Colonic fold detection from computed tomographic colonography images using diffusion-FCM and level sets

Ananda S. Chowdhury 1, Sovira Tan, Jianhua Yao, Ronald M. Summers *

Radiology and Imaging Sciences Department, National Institutes of Health Clinical Center, Building 10, Room 1C368X, Bethesda, MD 20892–1182, USA

A R T I C L E   I N F O

Article history:
Received 18 May 2009
Received in revised form 9 December 2009
Available online 15 January 2010
Communicated by F.Y. Shih

Keywords:
Heat diffusion
Fuzzy c-means
Level sets
Shape index
Computed tomographic colonography

A B S T R A C T

Colon cancer is the second major cause of cancer related deaths in industrial nations. Computed tomographic colonography (CTC) has emerged in the last decade as a new less invasive colon diagnostic alternative to the usually practiced optical colonoscopy. The overall goal is to increase the effectiveness of virtual endoscopic navigation of the existing computer-aided detection (CAD) system. The colonic/hausstral folds serve as important landmarks for various associated tasks in the virtual endoscopic navigation like prone–supine registration, colonic polyp detection and tenia coli extraction. In this paper, we present two different techniques, first in isolation and then in synergism, for the detection of hastral folds. Our input is volumetric computed tomographic colonography (CTC) images. The first method, which uses a combination of heat diffusion and fuzzy c-means algorithm (FCM), has a tendency of over-segmentation. The second method, which employs level sets, suffers from under-segmentation. A synergistic combination, where the output of the first is used as an input for the second, is shown to improve the segmentation quality. Experimental results are presented on digital colon phantoms as well as real patient scans. The combined method has a total erroneous (over-segmentation plus under-segmentation) detection of (6.5 ± 2)\% of the total number of folds per colon as compared to (12.5 ± 5)\% for the diffusion-FCM-based method and (11.5 ± 3)\% for the level set-based method. The p-values obtained from the associated ANOVA tests indicate that the performance improvements are statistically significant.

Published by Elsevier B.V.

1. Introduction

Colon cancer is the second major cause of cancer related deaths in industrial nations (Potter et al., 1993). Colorectal polyps are precursors of colon cancer. For colorectal polyp and colon cancer screening, Computed tomographic colonography (CTC), a minimally invasive technique, has become a potential alternative to the usually practiced optical colonoscopy (Johnson and Dachman, 2000). The goal of CTC is to achieve a high patient acceptance, diagnostic accuracy and screening effectiveness thereby decreasing mortality rates due to colon cancer (Frentz and Summers, 2006). Computer-aided detection (CAD) for CTC is a computerized scheme that automatically detects colonic polyps and reveals their locations to radiologists (Yoshida and Dachman, 2005). As an integral component of the CAD system, advanced imaging techniques are employed to visualize 3-dimensional endoscopic flight paths through the inside of the colon (Kuwayama et al., 2002). Proper identification of the colonic/hausstral folds is essential for many important tasks like prone–supine registration (Nain et al., 2002), better colonic polyp detection (Yao and Summers, 2007) and tenia coli extraction (Chowdhury et al., 2008). Improvement of the above tasks will increase the effectiveness of virtual endoscopic navigation in the existing CAD system. In case of prone–supine registration, the extracted fold information can be incorporated in the objective function of a dynamic programming-based approach. As far as colonic polyp detection is concerned, fold enumeration may lead to precise localization. For example, one can specify the two consecutive folds containing a polyp. Teniae coli are longitudinal muscles that can serve as ideal references for guiding virtual navigation and polyp registration. The detection of folds help in detection of teniae as the teniae are a flat band perpendicular to the folds and they interrupt the folds. The haustra of the colon are the small pouches caused by sacculation, which give the colon its segmented appearance. The tenia coli runs the length of the large intestine. Because the tenia coli is shorter than the intestine, the colon becomes sacculated between the tenia, forming the haustra. In between adjacent haustra, lie the folds. Visually, the folds appear as indentations on the colon wall. It is relevant to mention that the folds are better visualized on colon surface, compared to volumetric computed tomography colonography (CTC) images. However, in this paper, we combined both volumetric CTC data...
(heat diffusion-FCM) and colon surface (level sets) for more faithful fold detection. We show two figures, Figs. 1 and 2, with folds on volumetric CTC data and on colon surface respectively.

There is a relative paucity of literature on haustral fold detection. The existing techniques have some shortcomings and/or lack detailed validations. Current methods of haustral fold detection include threshold-based curvature filter (Huang et al., 2005) and computation of eigenvalues of the Hessian matrix using local intensity structure information (Oda et al., 2008). The threshold-based curvature filter is applied on the colon surface and is found to miss some folds. In contrast, the computation of the eigenvalues of Hessian matrix is done on volumetric computed tomography colonography (CTC) images. A typical disadvantage of the Hessian-based approach is that they are highly sensitive to noise due to the second-order derivatives (Truc et al., 2008). A related research work on probabilistic modeling of haustral curvatures from volumetric CTC data employs relatively complex methods like stochastic differential equation, Gaussian random fields along with computation of the eigenvalues of the Hessian matrix (Melonakos et al., 2007). In this paper, we present two different methods for haustral fold detection using both colon volume as well as colon surface, first in isolation and next in a synergistic manner. The first method employs a combination of heat diffusion and fuzzy c-means (FCM) clustering on volumetric CTC images. The folds, detected on the volume, are then mapped on the colon surface. The second method, based on level sets, detects the folds directly on the colon surface. We notice that both of these methods have some limitations. The method based on heat diffusion and FCM has a tendency of over-segmentation while the one using level sets suffers from under-segmentation. So, we propose a synergistic combination of the above methods for enhancing the performance of fold segmentation. We present the results on digital phantoms of human colons as well as real patient scans. Thorough qualitative and quantitative performance evaluations are provided. Apart from addressing a problem of substantial clinical significance, our work has some contributions from a purely pattern recognition perspective. Level set segmentation on a 3D surface represented by mesh instead of the rectangular grid is relatively new. Our work shows that it is a promising general technique that can solve other surface segmentation problems with an appropriately designed speed function. To the best of our knowledge, synergism of heat diffusion-FCM and level sets has not been applied before to solve any problem in the area of pattern recognition. A preliminary version of the paper appeared in (Chowdhury et al., 2009).

The paper is divided as follows: in Section 2, we briefly state the image preprocessing tasks undertaken to segment the colon from CTC images. In Section 3, we describe heat diffusion and FCM clustering to segment the haustral folds. In Section 4, we discuss the level set technique for detection of the same haustral folds. In Section 5, we present a synergistic combination of the diffusion-FCM and level set method, which can improve the performance of fold detection. In Section 6, we discuss datasets and control parameters. Section 7 illustrates experimental results on digital colon phantoms as well as several CTC datasets. Finally, we conclude the paper in Section 8 with an outline of the directions for future research.

2. Preprocessing

The colon lumen is first segmented from the input volumetric computed tomographic virtual colonoscopy images. The segmentation is performed using modified region growing and fuzzy connectedness (Franaszek et al., 2006). The segmented colon is further processed to extract its surface using the Marching Cubes algorithm (Lorensen and Cline, 1987). The centerline, which serves as the approximate axis of the colon, is extracted using level sets during the same pre-processing phase (Vanlith et al., Summers, 2007). The centerline in this work is used for flattening a colon. Folds, detected on the original colon surface, are better visualized on the corresponding flattened versions.

3. Fold detection with heat diffusion and FCM

In this section, we describe a method of fold detection using heat diffusion and FCM clustering algorithm.

3.1. The heat diffusion

Every voxel inside the segmented colon is treated as a ‘hot’ voxel and every voxel outside the segmented colon is treated as a ‘cold’ voxel (see Fig. 3A). Thus, the segmented colon is set to a constant temperature $t_b = 1$ and is then allowed to ‘cool down’ (Konukoglu and Acer, 2007). The heat diffusion equation is given by:

$$\partial \theta(r,t)/\partial t = \nabla \cdot D \nabla \theta(r,t),$$

where $\theta(r,t)$ represents the temperature at position $r = (x,y,z)$ and time $t$. $D$ is called the diffusion coefficient. The temperature diffuses
Throughout the colon boundary and fills in the fold spaces after certain time. Perona–Malik’s gradient anisotropic diffusion is employed (Perona and Malik, 1990) to generate such candidate voxels for folds, which also includes voxels surrounding the edges. The rationale behind choosing gradient anisotropic diffusion over a standard distance transform (e.g. (Danielsson, 1980)) to generate the desired set of voxels for folds are to ensure some noise removal and preserve the fold structures better.

3.2. FCM clustering

The generated diffusion map identifies a set of voxels as candidates for folds. However, the candidate set contains some non-fold voxels also. This calls for applying the fuzzy c-means (FCM) algorithm (Bezdek et al., 1999) to classify the set of voxels into folds and non-folds. The classification for Q such voxels is done in the two-dimensional feature space \( R^2 \), constituted by the number of ‘hot’ (having temperature > 0.9) and the number of ‘cold’ (having temperature < 0.0001) neighbors (within a spherical neighborhood of radius 6 voxels centered at the candidate voxel under consideration). Number of clusters \( C \) for the present problem is 2: one corresponds to the fold and the other to the non-fold. Let \( u_{ij} \in [0,1] \) be the membership of a voxel \( v_j \) to belong to cluster \( c_i \). Then, \( u_{ij} \) is given by Thanapong et al. (2007):

\[
J_m(U,V) = \frac{1}{C} \sum_{i=1}^{C} \left( \frac{1}{Q} \sum_{j=1}^{Q} \left( u_{ij} - c_i \right)^m \right)^{-\frac{1}{m}},
\]

where \( 1 < i < C \) and \( 1 < j < Q \). Let \( U \) be the fuzzy partition matrix and \( V \) be the cluster center vector. Then the objective function, which is minimized in the FCM algorithm, is given by:

\[
J_m(U,V) = \frac{1}{C} \sum_{i=1}^{C} \left( \frac{1}{Q} \sum_{j=1}^{Q} \left( u_{ij} - c_i \right)^m \right)^{-\frac{1}{m}}.
\]

The weighting exponent \( m \) is chosen as 2. The FCM-based classification of any such candidate voxel to ‘fold’ voxel and ‘non-fold’ (edge and near edge) voxel essentially depends on spatial distribution of ‘hot’ and ‘cold’ voxels within the above spherical neighborhood, centered on the particular candidate voxel under consideration. From Fig. 3A and B, it can be seen that within the above-mentioned spherical zone, more neighbors of the internal fold voxels are hot than cold and more neighbors of the non-fold voxels are cold than hot. Hence, the classification works correctly with well separated fold and non-fold cluster centers. Table 1 shows different parameters used in the heat diffusion and the FCM algorithm.

<table>
<thead>
<tr>
<th>Method</th>
<th>Conductance parameter</th>
<th>No. of iterations</th>
<th>Time step</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diffusion</td>
<td>3.0</td>
<td>15</td>
<td>0.0625</td>
</tr>
<tr>
<td>FCM clustering</td>
<td>Convergence criterion: 0.1</td>
<td>3–4</td>
<td></td>
</tr>
<tr>
<td>Level sets</td>
<td>( \alpha = 1.0, \beta = 1.0, \gamma = 2.0 ) (Eq. (14))</td>
<td>50</td>
<td></td>
</tr>
<tr>
<td></td>
<td>( \mu = 0.2, \nu = 0.1 ) (Eq. (15))</td>
<td>15</td>
<td></td>
</tr>
</tbody>
</table>

Table 1 Parameters for different methods.

4. Fold detection using level sets

In this section, we describe the level set-based fold detection method. Level sets have so far mainly been implemented on rectangular grids (Sethian, 1999). In the present work, we implement them on a triangular mesh that represents the surface of the colon following the method outlined in (Tan et al., 2008). In the work described in (Tan et al., 2008), the mesh level set was used to extract the ridgelines of vertebral bodies. In order to adapt it to the problem of segmenting folds from colonic walls, we introduce important innovations concerning the speed function and the timestep. To implement level sets on a mesh, two important adjustments (relative to level sets in rectangular grids) have to be made:

1. Gradients and curvatures have to be computed in local coordinate systems defined around each vertex (small enough neighborhoods can reasonably be considered planar).

Fig. 3. Fold detection using diffusion-FCM. A single CT colonography slice through the transverse colon is shown. (A) Segmented colon. (B) Diffusion map (voxels in gray represent candidates for folds). (C) Fuzzy c-means clustering. (D) Label filtering and thresholding of fuzzy membership values (white arrow points to a colonic fold).
(2) Gradients and curvatures have to be computed using least square estimation methods rather than finite differences.

In the rest of this section we use the following definitions and notations. A function \( f(V) \) defined on a mesh associates to each vertex \( V \) the quantity \( f(V) \). A vertex \( V \) is defined by its three coordinates \((x, y, z)\) which can be relative to a global or a local orthonormal frame. \( V \) can therefore also be seen as a vector. By immediate neighbor of vertex \( V \), we mean a vertex linked to \( V \) by an edge. The 1-ring neighborhood of \( V \) is the set of immediate neighbors of \( V \). The 2-ring neighborhood of \( V \) consists of its 1-ring neighborhood and all the immediate neighbors of the vertices in the 1-ring neighborhood. The process can be iterated. Thus, the \( n \)-ring neighborhood of \( V \) is comprised of its \((n - 1)\)-ring neighborhood and all the immediate neighbors of the vertices in the \((n - 1)\)-ring neighborhood.

4.1. Background on level sets

Level sets are evolving contours that can expand, contract and even split or merge (Sethian, 1999). For the purpose of segmentation, they are designed to deform so as to match an object of interest and stop at its boundaries. The level set we chose to implement is the geodesic active contour (GAC) which evolves according to (Caselles et al., 1997):

\[
\frac{d}{dt} \psi(x, t) = \alpha g_c \nabla \psi + \beta g_k \nabla \psi + \gamma \nabla g \nabla \psi.
\]  

(4)

The evolving contour is encoded as the zero level set of the distance function \( \psi(x, t) \). In other words, points that verify \( \psi(x, t) = 0 \) form the contour. By convention, the distance is negative for points inside the contour and positive for the ones outside:

\[
\psi(x, t) = \pm \delta,
\]  

(5)

where \( \delta \) is the distance from point \( x \) to the zero level set contour. The first term on the right-hand side of the Eq. (4) is the propagation term that makes the contour move with velocity \( c \). The second is the curvature term which controls the smoothness of the contour using the mean curvature \( \kappa \). The third term, the advection term, was introduced in (Caselles et al., 1997) to lock the contour to the boundary. The parameters \( \alpha, \beta \) and \( \gamma \) weight the importance of each term.

The spatial function \( g \) derived from the image or surface to be segmented, encodes the information about the objects’ boundaries and is called the speed function. This function is crucial as it guides the contour. While in rectangular grids guiding features are mainly grey level edges, on 3D surfaces, features are based on curvature properties. Shape Index (SI) and Curvedness (CV) are two meaningful ways of quantifying curvature properties (Koenderink, 1990). At a local level on a surface, SI quantifies the nature of the shape (cap, ridge, saddle, rut, and cup) while CV quantifies the degree of curvature.

4.2. Numerical implementation of the level set evolution on a 3D mesh

The level set evolves as the results of updating Eq. (4). A Runge–Kutta procedure (Press et al., 2007) is used for that purpose. At each vertex \( V \) the distance function is updated as follows:

\[
\psi_{t+1} = \psi_t + (\alpha g_c \nabla \psi_t + \beta g_k \nabla \psi_t + \gamma \nabla g \nabla \psi_t) \Delta t,
\]  

(6)

where \( \Delta t \) is the timestep. Its computation is detailed below. The gradients of the functions \( \psi \) and \( g \) have to be evaluated. We do this locally on the mesh in a 1-ring neighborhood around each vertex. The coordinates of \( \nabla f(V) \) the gradient of any function \( f \) at vertex \( V \) can be found by minimizing:

\[
E = \sum_{i=1}^{N} (\nabla f(V) \cdot \hat{n}_i - \nabla f(V_i))^2.
\]  

(7)

The summation is over the \( N \) immediate neighbors of \( V \). The \( i \)th neighbor \( V_i \) of \( V \) defines the unit directional vector \( \hat{n}_i \):

\[
\hat{n}_i = \frac{V_i - V}{|V_i - V|}.
\]  

(8)

The quantity \( \nabla f(V_i) \) is the finite difference of function \( f \) in the direction of the \( i \)th neighbor \( V_i \):

\[
\nabla f(V_i) = \frac{f(V_i) - f(V)}{|V_i - V|}.
\]  

(9)

The gradients \( \nabla \psi \) and \( \nabla g \) in the Eq. (6) are computed in this fashion. \( \nabla \psi \) is then used to compute the mean curvature \( \kappa \).

The gradient \( \nabla \psi \) is used to form two functions on the mesh: \( \psi_x(V) \) and \( \psi_y(V) \), which respectively associate the \( x \) and \( y \) components of \( \nabla \psi \) to each vertex \( V \). We can then evaluate the gradients of \( \psi_x(V) \) and \( \psi_y(V) \) using Eqs. (7)–(9), which in turn yields \( \psi_x, \psi_y \) and \( \psi_{xy} \). Those are the quantities necessary to compute the mean curvature \( \kappa \) of the distance function \( \psi \):

\[
\kappa = \psi_{xx} \psi_y^2 - 2 \psi_x \psi_y \psi_{xy} + \psi_y \psi_{yy}^2 \left( \psi_x^2 + \psi_y^2 \right)^2.
\]  

(10)

More detailed practical information on the implementation of the level set on a mesh can be found in (Tan et al., 2008).

4.3. Timestep computation

Since the length of the edges can vary considerably on a mesh, a specific timestep is computed for each vertex on the contour. Note that in (Tan et al., 2008), the timestep was a fixed quantity. The level set locally advances in the direction normal to the contour. In the preceding subsection, we have presented the computation of gradient of the distance function \( \nabla \psi \). For each vertex \( V \) on the contour, this gradient, after normalization, can be taken as the local normal to the contour. We consider the set \( \{V\} \) of immediate neighbors of \( V \). Let \( V_n \) be the neighbor that lies in the direction closest to the normal at \( V \). We take the timestep \( \Delta t \) at the vertex \( V \) to be:

\[
\Delta t = \lambda |V_n - V|,
\]  

(11)

where \( \lambda \) is a proportionality constant, which was set to 1 here.

4.4. Speed function

In the present work, our strategy is to initiate the level set from a seed on the ridge of a haustural fold and stop it at the fold/wall boundary. To achieve this, we design an appropriate new speed function based on both SI and CV. It is a combination of two sigmoids:

\[
g(V) = g_{SI}(V)g_{CV}(V),
\]  

(12)

\( g(V) \) is the function which attributes a speed value at each vertex \( V \). The first sigmoid \( g_{SI}(V) \) can be written as:

\[
g_{SI}(V) = 1 - \frac{1}{1 + \exp \left( -\frac{\xi - \eta}{\delta} \right)},
\]  

(13)

where SI is the shape index at vertex \( V \). This function is widely used in grid level sets (Sethian, 1999). We have adapted it to mesh level sets by replacing gradient values by SI values. The guidelines for determining \( x \) and \( y \) are (Tan et al., 2008):
\[
\eta = \frac{SI_1 - SI_2}{6}, \quad \xi = \frac{SI_1 + SI_2}{2}.
\] (14)

Such a choice of \(\eta\) and \(\xi\) ensures that \(g_{\eta}(V)\) attributes (i) a speed value increasingly lower than 0.05 to SI values increasingly larger than \(SI_1\) and (ii) a speed value increasingly larger than 0.95 to SI values increasingly smaller than \(SI_2\). SI values fall in the range of 0–1, with the following correspondences between shapes and values (Koenderink, 1990):

\[
\text{[cap, ridge, saddle, rut, cup]} = [0, 0.25, 0.5, 0.75, 1].
\]

An appropriate choice for \(SI_1\) and \(SI_2\) will attribute high speed to fold vertices and low speed to wall vertices. The other sigmoid \(g_{CV}(V)\) reinforces the first one by adding another guiding force based on the curvedness feature. It can be written as:

\[
g_{CV}(V) = \frac{1}{1 + \exp \left( - \frac{CV - \mu}{\sigma} \right)},
\] (15)

where \(CV\) is the curvedness at vertex \(V\). The function for \(g_{CV}(V)\) in the Eq. (15) attributes higher speeds to \(CV\) values larger than \(\mu\) and lower speeds to those smaller than \(\mu\). The parameter \(\sigma\) controls the smoothness of the sigmoid function. Haustral fold ridges are more curved than the colon wall and thus have higher \(CV\) values.

4.5. Seeding

Level sets require an initial seed from which to evolve the contour. We perform that seeding by clustering fold vertices with a simple blob-labeling technique. The criteria for these fold vertices are: \(0.2 < SI < 0.55\) and \(CV > 0.3\). This SI range includes ridge and saddle because significant parts of a fold near the wall are saddle shaped. A size threshold (50 vertices) is then chosen to avoid putting seeds in noisy non-fold areas. The boundaries of those clusters are the initial contours from which our level set evolves.

5. Fold detection using heat diffusion-FCM and level set synergism

Each of the two methods described in Sections 3 and 4 have some limitations. The method using heat diffusion and FCM utilizes the shape of the segmented colon to identify the haustral folds. The technique entirely focuses on volumetric image information. It is observed that the method has a tendency of under-segmentation. A typical example is when two parts of the colon come very close due to its twisted shape. In such situations, some of the non-fold candidate voxels between the two parts, identified by the diffusion, will have a high ratio of hot neighbors and cold neighbors, very similar to that of a true fold voxel. Consequently, there is a possibility that the FCM algorithm would wrongly classify these non-fold voxels as fold voxels. For the level set method, which applies the shape index and the curvedness information in the speed function for the identification of true folds, this is not a potential problem as the above areas are not ridge-like. However, the level set method requires correct initial seeds to evolve the contour. The seeds are selected from prior clustering of ridge-like vertices based on a simple blob-labeling technique. A size threshold needs to be chosen here to avoid putting seeds in noisy non-fold areas. When this threshold is set so that few false folds are seeded, some true folds are also missed in the process, resulting in under-segmentation.

The rationale behind the synergism of the two proposed methods is to achieve correct segmentation by using the over-segmented output of the first method (i.e., diffusion-FCM) as an input to the second method (i.e., level sets), which suffers from potential under-segmentation. In particular, folds detected from the method with diffusion-FCM are used to find seeds for the one based on level set evolution. We first mark vertices, whose coordinates correspond to voxels segmented as folds, to be seeds. The vertices with low speed function are then removed from the already created seed list. This process corrects the over-segmentation and leaves the central part of the folds’ ridges as seeding regions. Level set evolution finally completes the segmentation. Note that the level set, by itself, relies on clustering for its seeds, which can cause under-segmentation as one fold might be divided into several clusters with size of all the clusters falling below the chosen size threshold (i.e., 50 vertices). When used with diffusion-FCM, all seeds from the diffusion-FCM are accepted without any clustering. False positives are eliminated because they have low speed.

6. Control parameters and datasets

In this section, we discuss the control parameters for the different methods described in the previous section. Next, we discuss the construction of digital colon phantoms, needed for a rigorous validation of the experimental results.

We have used ITK (Ibanez et al., 2003) for the implementation of Perona–Malik’s gradient anisotropic diffusion. All the parameters in Table 1 are experimentally chosen and their values are kept the same for all the experimental datasets. As far as the parameters for the level sets are concerned; the curvature weight \(\mu\) is reduced from 1.5 in (Tan et al., 2008) to 1 because a fold is longitudinal in shape. Higher curvature weight would force the contour to be more circular. Values of other parameters for the level sets are kept same as in (Tan et al., 2008).

In order to provide additional validation for the haustral fold detection methods, we generate digital phantoms of human colons. Without any loss of generality, the phantom is designed to resemble the transverse part of the human colon. A cylinder is used to simulate the transverse colon. Anatomical observations reveal that (i) the haustral folds are centered on different points on the axis (centerline) of the colon; (ii) each fold is circumferentially trisected by three teniae coli and (iii) the folds can vary in thickness and depth. An attempt has been made to conform to the above anatomical observations in the modeling process. Accordingly, a trisected torus (Wolfram, 1996) is used to model a colonic fold. The centers of these tori are chosen as different points lying on the axis of the cylinder. Non-uniformly spaced centers correspond to non-uniform spacing between the folds. The cross-sections of the simulated folds are made elliptical to independently vary their thickness and depth. More precisely, a simulated fold can be thin or thick depending upon the value of the minor axis of its elliptical cross-section. In contrast, a simulated fold can be deep or shallow depending upon the value of the major axis of its elliptical cross-section. In addition, use of different circumferential spans for different folds on the body of the colon automatically leads to simulation of curvilinear teniae. In order to study the impact of noise on the proposed methods; the phantom surfaces are contaminated with blobs of varying size. We randomly choose 20% of the surface voxels on the phantom surfaces as the centers of these blobs. The blobs are cubes with each dimension ranging from 2 to 4 voxels.

We test our algorithms on 4 sets of digital colon phantoms (with and without noise) and 18 real patient scans (with 9 sets of prone and 9 sets of supine data). The 18 datasets are randomly chosen from three different colonography databases and have different resolutions as well as image dimensions. The three databases used are from Bethesda Naval Hospital, Naval Medical Center San Diego and Walter Reed Army Medical Center (Summers and Yao, 2005).
7. Experimental results

Fig. 3A is a 2D slice of a segmented volume of a colon. It is evident from this figure that only some of these voxels, part of thin protruding structures into the segmented colon, actually constitute the folds. Fig. 3B shows the effect of anisotropic diffusion after 15 iterations on the 2D slice, shown in Fig. 3A. The voxels in gray in Fig. 3B represent the candidates for folds. Fig. 3C indicates the final fuzzy membership values of the previously identified voxels (in Fig. 3B) to belong to the cluster ‘fold’, after stable clustering is achieved. The higher the fuzzy membership value of a voxel to belong to the class ‘fold’, the brighter is the intensity of that voxel in this figure. Fig. 3D shows the effect of thresholding the fuzzy membership values followed by a label filtering operation. All the fold voxels are represented in gray on a white colon in Fig. 3D.

Table 2 shows the performance of the fold detection on the digital phantoms. Since, we have the precise ground-truth information for the folds in the phantoms; we use sensitivity and specificity (on vertices of the phantom surfaces) as the two performance metrics. The ‘combined’ method has a higher sensitivity compared to the one based on level sets and has a greater specificity compared to the one based on diffusion-FCM. Table 2 also indicates that the proposed methods are fairly robust to noise as the sensitivity and specificity values for the noisy phantoms only show a marginal decrease compared to the noiseless phantoms. In fact, the values show that the ‘combined’ method is more robust to noise compared to the other two methods. In Fig. 4A–D, we, respectively, show the ground-truth, and folds detected using diffusion-FCM, level sets and ‘combined’ methods on a noisy phantom surface. Folds in Fig. 4D appear to be closest to the folds in Fig. 4A, suggesting the superiority of the ‘combined’ method over the other two.

In Table 3, we quantitatively demonstrate the performance of the three methods for fold detection on real patient data. Note that for the digital colon phantoms, we use precise ground-truth information and undertake the sensitivity–specificity analysis for comparing the performance of the three methods. For the real patient CTC data, there exists practical difficulty of extracting precise ground-truth here. We perform visual inspection and use over-segmentation (criterion: >25% of the area between two folds is detected) and under-segmentation (criterion: <75% of a fold area is detected) to evaluate the three methods. Intuitively, over-segmentation and under-segmentation can be considered as inverse measures of specificity and sensitivity, respectively. The ‘combined’ method considerably improves the over-segmentation of the diffusion-FCM and under-segmentation of the level sets. It is evident from Table 3 that the ‘combined’ method (with a total of 13 ± 4 erroneous detections) outperforms both the diffusion-FCM (with a total of 25 ± 10 erroneous detections) and the level set (with a total of 23 ± 6 erroneous detections). On average, a colon is found to contain around 200 folds. Thus, we also present the detection errors in terms of the percentage of total number of folds. As shown in Table 3, the ‘combined’ method has a total erroneous (over-segmentation plus under-segmentation) detection of only (6.5 ± 2)% of the total number of folds per colon as compared to (12.5 ± 5)% for the diffusion-FCM-based method and (11.5 ± 3)% for the level set-based method.

In order to determine whether the performance improvement of the ‘combined’ method compared to the diffusion-FCM-based method as well as the level set-based method is statistically significant, we apply the ANOVA test (Johnson and Bhattacharya, 2006) with total number of erroneous (i.e., over-segmented plus under-segmented) folds (T) in Table 3 as the ‘factor’. The p-value of the test between the combined method and the diffusion-FCM-based method is 2.87 × 10⁻⁷ and that between the combined method and the level set method is 5.07 × 10⁻², indicating performance improvements in both the cases to be statistically significant. Fig. 5 graphically illustrates the better performance of the ‘combined’ method as the curve for this method appears below the curves of the other two for all the 18 real patient datasets. In Fig. 6A–C, we respectively show the impacts of fold detection on the flattened surface of a real patient colon using the three methods. Note that all the processing is done on the actual colon surface first and the detected folds are then visualized on corresponding flattened surfaces. Clearly, Fig. 6C gives the best results, which is also corroborated by the ANOVA test. As the datasets are chosen

Table 2

<table>
<thead>
<tr>
<th>Data</th>
<th>Diffusion-FCM</th>
<th>Level set</th>
<th>Combined</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noisless phantom</td>
<td>Sensitivity:</td>
<td>Sensitivity:</td>
<td>Sensitivity:</td>
</tr>
<tr>
<td></td>
<td>99.56 ± 0.43</td>
<td>85.55 ± 0.91</td>
<td>96.73 ± 0.28</td>
</tr>
<tr>
<td></td>
<td>Specificity:</td>
<td>Specificity:</td>
<td>Specificity:</td>
</tr>
<tr>
<td></td>
<td>67.90 ± 1.06</td>
<td>98.59 ± 1.49</td>
<td>88.77 ± 1.74</td>
</tr>
<tr>
<td>Noisy phantom</td>
<td>Sensitivity:</td>
<td>Sensitivity:</td>
<td>Sensitivity:</td>
</tr>
<tr>
<td></td>
<td>98.96 ± 0.55</td>
<td>84.29 ± 2.19</td>
<td>96.02 ± 0.47</td>
</tr>
<tr>
<td></td>
<td>Specificity:</td>
<td>Specificity:</td>
<td>Specificity:</td>
</tr>
<tr>
<td></td>
<td>61.01 ± 3.21</td>
<td>96.12 ± 0.41</td>
<td>87.69 ± 1.54</td>
</tr>
</tbody>
</table>
from three different virtual colonoscopy databases with varying dimensions and resolutions, we can clearly claim the superiority of the 'combined' method over the other two. On average, the diffusion-FCM-based method takes a few minutes while the level set as well as the 'combined' methods take a few hours to complete for a typical real dataset. So, one limitation of the proposed 'combined' method is its somewhat high execution time.

8. Conclusions and future scope

In this paper, we address an important problem in the area of virtual colonoscopy, namely haustral fold detection. Identification of haustral folds can potentially improve prone–supine registration, colonic polyp detection and tenia coli extraction, which can lead to a better CAD system with more effective virtual endoscopic navigation. Two different methods are proposed for the detection of colonic folds. The first method uses physics-based modeling (heat diffusion) followed by a pattern clustering technique (FCM) to identify the haustral folds on volumetric CTC images. The folds are then transferred on corresponding colon surfaces. The second method employs level sets to detect the same folds directly on colon surfaces. The first method is found to suffer from over-segmentation while the second method has a tendency of under-segmentation. An innovative synergism of the two methods, where the diffusion-FCM method is used to generate seeds for the level set method, is proposed. Extensive experimental validations on colonic fold detections, which remain so far unavailable in the literature, are presented using appropriately constructed digital colon phantoms as well as several real patient CTC datasets. Sensitivity–specificity analysis, number of erroneous fold detections, ANOVA tests, and visual comparisons indicate the superiority of the 'combined' method. As a summary of technical innovation, it can be stated that we have (i) improved the mesh structure level set method initially reported in (Tan et al., 2008), (ii) proposed a synergistic combination of diffusion-FCM and level sets with improved performance and (iii) designed a novel speed function as the stopping criterion for the level sets.

Our main focus in this paper is on accurate detection of haustral folds and not on the processing time. As part of our future work, we plan to optimize the performance of the discussed methods, e.g. by using narrow-band level set framework to speed up the computation of the combined method. To have more accurate validation,
we will construct more realistic digital colon phantoms involving more intricate and complex geometry. In addition, we will use the haustral folds for improving prone–supine registration, colonic polyp detection and tenia coli extraction.

Acknowledgements

This research was supported by the Intramural Research Program of the NIH Clinical Center. We greatly thank Dr. Nicholas Petrick from Center for Devices and Radiological Health, US Food and Drug Administration for his suggestions on construction of digital colon phantoms. We also thank Dr. Perry Pickhardt, Dr. J. Richard Choi, and Dr. William Schindler for providing CT colonography data.

References