u + \frac{\lambda}{2} \|Ru - b\|_2^2. \quad (17)$$

By this way, we don't need to separate the seeded nodes out from the unseeded ones. The Euler-Lagrangian equation of (17) reads

$$(L + \lambda R^T R)u = \lambda R^T b, \quad (18)$$

where  $L$  is a real symmetric five-diagonal matrix. It's easy to verify that  $R^T = R$ , so  $R^T R$  is a sparse diagonal matrix, and  $L + \lambda R^T R$  is a symmetric five-diagonal sparse matrix. On the continuous setting, the corresponding Euler-Lagrangian equation should read

$$(\Delta_L + \lambda \mathfrak{R}^* \mathfrak{R})v = \lambda \mathfrak{R}^* b, \quad (19)$$

where  $\mathfrak{R}^*$  denotes the adjoint operator of  $\mathfrak{R}$ .

Similar ideas for preserving grid structure have been proposed in [31], [37], and [54]. For the penalty method to work, usually the parameter  $\lambda$  must be set very large to effectively impose the constraint. However, we will show in the next subsection that for our model (17), the parameter  $\lambda$  could be fixed to any relative large value, e.g. within the range [1, 100].

The penalty method reveals a close connection between the RW algorithm and the PARW algorithm. In the above context, when a constant absorbing rate (which is parameterized by  $\lambda$ ) is set only for the foreground seeds, then the linear system solved by the PARW algorithm is

$$(L + \lambda R^T R)u = R^T b. \quad (20)$$

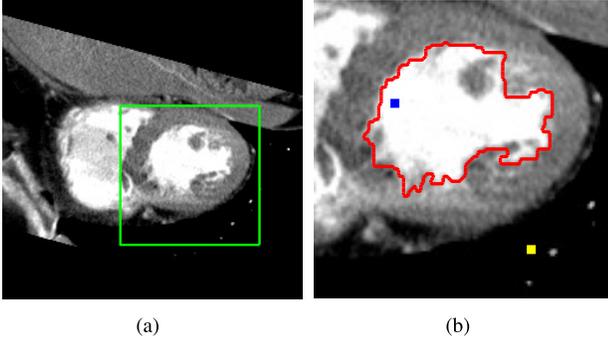


Fig. 1. Sensitivity tests for the parameter  $\lambda$ . (a) The cardiac image slice; (b) the segmentation result by the traditional RW algorithm, with  $\lambda = 100$ .

Compared to (18), the constant  $\lambda$  is absent from the right hand term  $b$ . However, for the PARW algorithm, multiplying a constant to the right-hand term will not change the final labeling. In fact, two similar linear systems will be solved by the PARW algorithm, and the labeling is determined by comparing the magnitudes of the two solutions, which shall not be affected by a scaling constant.

### B. Selecting the Penalty Parameter $\lambda$

In the continuous setting, the unconstrained RW model has the form

$$\min D[u] = \frac{1}{2} \int_{\Omega} |\nabla_L u|^2 d\Omega + \frac{\lambda}{2} \int_{\Omega} (\mathfrak{R}u - \mathfrak{b})^2 d\Omega, \quad (21)$$

where  $\nabla_L u = (g(I_x)u_x, g(I_y)u_y)^T$ , i.e.  $\nabla_L$  denotes the weighted gradient operator  $\left(g(I_x)\frac{\partial}{\partial x}, g(I_y)\frac{\partial}{\partial y}\right)^T$ . Generally speaking, to obtain the solution of the CRW model through (21), the parameter  $\lambda$  must be very large so as to enforce the constraint. Fortunately, we will show below that this general rule can be compromised when solving (21). In our tests, equally good solutions could be produced when we vary  $\lambda$  within the range  $[1, 100]$ . This is demonstrated by the experiments performed on the medical cardiac CT image slice shown in Figure 1(a), which is size of  $256 \times 256$  pixels. Figure 1(b) shows the segmentation result for the framed region specified in Figure 1(a). For this test, one foreground seed and one background seed are marked, as indicated in Figure 1(b). The parameter  $\beta$  is set to 90. The original probability field  $P_{ori}$  was computed through the formula  $x_U = -L_U^{-1} B x_M$ , while the probability field  $P_{\lambda}$  was computed by the formula  $x = (L + R^T R \lambda)^{-1} (\lambda R^T b)$ . Define  $\Delta P = P_{ori} - P_{\lambda}$ . The  $L^{\infty}$ -norm of  $\Delta P$  and the rooted mean square error (RMSE):  $\frac{\|\Delta P\|_2}{\sqrt{N}}$ , where  $N$  is the length of vector  $\Delta P$ , were computed against different values of  $\lambda$ , and the results are illustrated in Figure 2. When  $\lambda > 1$ , both  $\|\Delta P\|_{\infty}$  and RMSE are less than 0.05, which are accurate enough for the final thresholding segmentation. In order to ensure the correctness, we set  $\lambda = 100$  in all the numerical experiments performed in the following sections.

### C. The PR Scheme for Solving the Linear System

Since  $\Delta_L$  could be decomposed into  $\Delta_{L_x}$  and  $\Delta_{L_y}$ , the Operator Splitting method or semi-implicit methods could

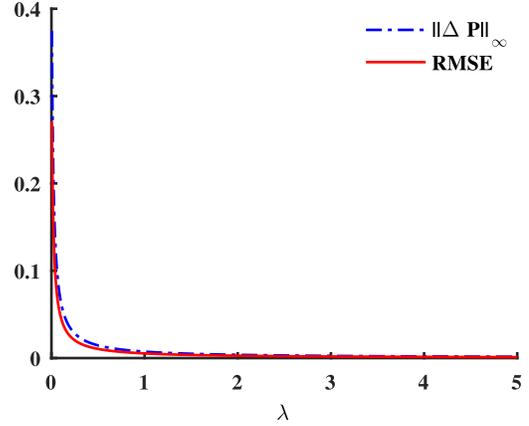


Fig. 2.  $\|\Delta P\|_{\infty}$  and RMSE against the parameter  $\lambda \in [0.005, 5]$ .

be applied. We choose the PR scheme as our solver which is unconditionally stable and second-order accurate in both temporal and space. The solution of equation (19) could be thought of as being the steady state of the following heat conduction equation

$$\frac{\partial v}{\partial t} + (\Delta_L + \lambda \mathfrak{R}^* \mathfrak{R})v = \lambda \mathfrak{R}^* b, \quad (22)$$

where a pseudo-time variable  $t$  has been associated to the continuous potential field  $v$ , and one can use  $v(x, y, t)$  to denote its value at the space-time point  $(x, y, t)$ .

Suppose the temporal domain is discretized by placing a grid on the temporal axis with grid spacing  $\Delta t$ , which could be indexed as  $n = 0, 1, 2, \dots$ . Let  $v^n = v(x, y, n\Delta t)$ . Given initial guess  $v^0$ , for  $n = 0, 1, \dots$ , the PR scheme solves the following two equations sequentially and iteratively until convergence:

$$\frac{v^{n+\frac{1}{2}} - v^n}{\Delta t} + \Delta_{L_x} v^{n+\frac{1}{2}} + \Delta_{L_y} v^n + \lambda \mathfrak{R}^* \mathfrak{R} v^{n+\frac{1}{2}} = \lambda \mathfrak{R}^* b, \quad (23)$$

$$\frac{v^{n+1} - v^{n+\frac{1}{2}}}{\Delta t} + \Delta_{L_x} v^{n+\frac{1}{2}} + \Delta_{L_y} v^{n+1} + \lambda \mathfrak{R}^* \mathfrak{R} v^{n+1} = \lambda \mathfrak{R}^* b. \quad (24)$$

The above equations need to be further discretized in the space domain. Let  $u_{i,j}^n$  denote the grid function that approximates  $v(x, y, t)$  on the grid  $\{(i, j, n), i = 0, 1, \dots, M-1; j = 0, 1, \dots, N-1, n = 0, 1, \dots\}$ , i.e.  $u_{i,j}^n \approx v(i, j, n\Delta t)$ . Define

$$\begin{aligned} \delta_{L_x}^2 u_{ij} &= -g_{i-\frac{1}{2},j} u_{i-1,j} + (g_{i-\frac{1}{2},j} + g_{i+\frac{1}{2},j}) u_{i,j} \\ &\quad - g_{i+\frac{1}{2},j} u_{i+1,j}, \\ \delta_{L_y}^2 u_{ij} &= -g_{i,j-\frac{1}{2}} u_{i,j-1} + (g_{i,j-\frac{1}{2}} + g_{i,j+\frac{1}{2}}) u_{i,j} \\ &\quad - g_{i,j+\frac{1}{2}} u_{i,j+1}, \end{aligned}$$

which are second order approximations to  $\Delta_{L_x} v$  and  $\Delta_{L_y} v$  on the grid respectively. Then the equations for  $u_{i,j}^n$  can be written as

$$\frac{u_{i,j}^{n+\frac{1}{2}} - u_{i,j}^n}{\Delta t} + \delta_{L_x}^2 u_{i,j}^{n+\frac{1}{2}} + \delta_{L_y}^2 u_{i,j}^n + \lambda R^T R u_{i,j}^{n+\frac{1}{2}} = \lambda R b, \quad (25)$$

$$\frac{u_{i,j}^{n+1} - u_{i,j}^{n+\frac{1}{2}}}{\Delta t} + \delta_{L_x}^2 u_{i,j}^{n+\frac{1}{2}} + \delta_{L_y}^2 u_{i,j}^{n+1} + \lambda R^T R u_{i,j}^{n+1} = \lambda R b. \quad (26)$$

$$i = 0, 1, \dots, M-1, \quad j = 0, 1, \dots, N-1.$$

Notice that equation (25) and (26) could be decoupled into  $M$  and  $N$  independent one-dimensional problems, respectively. In matrix form, these one-dimensional problems reduce to tridiagonal linear systems, which could be solved efficiently by the chasing algorithm (Thomas algorithm) [55]. The initial guess  $u^0$  is usually set to zero. One might also build the prior information about the foreground seeds into  $u^0$  by setting the values of their corresponding elements to 1.

#### D. Validation of the PR Scheme

The PR scheme boils down to an iterative solver for the linear system (18) after discretization. To verify the effectiveness of the proposed PR scheme, experiments need to be performed to compare the PR scheme with other popular sparse matrix solvers. We choose the PCG method as the competing solver, since the matrix is symmetric positive definite, which perfectly fits to the PCG method. Generally speaking, ICF preconditioners should be applied to accelerate the CG solver. However, as explained in the introduction section, for very large matrix, the construction and implementation of the ICF preconditioner are costly, or even prohibitive. So, in our experiments, the Jacobi preconditioner is employed when needed, since we aim to solve huge linear systems.

Suppose the size of the image is  $N \times N$ , then the linear system to be solved is size of  $N^2 \times N^2$ . The computational complexity can be analyzed as follows. For each iteration of the PCG method, it needs one matrix (5-diagonals) vector multiplication and three vector dot products, which give total  $8N^2$  multiplication operations. For each iteration of the PR scheme, it needs two matrix (3-diagonals) vector multiplications and  $2N$  Thomas algorithm, which give total  $18N^2$  multiplication operations. So, roughly speaking, compared to the PCG method, the number of floating-point operations needed by the PR method is doubled in each iteration. That's to say, to gain advantages, the PR method must consume no more than half number of the iterations consumed by the PCG method to compute the solution of the underlying linear systems.

The performance tests are done on the rock CT image slice shown in Figure 3(a), where one foreground seed is marked blue, while the background seeds are marked yellow. The framed region indicated by the green rectangle, which is shown in Figure 3(b), will be zoomed-in for better display in the following figures. The original rock image slice is size of  $680 \times 669$ , which comes from a CT scan of a road surface paved by rock and asphalt. To model large images, the rock image is resized to  $1380 \times 1338$  by calling the Matlab function `resize()`. This is a more challenging problem than the cardiac image case, since the image size is much larger and the gray value variations are more complex. So, the median filter with template window size of  $7 \times 7$  is applied on the image before performing any tests.

The PR scheme is unconditionally stable and any large timestep  $\Delta t$  could be utilized. Large  $\Delta t$  usually leads to

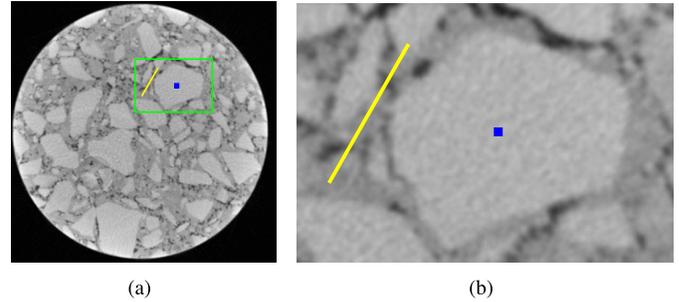


Fig. 3. (a) The rock image, size of  $1380 \times 1338$ . One foreground seed is marked as blue, while the background seeds are marked as yellow; (b) zoomed-in image for the framed region specified in (a).

acceleration. However, in the context of numerical solution of partial differential equations, the PR scheme's accuracy is  $O((\Delta t)^2) + O((\Delta x)^2) + O((\Delta y)^2)$ , which means that large  $\Delta t$  will introduce large error. So, a smarter strategy is that a large  $\Delta t$  is used for the first 10 or 20 iterations, then a small  $\Delta t$  is switched on for better accuracy. In the following tests of PR method, we set  $\Delta t = 10000$  for the first 20 iterations, then  $\Delta t = 100$  is switched on.

The results are shown in Figure 4, where PCG-500 refers to the solution of the linear system, i.e. the probability field, after 500 iterations, and PCG-500-ROI refers to the labeling function by thresholding the probability field (using the threshold 0.5). The other captions follow the same way of interpretations. As shown in Figure 4(a), 500 iterations of PCG only spread the foreground seed information a little bit. After 1000 iterations, the probability field seems quite good as shown in Figure 4(e). However, the labeling function shown in Figure 4(b) shows that the probability field is still far from convergence. After 2000 iterations, PCG converges to the ideal solution, as shown in Figure 4(c) and (f). On the other hand, 50 iterations of the PR method already compute rather good results for both the probability field and the labeling function, as shown in Figure 4(g) and (j). After just about 100 iterations, the PR scheme converges to the ideal solution, since the PR-100 and PR-200 are almost identical, which indicates convergence.

#### V. A NEW CONSTRAINT MODEL

In this section, a regularization term that measures the gray value coherence of the extracted ROI will be introduced and combined into the CRW model, which helps to reduce seeds sensitivity, so as to reduce user interactions. Besides, a very fast algorithm will be proposed to solve the regularized CRW model.

##### A. The Continuous RW Model With Coherence Regularization

The traditional RW model is sensitive to seeds distribution, especially when a small number of nodes are marked. Another issue with traditional RW is the boundary leakage problem, i.e. the boundaries of the extracted ROI are easy to grow out of the ideal ones. This phenomenon is actually observable if one checks the results of previous experiments shown in Figure 1(b) and Figure 4(f). To overcome such a

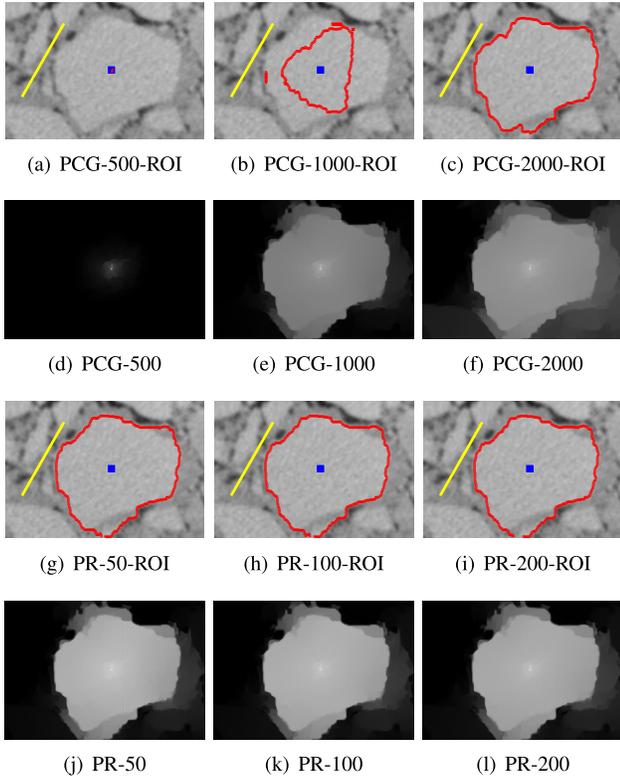


Fig. 4. Efficiency comparison of PCG method and PR scheme. (a)-(c) show the segmentation results by the PCG method; (d)-(f) show the probability field computed by the PCG method; (g)-(i) show the segmentation results by the PR scheme; (j)-(l) show the probability field computed by the PR scheme.

drawback, an intuitive idea is to add a regularization term that measures the intensity coherence inside the extracted ROI. This motivates the following constraint:

$$\int_{\Omega} v^2 |\nabla I| d\Omega < \delta, \quad (27)$$

where  $|\nabla I|$  denotes the norm of image gradient, and  $\delta$  controls the strength of the constraint. Note that for background region,  $v \approx 0$ , while for foreground region,  $v \approx 1$ . So, the constraint intends to measure the total variation of the ROI with respect to the image. The CRW model incorporating the constraint (27) can be formulated as

$$E_2(v) = \frac{1}{2} \int_{\Omega} [g(I_x)v_x^2 + g(I_y)v_y^2] d\Omega, \quad (28)$$

s.t.

$$\mathfrak{R}v = \mathfrak{b}, \quad \int_{\Omega} v^2 |\nabla I| d\Omega < \delta. \quad (29)$$

We name the above model as continuous random walk with coherence regularization (CRWCR). To solve this model, we apply again the penalty method to transform it into an unconstrained one

$$\min E_2[v] = \frac{1}{2} \int_{\Omega} |\nabla_L v|^2 d\Omega + \frac{\lambda}{2} \int_{\Omega} (\mathfrak{R}v - \mathfrak{b})^2 d\Omega + \frac{\gamma}{2} \int_{\Omega} v^2 |\nabla I| d\Omega, \quad (30)$$

whose Euler-Lagrangian equation is

$$(\Delta_L + \lambda \mathfrak{R}^* \mathfrak{R} + \gamma |\nabla I|)v = \lambda \mathfrak{R}^* \mathfrak{b}, \quad (31)$$

where  $\lambda$  and  $\gamma$  are penalty parameters for the seeds constraint and coherence constraint, respectively. The parameter  $\lambda$  can be fixed to 100, as discussed previously. The parameter  $\gamma$ , however, is problem dependent and needs to be tuned to achieve satisfactory result. This is a typical situation for regularization parameters.

There is an interesting connection between the coherence regularization and the label prior introduced in [52] by Leo Grady and further explored in [51]. In the spirit of [52], to incorporate the label prior, two extra seeded “floating” nodes should be added to the graph for two-classes, i.e. foreground and background segmentation, and the label prior is encoded into the weights for the edges connecting the floating nodes and all other nodes. Let  $h^1$  and  $h^2$  denote the foreground float node and background float node respectively, and  $\gamma \lambda_i^s$ ,  $s = 1, 2$  denote the weights for the edge  $e(v_i, h^s)$ , i.e. between node  $v_i$  and float node  $h^s$ , where  $v_i$  is any node other than the floating nodes. Without considering other seeded nodes, the linear equation to be solved in [52] reads

$$\left( L + \gamma \sum_{r=1}^2 \Lambda^r \right) x^s = \gamma \lambda^s, \quad (32)$$

where  $x^s$  denotes the probability field, whose  $i$ th element  $v_i^s$  denotes the probability that a random walker starting from node  $v_i$  first reaches a seeded node with label  $s$ . The symbol  $\Lambda^r$  is understood to be a diagonal matrix with the values of  $\lambda^r$  on the diagonal. Due to  $x^1 + x^2 \equiv 1$ , just one equation needs to be solved. Suppose we choose to solve the first equation that involves with  $x^1$ . If we set  $\lambda^1 \equiv 0$ , and define  $\lambda^2$  as the approximation to the gradient  $|\nabla I|$  on the discrete grid, then the equation (32) reduces to

$$(L + \gamma \Lambda^2) x^1 = 0, \quad (33)$$

which is discretely equivalent to (31) without considering seeded nodes. Alternatively, one can think that if just one floating node marked as background is added to the graph, then with properly defined weights, the CRWCR model (after discretization) reduces to the traditional RW model running on the augmented graph. This analysis leads to the conclusion that the solution of the CRWCR model possesses the same probability interpretation as the solution of the traditional RW algorithm. Especially, the solution of the CRWCR model should map to the range  $[0, 1]$ , i.e. the maximum (minimum) principle applies.

To verify the effectiveness of the proposed CRWCR model, we performed the following experiment. A one-dimensional signal was extracted from the cardiac image, which was shown as the green line in Figure 5(a). One foreground seed and two background seeds were specified. Then the CRWCR model was applied on this one-dimensional signal. Note that in this case, a tridiagonal linear system needs to be solved. Figure 5(b) shows the results of traditional RW model, while Figure 5(c) shows the result of the proposed CRWCR model. From the zoomed-in regions, one can clearly see that the traditional RW model suffers from boundary leakage problem, while the CRWCR model (with  $\gamma = 0.000001$ ) successfully and accurately extracts the foreground.

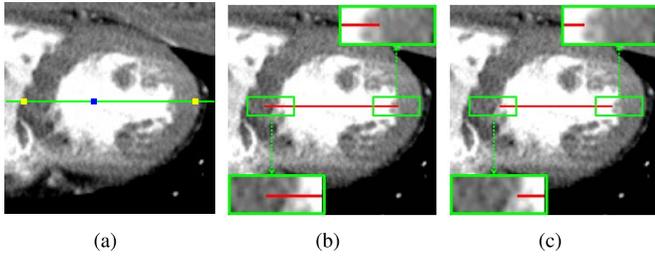


Fig. 5. (a) The green line shows the one-dimensional signal (139th line of the cardiac image), one foreground seed is marked as blue and two background seeds are marked as yellow; (b) the result of traditional RW algorithm; (c) the result of the CRWCR model.

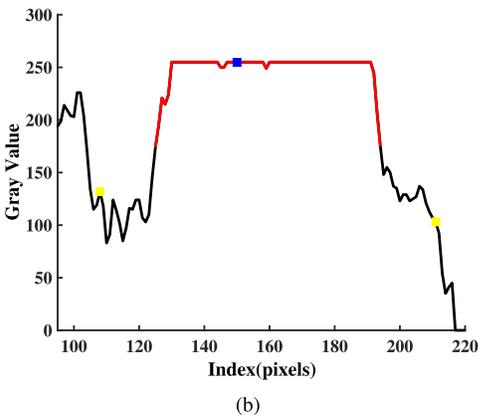
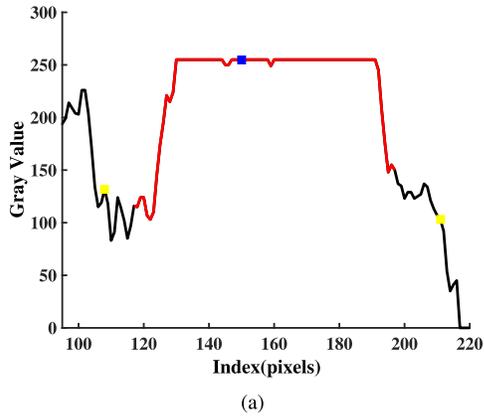


Fig. 6. One-dimensional comparison of the traditional RW model and the proposed CRWCR model. (a) the result of traditional RW algorithm; (b) the result of proposed CRWCR model.

To examine the results further, we plot the one-dimensional signal and the segmentation results in Figure 6, where the foreground seed, background seeds and extracted foreground are marked as blue square node, yellow square node and red curve, respectively. By comparing Figure 6(a) and (b), one can tell that the traditional RW model tends to extract larger foreground, i.e. some background pixels are mis-classified as foreground pixels. On the contrary, the CRWCR model tries to separate the foreground and the background at the favorable places.

Our next experiment is performed on the rock image (size of  $680 \times 669$ ) shown in Figure 7(a), where the framed region is to be zoomed-in for better display. The 420th column of

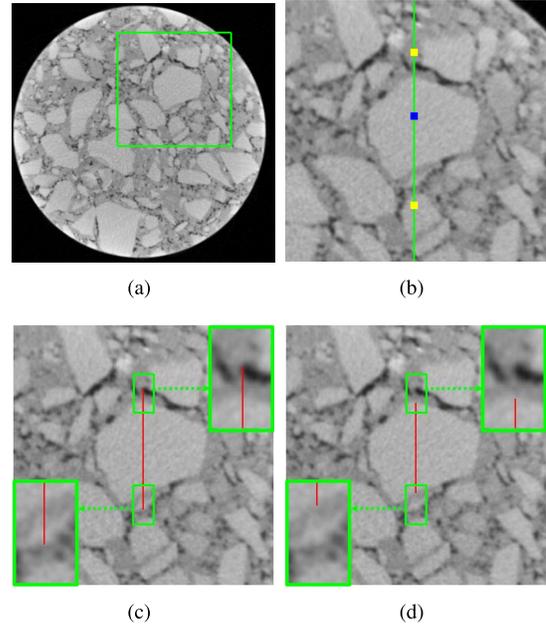
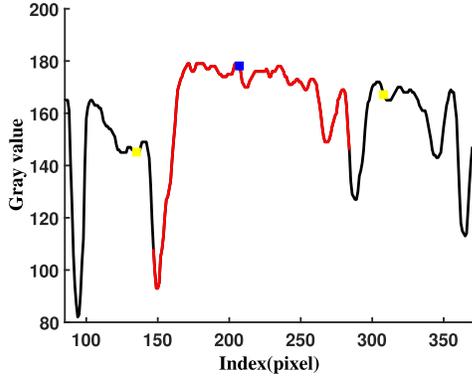


Fig. 7. (a) The rock image, with a framed region; (b) Zoomed-in image for the framed area in (a); (c) the result of the traditional RW model; (d) the result of the CRWCR model.

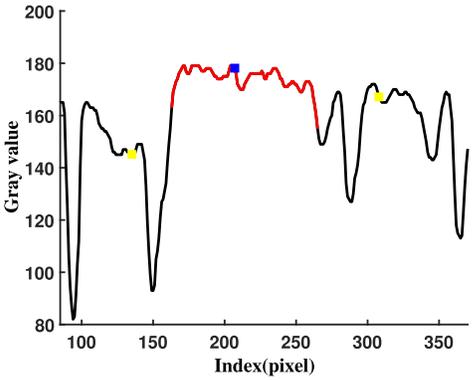
the rock image was extracted as a one-dimensional image to run the tests. As shown in Figure 7(b), one foreground seed and two background seeds are specified. Figure 7(c) and (d) show the segmentation results by the traditional RW model and the CRWCR model, respectively. By examining the zoomed-in regions, one can see that the extracted foreground region, i.e. the red line segment, from the traditional RW model has unfortunately exceeded the real boundaries, as shown in Figure 7(c). On the contrary, the CRWCR model (with  $\gamma = 0.001$ ) extracts the foreground rather accurately, and the ends of the red line segments anchor precisely at the ideal boundary, as shown in Figure 7(d).

To look into more details, we also plot the one-dimensional image and the segmentation results in Figure 8. By comparing Figure 8(a) and (b), one can tell that the traditional RW model seems to be a bit “aggressive”, which is actually unfavorable, in extracting the foreground region. On the contrary, the CRWCR model tries to separate the foreground and the background at the expected locations.

To further demonstrate the effectiveness of the CRWCR model, additional experiments were performed on the two dimensional cardiac and rock images. To avoid possible early stopping of iterative methods, when solving the linear systems (5-diagonal matrices) obtained from the discretization of equation (31), the Matlab backslash solver was called directly. For the CRWCR model, the parameter  $\gamma$  is set to 0.0006. Note that the results are not very sensitive to  $\gamma$ . In our tests,  $\gamma = 0.0001$  also works fine. For both the cardiac image and the rock image, with same seeds distribution, the proposed CRWCR model successfully captures the boundary of the foreground region, while the traditional RW model fails to do so, as shown in Figure 9 (corresponding to the framed region illustrated in Figure 1) and Figure 10 (corresponding to the framed region illustrated in Figure 7).

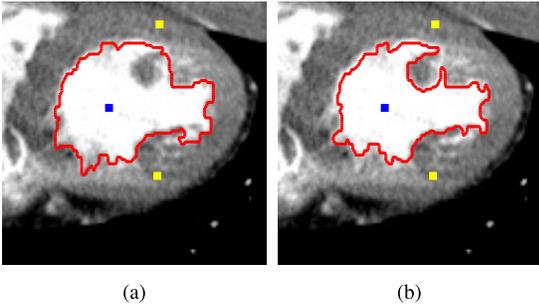


(a)



(b)

Fig. 8. One-dimensional comparison of the traditional RW model against the proposed CRWCR model. Line profiles are extracted from the 420th column of the rock image. (a) the result of the traditional RW model; (b) the result of the CRWCR model.



(a)

(b)

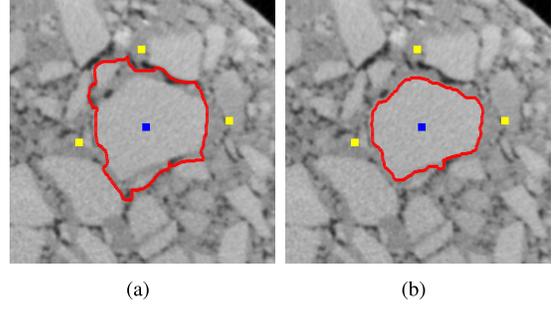
Fig. 9. (a) the result of the traditional RW model; (b) the result of the proposed CRWCR model.

### B. Fast Solver for the CRWCR Model

As discussed in Section I, for huge linear systems, even sparse symmetric positive definite ones, the iterative solvers might fail to work properly in an efficient manner. In this situation, the PR scheme provides a remedy. For our CRWCR model, the PR scheme solves the following two equations iteratively until convergence:

$$\begin{aligned} (1 + \Delta t(\Delta_{L_x} + \gamma|\nabla I| + \lambda\mathfrak{R}^*\mathfrak{R}))v^{n+\frac{1}{2}} \\ = v^n + \Delta t(\lambda\mathfrak{R}^*b - \Delta_{L_y}v^n), \end{aligned} \quad (34)$$

$$\begin{aligned} (1 + \Delta t(\Delta_{L_y} + \gamma|\nabla I| + \lambda\mathfrak{R}^*\mathfrak{R}))v^{n+1} \\ = v^{n+\frac{1}{2}} + \Delta t(\lambda\mathfrak{R}^*b - \Delta_{L_x}v^{n+\frac{1}{2}}). \end{aligned} \quad (35)$$



(a)

(b)

Fig. 10. (a) the result of the traditional RW model; (b) the result of the proposed CRWCR model.

Note that the PR scheme fits well for task-level parallel computing, since both equation (34) and (35) can be decoupled into a series of one-dimensional equations which can be solved independently. To further accelerate the PR scheme, we propose to combine it with an initialization procedure, which aims to provide a good initial guess. This is realized by solving one-dimensional CRWCR models for each row and column of the image. So, the proposed algorithm consists of two stages, which can be summarized as **Algorithm 1**, where  $\text{CRWCR}(I, F, B)$  applies the CRWCR model on image  $I$ , with foreground seeds  $F$  and background seeds  $B$ . Its output should be the set of pixel indices that define the foreground region. For convenience, instead of using one dimensional global indexing of the nodes (pixels), we return to the two dimensional pixel coordinate indexing, which we think is more appropriate to describe our algorithm. So, in the description of **Algorithm 1**,  $(i, j)$  refers to the node (pixel) resides in the  $i$ th row and  $j$ th column of the image.

For the traditional RW algorithm, the threshold for the final thresholding segmentation is usually set to 0.5. In the initialization stage, however, the parameter  $T1$  is set to 0.6. This is necessary to make sure that the foreground computed by the initialization procedure belongs to the ideal foreground. For the same reason, the parameter  $\gamma$  is set to 0.2, which is much larger than that in the PR correction stage. Based on our experience, the iteration number for both the initialization procedure and the PR correction procedure is set to 10. For more challenging problems, it might need to be increased accordingly. Some convergence measure might also be employed to adapt this number automatically. In the initialization procedure, it's possible that there exist only foreground seeds for certain rows or columns. To deal with such cases, the two ending pixels of the image rows and columns are always specified as background seeds.

## VI. NUMERICAL EXPERIMENTS AND ANALYSIS

To demonstrate the effectiveness and efficiency of the proposed CRWCR model and the two-stage algorithm, various experiments will be performed on the cardiac image, rock image and images from public image data set.

### A. Efficiency Test: Comparison With PCG

In previous tests, the efficiency of PR scheme against the PCG method has been validated on solving huge

**Algorithm 1** Two-Stage Algorithm

---

```

1: procedure STAGE 1: INITIALIZATION
2:   Input : image  $I$ , seeds  $S$ 
3:   Output: foreground seeds  $F$ , background seeds  $B$ .
4:   Set  $iter = 0, \beta = 90, \lambda = 100, \gamma = 0.2, T1 = 0.6$ .
5:   repeat
6:     For each  $i = 1, 2, \dots$ , if  $\exists j$ , such that  $(i, j) \in S$ ,
       compute  $u = \text{CRWCR}(I(i, :), S, B)$ .
7:      $F \leftarrow F \cup \{(i, k) : u(k) > T1\}$ 
8:      $S \leftarrow F$ 
9:     For each  $j = 1, 2, \dots$ , if  $\exists i$ , such that  $(i, j) \in S$ ,
       compute  $u = \text{CRWCR}(I(:, j), S, B)$ .
10:     $F \leftarrow F \cup \{(k, j) : u(k) > T1\}$ 
11:     $S \leftarrow F$ 
12:     $iter \leftarrow iter + 1$ 
13:  until  $iter > 10$ 
14: end procedure
15: procedure STAGE 2: PR SCHEME CORRECTION
16:   Input: image  $I$ , foreground seeds  $F$ , background seeds
        $B$ 
17:   Output: foreground  $F$ 
18:   Set  $u^0 = \text{zeros}(\text{size}(I)), u^0(F) = 1, n = 0, T2 =$ 
        $0.5, \gamma = 0.0006$ .
19:   repeat
20:      $n \leftarrow n + 1$ 
21:     Compute  $u^{n+1/2}$  by solving equation (34)
22:     Compute  $u^{n+1}$  by solving equation (35)
23:   until  $iter > 10$ 
24:    $F \leftarrow F \cup \{(i, j) : u^{n+1}(i, j) > T2\}$ 
25: end procedure

```

---

linear systems. For the two-stage algorithm, the second-stage, i.e. the PR scheme can theoretically be replaced by any iterative solver, especially by the PCG method. In this subsection, we will show that the PR scheme employed in the second stage usually still performs better than the PCG method.

It needs to be pointed out that if the first stage of initialization provides a very good approximation to the ideal solution, then PCG, or maybe any other iterative solver, will generally work efficiently, since just small local corrections are needed. However, the initialization might not always perform well. Figure 11(a) shows such a situation. With two foreground seeds (marked as blue) and two background seeds (marked as yellow), the initialization stage (10 rounds of sweeping) fails to capture the left part of the boundary. Note that the experiments are done on the original rock image, and the initialization is actually carried out with a large  $\gamma = 0.001$  on purpose, which will restrict the diffusion ability of the foreground seeds. For such a relatively bad initialization, PR could be still much more efficient than the PCG method. As shown in Figure 11(b) and (e), for both 10 iterations, PCG improves the initialization a bit, while PR already captures well the ROI's boundary. As a comparison, the PCG method needs about 100 iterations to obtain a satisfactory result.

The result of PR after 50 iterations is shown in Figure 11(f). As one can see that there is no obvious difference between

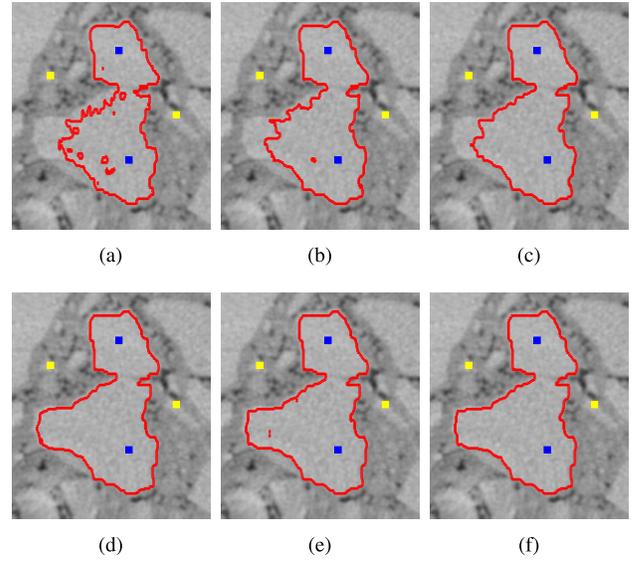


Fig. 11. Efficiency comparison with PCG. (a) the result of initialization; (b)-(d) the results of PCG after 10, 20 and 100 iterations, respectively; (e)-(f) the results of PR scheme after 10 and 50 iterations, respectively.

the results shown in Figure 11(d) and (f). In fact, the PR scheme converges in 20 iterations for this test. PR-50 rather than PR-20 is shown because it helps to demonstrate how PCG and PR perform with same computational complexity. By further comparing PR-10 and PCG-20, one can come to the conclusion that with enough iterations, PR and PCG produce same results. However, if one just want the results in 10 (or 20) iterations, then the PR scheme gives much better result than the PCG method. Please note that in this test, the parameter  $\Delta t$  is fixed to 100 and the smarter strategy for setting  $\Delta t$  is not adopted.

### B. Effectiveness Tests on Reducing User Interactions

1) *Test on the Cardiac Image:* Figure 12(a) shows the result of the traditional RW algorithm corresponding to the framed region illustrated in Figure 1. One foreground seed is specified inside the heart region, while the background seeds are specified by a yellow polygon around the heart. For both the initialization procedure (shown in Figure 12(b)) and the two-stage algorithm (shown in Figure 12(c)), the foreground seed is the same, while the image boundary is specified as background.

Even with plenty of background seeds, the traditional RW algorithm still can not produce satisfying result, and some background regions are misclassified as foreground. On the other hand, the extracted foreground by the two-stage algorithm fits the heart quite well. Note that the initialization procedure has computed a rough shape of the heart, as shown in Figure 12(b). However, the zigzagged boundary as well as small holes inside the heart make it clear that the initialization procedure itself is not enough for accurate segmentation, and the PR correction procedure is indispensable.

Another interesting aspect of the two-stage algorithm is that it even does not need user-specified background seeds, which greatly simplifies user interactions. Please note that

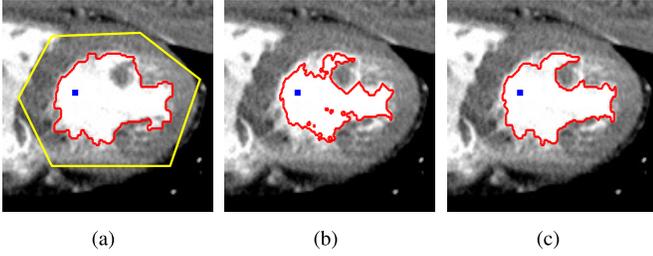


Fig. 12. Comparison of the traditional RW algorithm against the two-stage algorithm. (a) The result of traditional RW algorithm; (b) the result of initialization procedure; (c) the result of proposed two-stage algorithm.

the yellow polygon provides much stronger prior than the image boundary. Experiments show that for the traditional RW algorithm, one cannot just mark the image boundary as background seeds.

2) *Tests on the Rock CT Image*: Random walk algorithms allow interactive segmentation. Users might firstly mark some foreground seeds and background seeds, run the RW algorithm and then examine the results. If not satisfied, users could mark additional seeds and run the algorithm again.

The interactive segmentation process is demonstrated in Figure 13 for the traditional RW algorithm. Figure 13(a) shows the rock image, where the framed region will be zoomed-in for displaying the results better. As shown in Figure 13(b), two foreground seeds are marked in blue, while background seeds are specified by the yellow box. The segmentation result seems good except at the middle valley. Suppose that we want to separate the stone along the middle valley, then additional background seeds are marked as shown in Figure 13(c). The result, however, is still not satisfying. By adding more foreground as well as background seeds along the valley, the final result shown in Figure 13(d) seems quite good. As a comparison, the interactive segmentation process of the two-stage algorithm is shown in Figure 14. With same foreground and background seeds as shown in Figure 13(b), the segmentation result for the initialization procedure is shown in Figure 14(a). As one can see, the extracted boundary along the valley fits the stone boundary well. Then by marking additional background seeds in just one place, as shown in Figure 14(b), the two-stage algorithm already produces satisfying result.

This experiment suggests that the two-stage algorithm usually needs less user interactions than the traditional RW algorithm to achieve similar segmentation result.

### C. Comparison With State-of-the-Art Methods

The LRW [40], PARW [46], [47] and SMRW [51] algorithms are more recent developments of the basic idea of random walk. As stated previously, similar linear systems are also involved with these algorithms. So, the proposed two-stage algorithm could have been applied to reduce the computational complexity of these algorithms. In the following experiments, we will show that the above mentioned algorithms still suffer from the seeds sensitivity and boundary leakage problems, and the proposed CRWCR model overcomes this limitation.

The LRW and PARM algorithms are implemented by ourselves, while the implementation for the SMRW algorithm

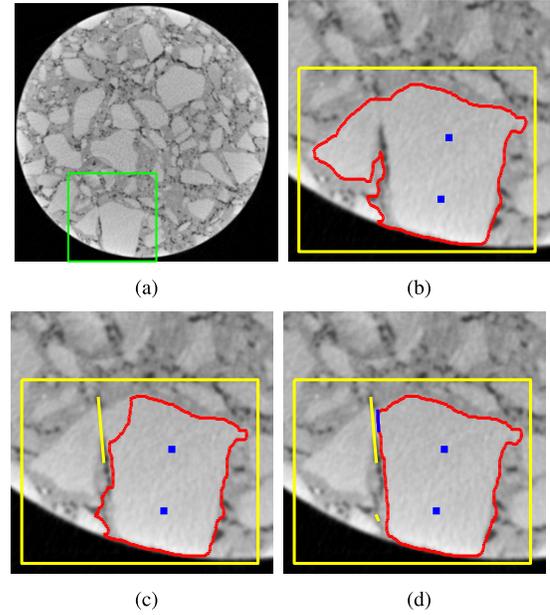


Fig. 13. Interactive image segmentation by the traditional RW algorithm. (a) the rock image with a framed area for zoomed-in display; (b) the first interactive segmentation result; (c) the segmentation result with added background seeds; (d) the final segmentation result with added foreground and background seeds.

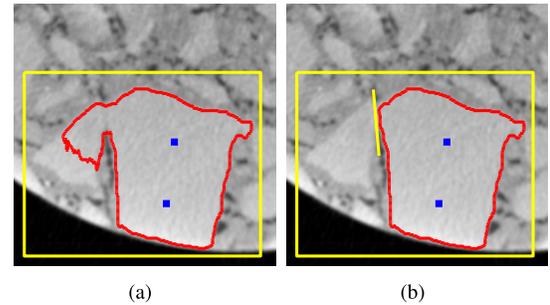


Fig. 14. Interactive image segmentation by the proposed two-stage algorithm. (a) the result of initialization; (b) the result of the two-stage algorithm.

is downloaded from <https://github.com/shenjianbing/Sub-Markov-Random-Walk-for-Image-Segmentation->.

The LRW algorithm is parameterized by a variable  $\alpha$  which denotes the probability of leaving the vertices. The labeling for a node is determined by its commute time to the seeded nodes of different classes, e.g. foreground and background. Let the Laplace matrix  $L = D - W$ , where  $D$  is the diagonal matrix formed by the main diagonal of  $L$ . Then the coefficient matrix of the linear systems that need to be solved for LRW is  $(I - \alpha S)$ , where  $I$  denotes the identity matrix, and  $S = D^{-1/2} W D^{-1/2}$ . In the following experiments, the parameters are set as  $\alpha = 0.9998$  according to the suggestions in [40]. The PARW algorithm solves linear systems with the coefficient matrix  $\Lambda^{-1}(\Lambda + L)$ , where  $\Lambda$  is a diagonal matrix. In the following experiments, we set  $\Lambda = \theta I$  as done in [47], in which case, the coefficient matrix reduces to  $(\theta I + L)$ , where  $\theta$  performs as a regularization parameter. For the SMRW algorithm, the parameters are set as  $c = 0.0004$ ,  $\sigma_c = 60$ ,  $\lambda = 2 * 10^{-10}$ . These parameters have subjected to fine tuning.

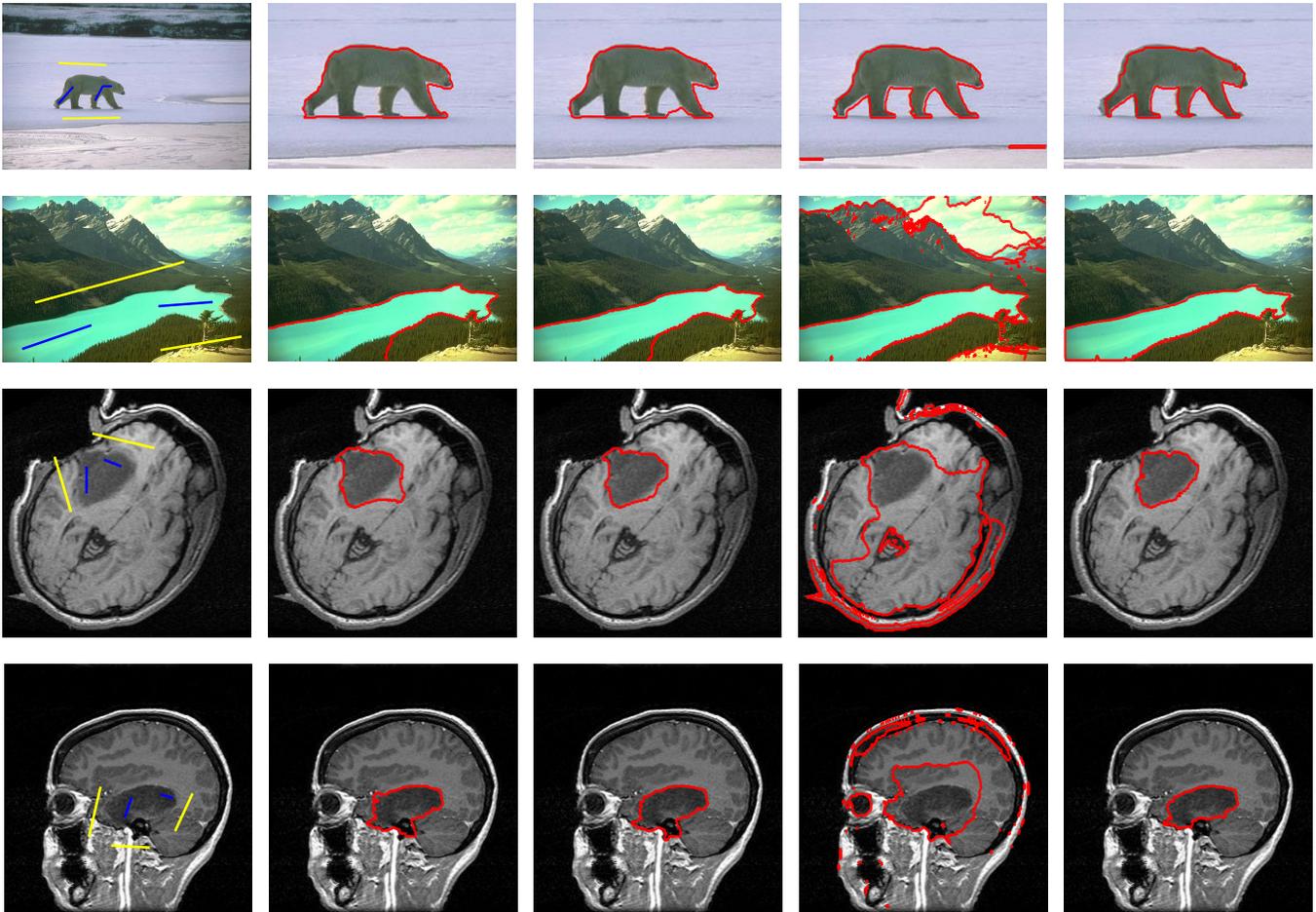


Fig. 15. First column: tested images with marked seeds; Second column: results of LRW; Third column: results of PARW with fine-tuned parameter  $\theta$ : 0.001(bear), 0.00001(river), 0.0001(brain tumor); Fourth column: SMRW; Last column: results of CRWCR.

The experiments are carried out on four images from public image data set, i.e. the Berkeley segmentation data set (the first two images of the first column, i.e. the bear and river, <https://www2.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/>) and the Brain Tumor Segmentation data set (the last two images of the first column, <http://spl.harvard.edu/publications/bitstream/download/5270>).

The results are demonstrated in Figure 15. The first column shows the tested images, as well as the foreground seeds (marked as blue) and the background seeds (marked as yellow). The second to fifth columns show the segmentation results of LRW, PARW, SMRW and the proposed CRWCR model ( $\gamma = 0.0001$ , implemented with the two-stage algorithm), respectively. For the bear image, the LRW and PARW fail to capture part of the boundary, while the SMRW algorithm and the CRWCR model successfully detect the bear from the context. The SMRW algorithm even captures the boundary better at the head and the rear sections. For the river image, the LRW and PARW algorithms still fail to capture some boundary sections, while both the SMRW algorithm and the CRWCR model capture the river's boundary quite well. However, the SMRW algorithm also captures other boundaries which don't belong to the ROI. This might be because the label priors (learned density distributions utilizing Gaussian Mixture

Models) built into the SMRW algorithm could not separate the foreground and background well. For tumor segmentation, the LRW and PARW algorithms demonstrate obvious boundary leakage by including part of background regions into the ROI, while the SMRW algorithm produces meaningless segmentation. On the other hand, the proposed CRWCR model always produces reliable and satisfactory results.

## VII. REMARKS AND CONCLUSIONS

We proposed a continuous random walk model with coherence regularization (CRWCR), and an efficient two-stage algorithm for solving it. Various numerical experiments against the traditional RW algorithm and the state-of-the-art algorithms show that the proposed CRWCR model could successfully reduce seeds sensitivity (i.e. user interactions).

The proposed two-stage algorithm could help to remove the difficulty of solving huge linear systems involved with the traditional RW algorithm as well as state-of-the-art random walk based algorithms including LRW, PARW and SMRW. Both procedures of the two-stage algorithm are very efficient since they fit well for parallel computing and usually very few iterations are needed. According to our experiments, for an  $1024 \times 1024$  image, running the two-stage algorithm once (with CUDA acceleration) needs about 0.1 seconds on a laptop with

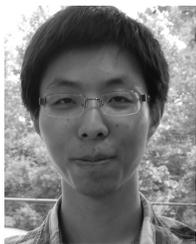
Intel i7-6700HQ CPU and GTX965M GPU. The source code (C++ with CUDA acceleration) for implementing our algorithm shall be available at <https://github.com/feifeizuo/crwcr>.

The proposed two-stage algorithm can be easily extended to deal with volume images. In this case, the PR scheme needs to be replaced by other semi-implicit schemes, since the PR scheme are not consistent in 3D domain. One can also apply the ADMM [56] framework on the energy functional of the CRWCR model to develop efficient algorithms. This will be investigated in our future work.

## REFERENCES

- [1] X.-C. Tai and C.-H. Yao, "Image segmentation by piecewise constant Mumford-Shah model without estimating the constants," *J. Comput. Math.*, vol. 24, no. 3, pp. 435–443, 2006.
- [2] X.-F. Wang, D.-S. Huang, and H. Xu, "An efficient local Chan–Vese model for image segmentation," *Pattern Recognit.*, vol. 43, no. 3, pp. 603–618, 2010.
- [3] P. Brault and A. Mohammad-Djafari, "Bayesian segmentation and motion estimation in video sequences using a Markov–Potts model," in *Proc. 5th WSEAS Int. Conf. Appl. Math.*, Stevens Point, Wisconsin, USA: World Scientific and Engineering Academy and Society, 2004, pp. 11:1–11:6. [Online]. Available: <http://dl.acm.org/citation.cfm?id=1378446.1378457>
- [4] M. Gastaud, M. Barlaud, and G. Aubert, "Combining shape prior and statistical features for active contour segmentation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 5, pp. 726–734, May 2004.
- [5] S. Kichenassamy, A. Kumar, P. Olver, A. Tannenbaum, and A. Yezzi, "Gradient flows and geometric active contour models," in *Proc. IEEE Int. Conf. Comput. Vis.*, Jun. 1995, pp. 810–815.
- [6] T. Pock, A. Chambolle, D. Cremers, and H. Bischof, "A convex relaxation approach for computing minimal partitions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 810–817.
- [7] A. Chambolle, D. Cremers, and T. Pock, "A convex approach to minimal partitions," *SIAM J. Imag. Sci.*, vol. 5, no. 4, pp. 1113–1158, 2012.
- [8] C. Zach, C. Häne, and M. Pollefeys, "What is optimized in convex relaxations for multilabel problems: Connecting discrete and continuously inspired MAP inference," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 1, pp. 157–170, Jan. 2014.
- [9] J. Lellmann, J. Kappes, J. Yuan, F. Becker, and C. Schnörr, "Convex multi-class image labeling by simplex-constrained total variation," in *Proc. SSMV*, vol. 5567, 2009, pp. 150–162.
- [10] J. Lellmann and C. Schnörr, "Continuous multiclass labeling approaches and algorithms," *SIAM J. Imag. Sci.*, vol. 4, no. 4, pp. 1049–1096, 2011.
- [11] S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton–Jacobi formulations," *J. Comput. Phys.*, vol. 79, no. 1, pp. 12–49, 1988.
- [12] J. Lie, M. Lysaker, and X.-C. Tai, "A binary level set model and some applications to Mumford–Shah image segmentation," *IEEE Trans. Image Process.*, vol. 15, no. 5, pp. 1171–1181, May 2006.
- [13] X.-C. Tai and C. Yao, "Fast piecewise constant level set methods (PCLSM) with Newton updating," in *Proc. UCLA CAM*, 2005, pp. 5–52.
- [14] C. Li, R. Huang, Z. Ding, J. C. Gatenby, D. N. Metaxas, and J. C. Gore, "A level set method for image segmentation in the presence of intensity inhomogeneities with application to MRI," *IEEE Trans. Image Process.*, vol. 20, no. 7, pp. 2007–2016, Jul. 2011.
- [15] J. Lie, M. Lysaker, and X.-C. Tai, "A variant of the level set method and applications to image segmentation," *Math. Comput.*, vol. 75, no. 255, pp. 1155–1174, 2006.
- [16] Z. Ma, J. M. R. Tavares, R. N. Jorge, and T. Mascarenhas, "A review of algorithms for medical image segmentation and their applications to the female pelvic cavity," *Comput. Methods Biomechanics Biomed. Eng.*, vol. 13, no. 2, pp. 235–246, 2010.
- [17] L. R. J. Ford and D. R. Fulkerson, "Maximal flow through a network," *Canadian J. Math.*, vol. 8, no. 3, pp. 399–404, 1956.
- [18] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images," in *Proc. 8th IEEE Int. Conf. Comput. Vis. (ICCV)*, vol. 1, Jul. 2001, pp. 105–112.
- [19] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1222–1239, Nov. 2001.
- [20] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1124–1137, Sep. 2004.
- [21] C.-W. Tan and A. Kumar, "Efficient iris segmentation using grow-cut algorithm for remotely acquired iris images," in *Proc. IEEE 5th Int. Conf. Biometrics, Theory, Appl. Syst.*, Sep. 2012, pp. 99–104.
- [22] Y. Cao, L. Ju, and S. Wang, "Grain segmentation of 3D superalloy images using multichannel EWCVT under human annotation constraints," in *Computer Vision—ECCV*. Springer, 2012, pp. 244–257.
- [23] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.
- [24] L. Grady, T. Schiwietz, S. Aharon, and R. Westermann, "Random walks for interactive organ segmentation in two and three dimensions: Implementation and validation," in *Medical Image Computing and Computer-Assisted Intervention—MICCAI*. Springer, 2005, pp. 773–780.
- [25] V. Vineet and P. J. Narayanan, "CUDA cuts: Fast graph cuts on the GPU," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2008, pp. 1–8.
- [26] Y. Peng, L. Chen, F.-X. Ou-Yang, W. Chen, and J.-H. Yong, "JF-Cut: A parallel graph cut approach for large-scale image and video," *IEEE Trans. Image Process.*, vol. 24, no. 2, pp. 655–666, Feb. 2015.
- [27] B. Appleton and H. Talbot, "Globally minimal surfaces by continuous maximal flows," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 1, pp. 106–118, Jan. 2006.
- [28] J. Yuan, E. Bae, X.-C. Tai, and Y. Boykov, "A continuous max-flow approach to Potts model," in *Computer Vision—ECCV*. Springer, 2010, pp. 379–392.
- [29] L. Grady, "Random walks for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 11, pp. 1768–1783, Nov. 2006.
- [30] C. Couprie, L. Grady, L. Najman, and H. Talbot, "Power watersheds: A new image segmentation framework extending graph cuts, random walker and optimal spanning forest," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep./Oct. 2009, pp. 731–738.
- [31] F. Wang and C. Zhang, "Label Propagation through linear neighborhoods," *IEEE Trans. Knowl. Data Eng.*, vol. 20, no. 1, pp. 55–67, Jan. 2008.
- [32] J. Zhang, J. Zheng, and J. Cai, "A diffusion approach to seeded image segmentation," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 2125–2132.
- [33] B. Ham, D. Min, and K. Sohn, "A generalized random walk with restart and its application in depth up-sampling and interactive segmentation," *IEEE Trans. Image Process.*, vol. 22, no. 7, pp. 2574–2588, Jul. 2013.
- [34] C. G. Bampis, P. Maragos, and A. C. Bovik, "Graph-driven diffusion and random walk schemes for image segmentation," *IEEE Trans. Image Process.*, vol. 26, no. 1, pp. 35–50, Jan. 2017.
- [35] J. Pang and G. Cheung, "Graph Laplacian regularization for image denoising: Analysis in the continuous domain," *IEEE Trans. Image Process.*, vol. 26, no. 4, pp. 1770–1785, Apr. 2017.
- [36] L. Grady, T. Tassdizen, and R. Whitaker, "A geometric multigrid approach to solving the 2D inhomogeneous Laplace equation with internal Dirichlet boundary conditions," in *Proc. IEEE Int. Conf. Image Process.*, vol. 2, Sep. 2005, pp. 642–645.
- [37] L. Grady, "A lattice-preserving multigrid method for solving the inhomogeneous Poisson equations used in image analysis," in *Computer Vision—ECCV*. Springer, 2008, pp. 252–264.
- [38] T. H. Xin Wang, "Efficient multigrid solver for the 3D random walker algorithm," *Proc. SPIE*, vol. 7259, no. 35, pp. 3R-1–3R-8, 2009.
- [39] A. Fabijańska and J. Goctawski, "The segmentation of 3D images using the random walking technique on a randomly created image adjacency graph," *IEEE Trans. Image Process.*, vol. 24, no. 2, pp. 524–537, Feb. 2015.
- [40] J. Shen, Y. Du, W. Wang, and X. Li, "Lazy random walks for superpixel segmentation," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1451–1462, Apr. 2014.
- [41] J. Málek and Z. Strakos, *Preconditioning and the Conjugate Gradient Method in the Context of Solving PDEs*, vol. 1. Philadelphia, PA, USA: SIAM, 2015.
- [42] E. Isaacson and H. B. Keller, *Analysis of Numerical Methods*. North Chelmsford, MA, USA: Courier Corp., 2012.
- [43] J. Stoer and R. Bulirsch, *Introduction to Numerical Analysis*. vol. 12. Springer Science & Business Media, 2013.
- [44] D. Zhou and B. Schölkopf, "Learning from labeled and unlabeled data using random walks," in *Proc. 26th DAGM Symp. Pattern Recognit.*, vol. 3175. Springer Science & Business Media, Aug./Sep. 2004, pp. 237–244.

- [45] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, pp. 321–328.
- [46] X.-M. Wu, Z. Li, A. M. So, J. Wright, and S.-F. Chang, "Learning with partially absorbing random walks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 3077–3085.
- [47] Y. Liang, J. Shen, X. Dong, H. Sun, and X. Li, "Video supervoxels using partially absorbing random walks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 5, pp. 928–938, May 2016.
- [48] W. Wang, J. Shen, R. Yang, and F. Porikli, "Saliency-aware video object segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 1, pp. 20–33, Jan. 2018.
- [49] W. Wang, J. Shen, and L. Shao, "Video salient object detection via fully convolutional networks," *IEEE Trans. Image Process.*, vol. 27, no. 1, pp. 38–49, Jan. 2018.
- [50] W. Wang, J. Shen, F. Porikli, and R. Yang, "Semi-supervised video object segmentation with super-trajectories," *IEEE Trans. Pattern Anal. Mach. Intell.*, to be published, doi: [10.1109/TPAMI.2018.2819173](https://doi.org/10.1109/TPAMI.2018.2819173).
- [51] X. Dong, J. B. Shen, L. Shao, and L. Van Gool, "Sub-Markov random walk for image segmentation," *IEEE Trans. Image Process.*, vol. 25, no. 2, pp. 516–527, Feb. 2016.
- [52] L. Grady, "Multilabel random walker image segmentation using prior models," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, May 2005, pp. 763–770.
- [53] J. Dodziuk, "Difference equations, isoperimetric inequality and transience of certain random walks," *Trans. Amer. Math. Soc.*, vol. 284, no. 2, pp. 787–794, 1984.
- [54] J. Shen, Y. Du, and X. Li, "Interactive segmentation using constrained Laplacian optimization," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 7, pp. 1088–1100, Jul. 2014.
- [55] Y. Zhang, J. Cohen, and J. D. Owens, "Fast tridiagonal solvers on the GPU," *ACM SIGPLAN Notices*, vol. 45, no. 5, pp. 127–136, 2010.
- [56] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.



**Mengfei Li** received the master's and Ph.D. degrees from the School of Mathematical Sciences, Capital Normal University, Beijing, China, in 2013 and 2017, respectively. His main research interests include image processing, such as denoising and segmentation, and CT image reconstruction, such as dual-energy CT reconstruction and low-dose reconstruction.



**Hong Gao** received the B.S. degree in applied mathematics from Hebei Agricultural University, Hebei, China, in 2012, and the master's degree in mathematics and information technology from Capital Normal University, Beijing, China, in 2015. Her research interests include image segmentation.



**Feifei Zuo** received the master's degree from the school of Capital Normal University, Beijing, China, in 2016. After graduating from CNU, he worked in the medical-imaging field at LargeV Instrument Corporation Ltd., Beijing. During his time there, he was responsible for several medical software developments. His primary research interest is image segmentation and CT data visualization. His current research encompasses computer graphics, mesh model segmentation, and image processing.



**Hongwei Li** received the Ph.D. degree in parallel computing from the Institute of Software, Chinese Academy of Sciences, in 2002. From 2002 to 2005, he was with the Institute of Software as an Assistant Professor. In 2005, he joined the Center for Integrated Petroleum Research, University of Bergen, Norway, as a Post-Doctoral Researcher for image processing and inverse problem computing. Since 2008, he has been with the Department of Mathematics, Capital Normal University, Beijing, as an Associate Professor. His research is highly application oriented. Algorithms and methods for image enhancement, segmentation, and artifacts reduction developed by him have been utilized by CT equipment manufactures for improving image qualities. He is also deeply involved with the CT reconstruction community, and part of his research interest has been directed to develop reconstruction algorithms for dealing with challenging CT applications, such as low-dose CT and limited angle CT. His research interests have been focused on image processing and computed tomography after joining the Capital Normal University.