Corresponding Supine and Prone Colon Visualization Using Eigenfunction Analysis and Fold Modeling

Saad Nadeem, **Joseph Marino**, **Xianfeng Gu**, **Arie Kaufman [Fellow, IEEE]**

Computer Science Department, Stony Brook University, Stony Brook, NY 11794-2424.

Abstract

We present a method for registration and visualization of corresponding supine and prone virtual colonoscopy scans based on eigenfunction analysis and fold modeling. In virtual colonoscopy, CT scans are acquired with the patient in two positions, and their registration is desirable so that physicians can corroborate findings between scans. Our algorithm performs this registration efficiently through the use of Fiedler vector representation (the second eigenfunction of the Laplace-Beltrami operator). This representation is employed to first perform global registration of the two colon positions. The registration is then locally refined using the haustral folds, which are automatically segmented using the 3D level sets of the Fiedler vector. The use of Fiedler vectors and the segmented folds presents a precise way of visualizing corresponding regions across datasets and visual modalities. We present multiple methods of visualizing the results, including 2D flattened rendering and the corresponding 3D endoluminal views. The precise fold modeling is used to automatically find a suitable cut for the 2D flattening, which provides a less distorted visualization. Our approach is robust, and we demonstrate its efficiency and efficacy by showing matched views on both the 2D flattened colons and in the 3D endoluminal view. We analytically evaluate the results by measuring the distance between features on the registered colons, and we also assess our fold segmentation against 20 manually labeled datasets. We have compared our results analytically to previous methods, and have found our method to achieve superior results. We also prove the hot spots conjecture for modeling cylindrical topology using Fiedler vector representation, which allows our approach to be used for general cylindrical geometry modeling and feature extraction.

Index Terms—

Medical visualization; colon registration; geometry-based techniques; mathematical foundations for visualization

1 Introduction

Colorectal cancer (CRC) is the third most frequently diagnosed cancer worldwide, and the fourth leading cause of cancer-related mortality with 700,000 deaths per year worldwide. Optical colonoscopy (OC) is an uncomfortable invasive technique commonly used to screen for CRC. In contrast, virtual colonoscopy (VC) was introduced as a non-invasive procedure

sanadeem@cs.stonybrook.edu.

for the mass screening of polyps [12], the precursors to CRC. VC reconstructs a 3D colon model from a computed tomography (CT) scan of a patient's abdomen and then a physician virtually navigates through the colon looking for polyps. The low radiation dosage, the recent assignment of CPT code 74263 for insurance reimbursement, the lack of side effects from anesthesia, and the quick turnaround time for results makes VC a prime method for CRC screening.

In a typical VC, the patient is scanned twice, supine (facing up) and prone (facing down) or, more recently, in a side position called the lateral recumbent position to avoid artifacts due to the squishing of the colon in the prone position. The two scans improve the sensitivity and specificity of polyp detection and reduce the extent of uninterpretable collapsed or fluid-filled segments. Because the colon changes between the supine and prone (or side) scans, the two colon models need to be registered to match detections. Computer-aided detection (CAD) techniques [13,36] have been introduced to automatically detect polyps, but they have issues. Registration can aid these CAD techniques by reducing false positives (primarily due to stool and colon folds) when matched across scans. Moreover, the recurrence of cancer due to the incomplete removal of polyps can be tracked via registration of scans captured at different periods. This allows physicians to perform an effective follow-up examination by visualizing the polyp growth over time. Hence, registration provides an effective tool to distinguish between lesions, aid in CAD, and visualize polyp growth over time. In this paper, we focus on registering VC data across multiple patient positions.

The first non-trivial eigenfunction of the Laplace-Beltrami operator is known as the Fiedler vector representation. This representation can be used by itself for complex shape modeling of tubular structures such as the colon. Lai et al. [16] proposed the use of the Fiedler vector for supine and prone colon registration. However, their registration depends on the accurate detection of important landmarks, based on assumptions which often did not hold in our patient colon datasets, leading to a higher registration distance error. Instead, we integrate the Fiedler vector representation with a more accurate feature detection based on haustral fold modeling to achieve better registration and visualization results. We first use the Fiedler vector representation for global registration of the scans. We then use 3D level sets computed based on this representation to segment the folds. Finally, we refine the registration locally via these folds as anatomical references. An overview of our pipeline is given in Figure 2. The accurate modeling of the colon folds and the Fiedler vector representation results in automatic cross visualization of consistent endoluminal views across multiple orientations.

Different flattening approaches have been introduced to visualize the complex geometry of medical organs [3, 10, 11, 13, 20, 21, 27]. The Fiedler vector representation and fold segmentation also allows us to create more accurate 2D flattening visualizations by automatically computing consistent cuts throughout the colon without crossing the folds. These cuts are then used as an input to a quasi-conformal mapping [35], which is used to flatten the colon. Previous flattening approaches require manually selecting extrema on the colon and computing geodesic paths to create a cut. This cut, however, crosses folds on sharp bends since this is considered the shortest path, resulting in flattened visualizations with chopped folds, as shown in Figure 8.

There have been other registration approaches which require either using the centerline or flattening approaches which distort texture and require high quality data. Moreover, once the data is flattened to a 2D rectangle, it is difficult to find correspondences between the original data and the flattened data due to the loss of geometry. It is our endeavor here to be able to easily visualize this correspondence and correlate anatomical features across scans in the original and the flattened data. Our registration framework makes use of accurate fold models to segment out folds with high precision. These fold segmentations are then used as anatomical references to register the VC data across scans. Moreover, the fold segmentations allow us to automatically find suitable cuts to map the 3D data to simpler 2D topologies and visualize the resultant data more effectively. Our contributions are as follows:

- Integrated colon registration framework using Fiedler vector representation for global registration and 3D level set fold segmentation for local refinements.

- Accurate fold modeling which provides state-of-the-art detection and segmentation results on colon models.

- Accurate correspondence visualization between 3D and 2D flattening visualizations due to the Fiedler vector representation and accurate fold segmentation.

- Extraction of a consistent cut along the colon for visually accurate colon flattening.

- The theoretical proof of the hot spots conjecture for modeling general cylindrical topology using Fiedler vector representation.

The paper is organized as follows. In the next section, we discuss related work, followed by a brief theoretical background of Fiedler vector representation in our context in Section 3. This provides the basis for our colon registration algorithm, which we outline in Section 4. In Section 5, we highlight the supine and prone correspondence and more effective flattening visualizations based on our algorithms. Finally, we evaluate our algorithm and show additional results in Section 6, followed by the conclusion and future work in Section 7.

2   Related Work

There have been several works in the colon registration domain based on the identification of landmarks and centerline matching across supine/prone/side scans. A basic method applies linear stretching and shrinking operations to the centerline, where local extrema are matched and used to drive the deformations [1, 2]. Correlating individual points along the centerline through the use of dynamic programming has also been suggested [7, 17, 24].

More recently, the taeniae coli (three bands of smooth muscle along the colon surface) have been used as features which can be correlated between the two scans [14]. This relies on a manual identification of one of the three taeniae coli, and then an automatic algorithm repeats the line representing the identified taenia coli at equal distances. Further progress has been made where the haustral folds and the points between them can be automatically

detected, and the taeniae coli are identified by connecting these points [30]. However, this method is only feasible on the ascending and transverse portions of the colon.

Deformation fields have also been suggested for use in supine-prone registration. Motion vectors can be identified for matched centerline regions, interpolated for non-matched regions, and then propagated to the entire volume [29]. It has also been proposed to use a free-form deformation grid to model the possible changes in the colon shape from supine to prone [26].

Conformal mapping has been successfully used for many medical applications, including a brain cortex surface morphology study [11] and colonic polyp detection [13]. Quasi-conformal mapping was used to map supine and prone colon datasets to a rectangular plane and convert the 3D registration problem into a 2D image matching problem [35]. This, however, required landmarks to be extracted in order to divide the colon into different well-defined segments. Moreover, the whole registration pipeline was dependent on flattening these segments by tenaie coli extraction and then using graph matching. The flattening was computationally expensive, and precise fold segmentation could have improved the accuracy of the graph matching algorithm.

There has been minimal use of folds as anatomical references for registration due to the imprecise modeling of folds, thus leading to poor fold detection and segmentation results. Effective segmentation of the colon folds is necessary for detecting polyps on the folds [34], extracting teniae coli [14,38], and performing supine-prone registration [1, 2, 31, 35]. Most of the previous works have focused on colon fold detection [5, 14, 33], disregarding the precise segmentation or delineation of the boundaries of the fold. A fold boundary modeling and segmentation algorithm was introduced, and the segmented-area ratio (SAR) metric was also introduced to measure the accuracy of the segmentation [37]. In this work, we model and segment the folds (using Fiedler vector representation) on the supine-prone or supine-side VC dataset pairs which in turn are used as anatomical references to register these pairs.

Fiedler vector representation has been used in various applications including mesh processing [8], mesh parameterization [23], and shape characterization [4]. Lai et al. [16] have proposed the use of Fiedler vector representation for supine and prone colon registration. Based on the Fiedler vector computation for supine and prone colons, they detect the hepatic and splenic flexures, and register the supine and prone colons piecewise using these detected landmarks. They assume that the splenic and hepatic flexures are denoted by local maximum z-coordinates nearest to the maximum and minimum Fiedler vector values, respectively. This assumption, however, is frequently invalid in real patient colon datasets which we have tested on and can result in higher distance registration error compared to the registration done solely based on the Fiedler vector without considering the landmarks. Moreover, their algorithm does not take into account the Fiedler vector flips that can occur between the corresponding supine and prone colons which could lead to erroneous registration results (e.g. the rectum-splenic segment being registered to the cecum-hepatic segment).

In contrast, we use the Fiedler vector representation to model the human colon, register the supine/prone/side models in 3D, and segment folds using level sets based on this representation. The Fiedler vector representation is used to register the supine/prone/side datasets globally and then the segmented folds are used as anatomical references to locally refine the registration results. The fold segmentation based on the level sets is efficient and the overall registration process does not depend on the computation of the centerline, the teniae coli, or the flexures. Our use of the Fiedler vector representation for fold segmentation is closely related to spectral image [19] and shape segmentation [28]. The colon registration in 3D, in essence, is similar to shape matching using functional maps [15, 25].

3   Theoretical Background

In this section, we briefly introduce the theoretical background necessary for this work, which allows us to compute extrema and model the complex geometry of the colon. The hot spots conjecture [6] states that the minimum and maximum of the Fiedler vector are the two points with the greatest geodesic distance on the surface. We prove in Lemma 2 that this conjecture holds for a topological cylinder.

Suppose S is a surface embedded in the three-dimensional Euclidean space $\mathbb{R}^3$, and the induced Euclidean Riemannian metric is denoted as $\mathbf{g}$. One can choose a special type of local coordinates (x, y), called *isothermal coordinates*, such that the metric has a canonical form $\mathbf{g} = e^{2\lambda(x,y)}(dx^2 + dy^2)$, where the function $\lambda : S \to \mathbb{R}$ is called the *conformal factor*. The *Laplace Beltrami* operator induced by the metric is given by $\Delta_{\mathbf{g}} = \frac{1}{e^{2\lambda}}\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}\right)$. The *Gaussian curvature* of the surface is given by $K(x,y) = -\Delta_{\mathbf{g}}\lambda(x,y)$. An *eigenfunction* of the Laplace-Beltrami operator is given by $\Delta_{\mathbf{g}}\phi_k = \lambda_k\phi_k$, where the *eigenvalue* $\lambda_k \geq 0$ is a non-negative real number. The Laplace-Beltrami operator has infinitely many eigenvalues and eigenfunctions. The eigenvalues are all sorted in ascending order $0 = \lambda_0 \leq \lambda_1 \leq \lambda_2 \cdots \lambda_k \cdots$. The eigenfunctions form an ortho-normal basis of the functional space of the surface, $\langle \varphi_i, \varphi_j \rangle = \int_S \varphi_i(p)\varphi_j(p)dA_{\mathbf{g}} = \delta_i^j$. The first eigenfunction φ_0 is a constant function. The second eigenfunction φ_1 is called *Fiedler*'s function.

The *heat kernel* of the surface is $K(p, q, t) = \sum_{k=0}^{\infty} e^{-\lambda_k t}\varphi_k(p)\varphi_k(q)$. The *heat diffusion* equation on the surface is

$$\frac{du(p, t)}{dt} = \Delta_{\mathbf{g}}u(p, t), \tag{1}$$

and the solution to the heat equation is given by the heat kernel $u(p,t) = \int_S K(p,q,t)u(q,0)dA_{\mathbf{g}}$. Another equivalent way to represent the solution to the heat equation is

$$u(p, t) = \sum_{k=0}^{\infty} \tau_k e^{-\lambda_k t}\varphi_k(p), \tag{2}$$

where

$$\tau_k = \left\langle \varphi_k, u(p,0) \right\rangle = \int_S u(q,0)\varphi_k(q)dA_{\mathbf{g}}.$$

When t goes to infinity, the right hand side of Equation 1 goes to 0, which means that the temperature function $u(p,t)$ becomes harmonic.

Lemma 1

Suppose $u: S \to \mathbb{R}$ is a harmonic function, then for any interior point $p \in S$,

$$u(p) = \frac{1}{2\pi} \oint_\gamma u(q)dq, \tag{3}$$

where γ is a small circle surrounding p.

Proof 1

If we choose isothermal coordinates (x,y), then $u(x,y)$ is a harmonic function and $\triangle u = 0$. We construct the conjugate harmonic function $v: S \to \mathbb{R}$, such that $u_x = v_y$ and $u_y = -v_x$, then the complex function $F: S \to \mathbb{C}$ is a holomorphic function, where $z = x+iy$. Note that F is constructed using u as a real part and v as an imaginary part. By Cauchy's formula, we obtain

$$F(z) = \frac{1}{2\pi i} \oint_\gamma \frac{F(w)}{w-z}dw,$$

where $w \in \gamma$, $w = z + \varepsilon e^{i\theta}$. Comparing the real and imaginary parts, we obtain Equation 3.

Corollary 1

A metric surface $(S,\mathbf{g})$ has boundaries ∂S. Suppose u is a harmonic function on S, then the maximal and minimal points of u are on ∂S.

Proof 2

Equation 3 means that the value of an interior point $u(p)$ is the mean of the values of its neighboring points on the small circle γ. Therefore, the maximal and minimal points cannot be in the interior of S and hence, these points must be on the boundaries ∂S.

Lemma 2

Suppose the metric surface $(S,\mathbf{g})$ is a topological cylinder, then the maximal and minimal points of its Fiedler vector φ_1 are on the boundaries of the surface ∂S.

Proof 3

Suppose the solution to Equation 1 is $u(p,t)$. When t goes to infinity, $u(p,t)$ converges to a harmonic function, and therefore the maximal and minimal points of $u(p,t)$ are close to the boundaries ∂S.

From Equation 2, when t becomes large enough, the high order eigenfunctions go to 0 much faster. Therefore, the behavior of $u(p,t)$ is mainly controlled by the first two terms,

$$u(p,t) \sim \tau_0 \varphi_0(p) + \tau_1 e^{-\lambda_1 t} \varphi_1(p),$$

where φ_0 is constant. Therefore, the maximal and minimal points of $u(p,t)$ are approximated by those of $\varphi_1(p)$, and thus the maximal and minimal points of $\varphi_1(p)$ approach the surface boundaries S.

Level Sets of Fiedler's Function—Suppose $\varphi : (S,\mathbf{g}) \rightarrow (T,\mathbf{h})$ is a diffeomorphic mapping between two metric surfaces. The local coordinates of S are (x,y) and those of T are (u,v). The *Jacobian* matrix of the mapping φ is J_φ. The *pull back* metric induced by the mapping φ is defined by $\varphi^*\mathbf{h} = J_\varphi^T \mathbf{h} J_\varphi$. The mapping φ is *conformal* if there is a function $\mu : S \rightarrow \mathbb{R}$,

$$\varphi^*\mathbf{h} = e^{2\mu}\mathbf{g}. \tag{4}$$

According to conformal geometry, if $(S,\mathbf{g})$ is a topological cylinder, then there is a conformal mapping which maps the surface onto a flat cylinder $\mathscr{C}, \mathscr{C} = \{(\cos\theta, \sin\theta, h) \mid 0 \leq \theta < 2\pi, 0 \leq h \leq H\}$. The second eigenfunction of $\mathscr{C}$ is $\varphi_1(\theta,h) = e^h$. The level sets of the second eigenfunction are circles with constant height $h = const$.

Suppose a topological cylindrical surface $(S,\mathbf{g})$ is conformally mapped onto a flat cylinder $\mathscr{C}$. Furthermore, the mapping is nearisometric, namely, the conformal factor μ in Equation 4 is close to 0. Then the level sets of the second eigenfunction of S are similar to those on $\mathscr{C}$. In our current work, we use the level sets of the second eigenfunction to locate the folds on the colon surface.

4 Computational Algorithm

In this section, we explain the computational algorithms for our registration method. We globally register the supine and prone colons based on their respective Fiedler vector representations. This is followed by bounded fold contour detection using 3D level set bundles computed using the Fiedler vector representation, and the eventual segmentation into individual folds based on the 2D normalization of the projected detected fold contour planes. These segmented folds are used as references on the globally registered colons to locally refine the registration and result in the final output of our registration algorithm.

4.1 Fiedler Vector

Suppose the colon surface is $(S,\mathbf{g})$, where $\mathbf{g}$ is the Riemannian metric. The colon surface is a topological cylinder with two boundaries, $S = \gamma_1 - \gamma_0$. We want to compute the Fiedler vector with the Dirichlet boundary condition

$$\begin{cases} \Delta_{\mathbf{g}}\xi = \lambda\xi, \lambda > 0 \\ \xi(p) = 1, p \in \gamma_1 \\ \xi(p) = 0, p \in \gamma_0 \end{cases}$$

where $\Delta_{\mathbf{g}}$ is the Laplace-Beltrami operator [22] induced by $\mathbf{g}$.

In the discrete setting, the colon surface is extracted from CT images, and approximated by a triangular mesh, denoted as $M = (V,E,F)$, where V is the set of vertices, E the set of edges, and F the set of faces. The *cotangent edge weight* is defined as follows: suppose $[v_i, v_j] \in E$ is an edge, shared by two triangular faces $[v_i, v_j, v_k]$ and $[v_j, v_i, v_l]$, then the edge weight is $w_{ij} = \cot\theta_{ij}^k + \cot\theta_{ji}^l$, where θ_{ij}^k is the corner angle at vertex v_k in the face $[v_i, v_j, v_k]$. Suppose a function is defined on the vertex, $f : V \to \mathbb{R}$. The *discrete Laplace-Beltrami operator* is defined as $\Delta f(v_i) = \sum_{v_j \sim v_i} w_{ij}(f(v_j) - f(v_i))$, where $v_j \sim v_i$ means the vertex v_j is adjacent to v_i. Therefore, the Laplace-Beltrami operator has a matrix representation $\Delta = (\delta_{ij})$, where

$$\delta_{ij} = \begin{cases} -w_{ij} & v_i \sim v_j \\ \sum_k w_{ik} & i = j \\ 0 & v_i \not\sim v_j \end{cases}$$

We compute the eigen decomposition of Δ; the eigenvalues are $\{\lambda_0, \lambda_1, \cdots, \lambda_n\}$ and the eigenvectors are $\{\eta_1, \eta_2, \cdots, \eta_n\}$. The first eigenvalue of Δ is $\lambda_0 = 0$ and the first eigenvector is $(1, 1, \cdots, 1)^T$. The first positive eigenvalue is λ_1 and the Fiedler vector is the corresponding eigenvector η_1. We scale η_1 such that the maximum value of η_1 is 1 and the minimum value is 0. In the following discussion, we treat η_1 as a function defined on the vertex set. Figure 4(a) illustrates the Fiedler vector of a colon surface with color encoding, and Figure 4(d) shows the Fiedler vector on the flattened colon surface.

4.2 Global Registration

Two colon surfaces can be registered directly based on their Fielder vectors. Suppose the two colon surfaces are S_1 and S_2, then the corresponding Fiedler vectors are ξ_1 and ξ_2, respectively. Furthermore, the boundaries of the two surfaces are $\partial S_k = \gamma_1^k - \gamma_0^k$, $k = 1,2$. We first match the boundary curves using their arc lengths. We choose base points $p_1 \in \gamma_0^1$ and $p_2 \in \gamma_0^2$, where the base points are the extrema which are computed using the maximum and minimum Fiedler vector values. To avoid Fiedler vector flips between the corresponding supine and prone data, we manually make these maximum or minimum Fiedler vector values consistent on the rectum or the cecum for both the supine and prone datasets. The arc length is then used to parameterize the boundary curves and normalize the total length to 2π. The arc length parameterizations are denoted as θ_1 and θ_2. The mapping $\theta_1 \mapsto \theta_2$ gives the mapping from γ_0^1 to γ_0^2.

We compute the gradient field on the surfaces, denoted as $\nabla\xi_1$ and $\nabla\xi_2$ respectively, then we trace the integration curves of the gradient fields. The curve $\tau_1(t)$ represents an integration

curve on S_1, $\tau_1(0) \in \gamma_0^1$ and $\tau_1(1) \in \gamma_1^1$. Similarly, $\tau_2(t)$ is an integration curve on S_2 starting from γ_0^2. A point $p \in S_1$ is the intersection of a level set of $\xi_1^{-1}(t)$, $t \in [0,1]$ and an integration curve τ_1 starting from a point on γ_0^1 with the arc length parameter θ_1. The whole colon surface S_1 is parameterized by (θ_1, t), and S_2 is globally parameterized by (θ_2, t). The initial global registration is given by the mapping $\varphi : (\theta_1, t) \mapsto (\theta_2, t)$.

4.3 Fold Detection

Suppose S is a colon surface with Fiedler vector ξ. We denote the level sets of ξ as γ_t, where $t \in [0,1]$, namely $\gamma_t = \xi^{-1}(t)$. We uniformly sample the level sets on the colon surface and obtain a family of level sets $\{\gamma_t\}$, as shown on the colon surface in Figure 4(b) and the flattened colon surface in Figure 4(e). We also compute the integration curves of the gradient field of ξ and obtain a family of integration curves $\{\tau_\theta\}$, $\theta \in [0, 2\pi)$, as shown in Figure 5, where the red loops are the level sets $\{\gamma_t\}$ and the yellow curves are the integration curves $\{\tau_\theta\}$.

We compute the normal curvature of points $\tau_\theta(t)$. If a point (θ, t) is in a fold area, then the normal curvature of $\tau_\theta(t)$ is negative; in other flat or convex areas, the normal curvature of $\tau_\theta(t)$ is zero or positive. We compute the *inflection points* of the integration curves, where the normal curvatures are 0's, and the points with minimal normal curvatures. Fixing an integration curvature τ_θ, suppose $t_0 < t_1 < t_2$, where t_0 and t_2 are inflection points, t_1 is the minimal curvature point, then the level sets γ_t, $t \in [t_0, t_2]$ form a level set bundle. In this way, we compute the clusters of level sets. We discard the initial uniformly sampled level sets, which do not belong to any bundles. The level set bundles are shown in Figure 6(d).

We densely sample the level sets within each bundle, and compute the normal curvatures. The points (θ, t) with negative normal curvature of curve τ_θ are in the fold area. In this way, we can detect the fold contour, as shown in Figure 6(e).

Bowel preparation, done prior to VC, might lead to local under-distention of some colon regions, which we refer to as collapsed regions. In this case, we ignore the extraction of folds in these regions since these folds do not exhibit any meaningful information. We set a threshold on the level set bundle such that if the encompassing circle is below 0.1 in radius, when normalized in 2D, we mark that region as locally under-distended or collapsed, as shown in Figure 9. However, severe cases of local under-distension can lead to a complete collapse of the colon at a particular region, resulting in multiple colon segments during the segmentation process. Our method does not cater to these severe cases, which will be a focus of future research.

4.4 Haustral Fold Segmentation

For each level set γ_t, we find a best fit plane π_t as follows. We choose samples on γ_t, denoted as $\{p_1, p_2, \cdots, p_n\}$, and the center of the samples is given by $c = 1/n \sum_{i=1}^n p_i$. We compute the covariance matrix of the samples, $A = 1/n \sum_{i=1}^n (p_i - c) \otimes (p_i - c)$, then compute the eigen decomposition of A. The eigenvectors are $\{\eta_1, \eta_2, \eta_3\}$, which form an orthonormal, then the fitting plane π_t goes through the center c and is spanned by η_1 and η_2, namely, π_t is

given by the equation $\langle p - c, \eta_3 \rangle = 0$. The fitting planes of the level sets are illustrated in Figure 6(c).

We then project the level set γ_t onto the fitting plane π_t. Suppose $p \in \gamma_t$, then the projection on the plane is given by $p - \langle p, \eta_3 \rangle \eta_3$. We denote the projected level set as $\tilde{\gamma}_t$. Suppose π_t and π_s are two fitting planes, with centers c_t and c_s, we align them together as follows: we shift c_t to c_s by a translation, then rotate π_t to coincide with π_s by a rotation $\mathcal{R}$, such that the rotation angle is the angle between the normals to the two planes and the rotation axis is along the cross product of the two normals. In this way, we align the fitting planes of all level sets within a bundle, and the spatial level set curves become a cluster of planar curves as shown in Figure 5(c). We denote each projected and aligned level set as $\hat{\gamma}_t$.

An integration curve $\tau_\theta(t)$ on the original surface becomes a planar curve $\hat{\tau}_\theta(t)$ by mapping $\gamma_t(\theta)$ to $\hat{\gamma}_t(\theta)$. Suppose the level set bundle is $\{\gamma_t\}$, where $t_0 \leq t \leq t_1$. We compute the total length of each $\hat{\tau}_\theta$ and find the local minima with respect to θ. Then each local maximum corresponds to a haustral fold. This procedure produces the haustral fold segmentation, as shown in Figure 6(e) for the 3D view and Figure 6(g) for the flattened view. We compare our automatic haustral fold segmentation with manual segmentation as shown in Figures 7 and 10, where the green contour shows the manual segmentation results and the red contours show the automatic results. One can see that the automatic segmentation has very high accuracy.

4.5 Local Registration Refinement

The folding areas and the segmented haustral folds are used as anatomical references. From the initial registration, we can find the correspondences among the haustral folds on colon surfaces obtained from the supine/prone/side data. We then locally deform the initial mapping in order to align the haustral folds more accurately. We denote the two flattened colon surfaces as S_1 and S_2 respectively, $\varphi : S_1 \rightarrow S_2$ is the mapping between them. We define the characteristic function $\chi_k : S_k \rightarrow \mathbb{R}$, $k = 1,2$, where $\chi_k(p)$ equals 1 if p is inside a haustral fold, and 0 otherwise. Then we smooth the characteristic functions out by convolving a Gaussian filter. We define an energy for the mapping φ as follows:

$$E(\varphi) = \int_{S_1} |\chi_1(p) - \chi_2 \circ \varphi(p)|^2 dA + \int_{S_1} |\nabla \varphi|^2 dA .$$

The first term ensures that haustral folds match haustral folds, and the second term measures the smoothness of the mapping. We obtain the Euler-Lagrange equation, and the following flow minimizes the energy,

$$\frac{\partial \varphi(p,t)}{\partial t} = -(\chi_1(p) - \chi_2 \circ \varphi(p)) \nabla \chi_2(\varphi(p)) + \Delta \varphi(p) .$$

By deforming the mapping along the flow, the registration is improved significantly. Figure 11 shows the registration result using the haustral folds as references.

5 Visualization

Due to the use of the Fiedler vector representation throughout the entire colon, it becomes possible to easily co-locate positions between two scans, either manually or automatically. Using consistent cuts along the colon based on the segmented folds, we achieve more effective flattened visualizations than other works. The Fiedler vector representation also provides easier visualizations to find correspondences between the 3D model and the flattened view while allowing for better polyp localization via labeling of segmented folds.

5.1 Corresponding Supine and Prone Visualization

The Fiedler vector representation can be used to automatically visualize corresponding regions in multiple patient orientations and in respective 3D and 2D mapped views. The consistent endoluminal views in Figures 1 and 3 highlight the polyps in supine-prone and supine-side orientations, respectively. In order to visualize both supine and prone in a consistent manner, we map the camera frustum viewing the supine mucosa (inner surface) to the camera frustum viewing the prone mucosa. This is done by computing the means of the level sets based on the Fiedler vector which result in corresponding centerlines on both supine and prone. We then match the camera orthonormal coordinate frames for the two views.

The centerlines are matched based on the eigenfunction bundles following the registration algorithm, outlined in Section 4. Given a point c_0 on the discretized supine centerline, we want to find the corresponding point c_1 on the discretized prone centerline. We first find the point d_0 on the supine centerline, which is the point nearest c_0 in $\mathbb{R}^3$. Given this point d_0, the corresponding point d_1 on the prone centerline is found at the same index. c_1 can then be found as the point on the prone centerline closest in $\mathbb{R}^3$ to d_1.

We focus our endoluminal correlation work on situations where the user is looking at something on the colon wall and wants to view the corresponding location in the other scan. Using the correspondence along the skeletons, it is possible to create a correlated automatic navigation, though we found running two fly-throughs side-by-side to be distracting and more of a hindrance than help. Orienting rotation around the centerline for two automatic navigation views is possible based on the individual haustral fold registration on the corresponding supine and prone colons.

5.2 Flattened Visualizations

Since the Fiedler vector representation captures extreme points on a given colon dataset, we can use these extreme points along with the fold segmentation results to trace a consistent cut throughout the colon. This cut can then be used as input to a quasi-conformal mapping algorithm [35] to flatten the colon to a 2D plane. In essence, the following steps are used to automatically extract a consistent cut: (1) extract the extreme points on the colon dataset, (2) remove the 1-ring vertex set corresponding to the extreme vertices, (3) pick two haustral folds (in the same level set bundle) closest to one of the extreme points, (4) connect the mean of the endpoints of these folds to the next level set bundle of haustral fold endpoints, closest to the centroids of the folds in the previous bundle, and (5) repeat the previous step

until the other extreme point is reached. The result of this process, compared to a typical cut based on a geodesic path, is illustrated in Figure 8.

To the best of our knowledge, there are no automatic algorithms for extracting a consistent cut throughout the colon. Normally, two extreme points are manually picked on a given colon dataset and a geodesic path is computed from one point to the other and this constitutes the cut. However, the problem with this cut is that it crosses the folds when a sharp bend is incurred, as shown in Figure 8, whereas our method takes into account individual haustral folds before tracing the cut which allows for consistency.

In addition to greater consistency, our cut produces less angle distortion as compared to the geodesic path cut. We quantify the angle distortion metric of the resultant flattened colon by using the signed singular values of the Jacobian of the transformation for each triangle [18]. Small angular distortion is indicated by a distortion value approaching 2. The colon flattened via geodesic path cut in Figure 8(b) produces an overall angle distortion value of 2.181, whereas our method in Figure 8(c) produces a value of 2.126.

6   Experimental Results

We have validated our algorithms using real VC colon data from the publicly available National Institute of Biomedical Imaging and Bioengineering (NIBIB) Image and Clinical Data Repository provided by the National Institute of Health (NIH). We perform electronic colon cleansing incorporating the partial volume effect [32], segmentation with topological simplification [13], and reconstruction of the colon surface as a triangular mesh via surface nets [9] on the original CT images in a pre-processing step. Though the size and resolution of each CT volume varies between clinical datasets, the general data size is approximately $512\times512\times450$ voxels with a resolution of approximately $0.7\times0.7\times1.0$ mm.

We have developed our algorithms using generic C++ on the Windows 7 platform. The linear systems for the Laplace equation were solved using the Matlab C++ library. All of the experiments are conducted on a workstation with a Core 2 Quad 2.50GHz CPU with 4GB RAM. On average, the Fiedler vector computation takes 3.5 seconds, the haustral fold detection and segmentation takes 3.2 seconds, and the final local refinement takes 1.8 seconds.

6.1   Fold Detection and Segmentation Evaluation

To evaluate our fold segmentation algorithm, we manually segmented the folds on 10 supine-prone and supine-side colon pair datasets. These manual segmentations were approved by a VC-trained radiologist who reviewed them. The accuracy of the proposed segmentation algorithm was measured on a per-fold basis, using the SAR metric [37]. Given the manually-established ground truth, a fold was assumed to be successfully detected if more than 50% of its area has been detected. Hence, we can calculate the true positives (TP), false positives (FP), true negatives (TN), false negatives (FN), sensitivity ($= TP/(TP+FN)$), and specificity ($= TN/(FP+TN)$). We calculate the SAR as follows:

$$SAR = A_0/(A_t + A_d - A_0)$$

where A_t is the ground truth area of the fold. A_d is the area of the segmented fold, and A_0 is the area of the overlap of the above two areas. SAR is defined as the ratio between the area of the intersection and the area of the union of the expert-drawn folds and the automatically segmented folds. In essence, a larger SAR suggests a better segmentation result, and $SAR = 1$ indicates that the segmented fold perfectly matches the ground truth without any over- or under-segmentation.

Table 1 lists the quantitative results from the 20 patient scans. About 90.4% of all the ground truth folds were detected with approximately 18 missed per dataset, under the assumption that a fold would be detected if more than 50% of its area has been segmented. The missed folds were mostly shallow and flat shaped structures. The FP detections were very few for each scan. For all the detected folds, the SAR is approximately 84.8%, indicating that the automatically segmented fold boundaries have matched fairly well with those of the manually-drawn counterparts. A visual comparison of the detection accuracy of our algorithm with a previous fold detection method using heat diffusion and fuzzy C-means clustering is provided as part of the Appendix A.2.

6.2   Analytic Registration Evaluation

To evaluate our registration results, we use the distance measurement between corresponding features located on registered colon surfaces. We use the haustral fold segmentation algorithm, detailed in Section 4, to find corresponding anatomical references on a given supine-prone colon pair. Based on these computed references and our manual labels, we use a subset of the segmented features for our registration and measure the distance errors on the remaining features. We generally extract more than 100 features from both pairs. For this registration error evaluation, we used 75 feature points for our registration and measure the distance errors on the remaining features.

A comparison between our method and other methods is performed using our analytic evaluation results in $\mathbb{R}^3$. For those papers that present their distance error, we compare our results with their results in Table 2. Our method produces a registration with significantly smaller distance error between corresponding points than other methods. When comparing to the published algorithm of Lai et al. [16], 3 out of our 10 colon pair datasets had flips and needed to be made consistent manually in order for their algorithm to work. Moreover, in 6 out of our 10 colon pair datasets, the Lai et al. algorithm performed worse than our global registration baseline ($\varepsilon = 0$ in their case) due to the incorrect detection of the landmarks. Further details about issues we encountered with their algorithm can be found in Appendix A.1.

6.3   Visual Registration Verification

We provide a visual registration verification of our algorithm by flattening the entire supine and prone colon models and then using the segmented folds (see Figure 9) as anatomical references to align the supine and prone datasets. In Figure 11, we show the entirety of both the supine and prone colon models, mapped to the plane, both unregistered and registered. The images of the registered segments clearly show very good alignment of the supine and prone colon structures, whereas the unregistered segments show poor alignment. Compared

to previous methods, our algorithm provides a visually superior result for viewing corresponding locations on the two colon models.

We have also shown our results to a VC-trained radiologist. Due to the use of the Fiedler vector color mapping on the colon, he was able to easily find the corresponding regions across supine and prone colons along with a strong correlation between the 3D endoluminal views and the flattened views.

7 Conclusions and Future Work

Shape registration is fundamental for shape analysis problems, especially for abnormality detection in medical applications. We have introduced an efficient framework for the registration of supine and prone colons, through the use of Fiedler vector representation, to improve the accuracy of polyp detection. Specifically, we use the Fiedler vector representation to globally register the supine and prone colon models. We then use level sets computed based on the Fiedler vector representation to detect and segment folds which are in turn used as anatomical references to locally refine the global registration results. The use of the Fiedler vector representation can help in easily visualizing the corresponding regions on 3D and 2D mappings as well as across supine and prone models. We have also proven the hot spots conjecture for modeling cylindrical topology using Fiedler vector representation, which allows our approach to be used for general cylindrical modeling and feature extraction. Furthermore, the fold segmentation allows for more consistent cuts along the colon surface which in turn provides more accurate flattened visualizations.

We have provided a thorough evaluation of our fold segmentation results by using the *SAR* metric on 20 manually labeled datasets (10 multi-orientation colon pairs). We have compared our method with other registration algorithms based on the computed registration error metric and have found our method to provide a significant improvement. Finally, we have also provided visual verification of our results on complete supine and prone colon pairs.

In the future, we will leverage the Fiedler vector level set approach for polyp detection in a given dataset and create a more integrated framework for computer-aided detection based on our registration results. It has been shown in previous works that the performance and accuracy of the computer-aided detection can be increased dramatically if the detection of polyps is done separately on haustral folds and endoluminal walls. We will use our fold segmentation framework to build this dichotomy and to help improve the accuracy of the CAD algorithms. We will also deploy our method in longitudinal analysis to register colons for the same patient across multiple visits, rather than just for single patient visits. Moreover, in the future, we will also cater to the severe under-distended colon cases where the segmentation can result in more than one colon segment per patient dataset. These cases can vary in complexity from a simpler (single supine/prone colon segment being registered to multiple supine/prone colon segments) to a more cumbersome (multiple supine/prone colon segments being registered to multiple supine/prone colon segments) scenario.

Acknowledgments

Appendix

A.1. Previous Fiedler vector colon registration

We have compared our work against a similar method using Fiedler vector introduced by Lai et al. [16]. Lai et al.'s automatic registration algorithm is dependent on the detection of two landmarks, namely the splenic and hepatic flexures, once the Fiedler vector has been computed on both the supine and prone models. Theoretically, the hepatic flexure forms the topmost point of the ascending colon and the splenic flexure forms the topmost point of the descending colon; this is the assumption of the Lai et al. algorithm as well. However, on real patient colon datasets that we tested, splenic and hepatic flexure detection is a much more complex problem and cannot simply be characterized by the local maximum z-coordinate near one Fiedler vector extrema or the other (as described in Lai et al.'s paper) due to the inflection points and the large variation in the troughs and ridges of the folds at the top. In general, local maximum z-coordinates can span several folds at the top and hence, the flexures detected within an epsilon value (using Lai et al.'s published approach) on the supine and prone colon models can vary considerably.

Due to this issue on some datasets when using Lai et al.'s registration algorithm, rather than improving the results (over our global registration baseline, which is essentially the registration with epsilon value of 0 in their case) once the landmarks (splenic and hepatic flexures) are detected, the results deteriorated in six out of our ten colon pair datasets (with an average registration distance error of 14.19 mm) due to the incorrect detection of the flexures on the corresponding colon pair (supine/prone/side) datasets. The average registration distance error for our complete algorithm, including the fold matching, was 5.24 mm for the same datasets.

Some of the registration problems that we encountered with the Lai et al. algorithm are highlighted in Figure 13 with three of our datasets. In Figure 13(b) for example, two local maxima are detected on the prone model within the $\varepsilon = 0.05$ value. This case is not handled based on the published Lai et al. approach. To compute the results, we picked the one closer to the maximum Fiedler vector value (as done by the Lai et al. approach for the supine but not for the prone model).

Moreover, the details of Lai et al.'s published algorithm do not deal with the flipping of the Fiedler vector between the supine and prone colon pairs. If the Fiedler vector values are flipped, this can lead to the cecum-hepatic segment being registered with the rectum-splenic segment on the corresponding model, which gives an erroneous result. To avoid the flipping in our case, we manually make the Fiedler vector minimum and maximum values consistent between the colon pairs.

A.2. Previous fold segmentation using fuzzy C-means

We provide a visual comparison of the detection accuracy of our algorithm with a previous fold detection algorithm using heat diffusion and fuzzy C-means clustering [5] in Figure 12. As can be observed from a visual inspection of the results, our method succeeds in segmenting the individual folds, whereas the fuzzy C-means clustering often results in multiple folds being segmented together as a single fold.

References

[1]. Acar B, Napel S, Paik D, Li P, Yee J, Beaulieu C, and Jeffrey R. Registration of supine and prone CT colonography data: Method and evaluation. Radiology, 221(332):332, 2001.

[2]. Acar B, Napel S, Paik DS, Li P, Yee J, Jeffrey RB Jr., and Beaulieu C. Medial axis registration of supine and prone CT colonography data. Proc. of Engineering in Medicine and Biology Society (EMBS), pp. 2433–2436, 10 2001.

[3]. Bartrolí AV, Wegenkittl R, König A, and Gröller E. Nonlinear virtual colon unfolding. IEEE Visualization, pp. 411–420, 2001.

[4]. Boscaini D, Eynard D, Kourounis D, and Bronstein MM. Shape-from-operator: Recovering shapes from intrinsic operators. Computer Graphics Forum, 34(2):265–274, 2015.

[5]. Chowdhury AS, Tan S, Yao J, and Summers RM. Colonic fold detection from computed tomographic colonography images using diffusion-FCM and level sets. Pattern Recognition Letters, 31(9):876–883, 2010.

[6]. Chung MK, Seo S, Adluru N, and Vorperian HK. Hot spots conjecture and its application to modeling tubular structures. Machine Learning in Medical Imaging, pp. 225–232, 2011. [PubMed: 27796012]

[7]. de Vries AH, Truyen R, van der Peijl J, Florie J, van Gelder RE, Gerritsen F, and Stoker J. Feasibility of automated matching of supine and prone CT-colonography examinations. British Journal of Radiology, 79:740–744, 9 2006.

[8]. Gebal K, Bærentzen JA, Aanæs H, and Larsen R. Shape analysis using the auto diffusion function. Computer Graphics Forum, 28(5):1405–1413, 2009.

[9]. Gibson SFF. Constrained elastic surface nets: Generating smooth surfaces from binary segmented data. Proc. of MICCAI, pp. 888–898, 1998.

[10]. Gurijala KC, Shi R, Zeng W, Gu X, and Kaufman A. Colon flattening using heat diffusion riemannian metric. IEEE Transactions on Visualization and Computer Graphics, 19(12):2848–2857, 2013. [PubMed: 24051852]

[11]. Haker S, Angenent S, Tannenbaum A, Kikinis R, Sapiro G, and Halle M. Conformal surface parameterization for texture mapping. IEEE Transactions on Visualization and Computer Graphics, (2):181–189, 2000.

[12]. Hong L, Muraki S, Kaufman A, Bartz D, and He T. Virtual voyage: Interactive navigation in the human colon. Proc. of SIGGRAPH, pp. 27–34, 1997.

[13]. Hong W, Qiu F, and Kaufman A. A pipeline for computer aided polyp detection. IEEE Transactions on Visualization and Computer Graphics, 12(5):861–868, 9 2006. [PubMed: 17080810]

[14]. Huang A, Roy D, Franaszek M, and Summers RM. Teniae coli guided navigation and registration for virtual colonoscopy. IEEE Visualization, pp. 279–285, 10 2005.

[15]. Kovnatsky A, Bronstein MM, Bronstein AM, Glashoff K, and Kimmel R. Coupled quasi-harmonic bases. Computer Graphics Forum, 32(24):439–448, 2013.

[16]. Lai Z, Hu J, Liu C, Taimouri V, Pai D, Zhu J, Xu J, and Hua J. Intra-patient supine-prone colon registration in CT colonography using shape spectrum. Medical Image Computing and Computer-Assisted Intervention, pp. 332–339, 2010. [PubMed: 20879248]

[17]. Li P, Napel S, Acar B, Paik DS, Jeffrey RB Jr., and Beaulieu CF. Registration of central paths and colonic polyps between supine and prone scans in computed tomography colonography: Pilot study. Medical Physics, 31(10):2912–23, 10 2004. [PubMed: 15543800]

[18]. Liu L, Zhang L, Xu Y, Gotsman C, and Gortler SJ. A local/global approach to mesh parameterization. Computer Graphics Forum, 27(5):1495–1504, 2008.

[19]. Liu R and Zhang H. Segmentation of 3D meshes through spectral clustering. Pacific Conference on Computer Graphics and Applications, pp. 298–305, 2004.

[20]. Marino J and Kaufman A. Planar visualization of treelike structures. IEEE Transactions on Visualization and Computer Graphics, 22(1):906–915, 2016. [PubMed: 26529735]

[21]. Marino J, Zeng W, Gu X, and Kaufman A. Context preserving maps of tubular structures. IEEE Transactions on Visualization and Computer Graphics, 17(12):1997–2004, 2011. [PubMed: 22034317]

[22]. Meyer M, Desbrun M, Schroder P, and Barr AH. Discrete differential-geometry operators for triangulated 2-manifolds. Visualization and mathematics III, pp. 35–57, 2003.

[23]. Nadeem S, Su Z, Zeng W, Kaufman A, and Gu X. Spherical parameterization balancing angle and area distortions. IEEE Transactions on Visualization and Computer Graphics, PP(99):1–14, 2016.

[24]. Nain D, Haker S, Grimson WEL, Cosman E Jr., Wells WW, Ji H, Kikinis R, and Westin CF. Intra-patient prone to supine colon registration for synchronized virtual colonoscopy. Proc. of MICCAI, pp. 573–580, 2002.

[25]. Ovsjanikov M, Ben-Chen M, Solomon J, Butscher A, and Guibas L. Functional maps: a flexible representation of maps between shapes. ACM Transactions on Graphics (TOG), 31(4):30, 2012.

[26]. Plishker W and Shekhar R. Virtual colonoscopy registration regularization with global chainmil. Proc. of MICCAI Workshop on Virtual Colonoscopy, pp. 116–121, 2008.

[27]. Schwartz EL and Merker B. Computer-aided neuroanatomy: Differential geometry of cortical surfaces and an optimal flattening algorithm. IEEE Computer Graphics and Applications, 6(3):36–44, 1986.

[28]. Shi J and Malik J. Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(8):888–905, 2000.

[29]. Suh JW and Wyatt CL. Deformable registration of supine and prone colons for computed tomographic colonography. Journal of Computer Assisted Tomography, 33(6):902–911, 11 2009. [PubMed: 19940658]

[30]. Umemoto Y, Oda M, Kitasaka T, Mori K, Hayashi Y, Suenaga Y, Takayama T, and Natori H. Extraction of teniae coli from CT volumes for assisting virtual colonoscopy. Proc. of SPIE Medical Imaging, 6916:69160–D–1–69160–7, 2008.

[31]. Wang S, Yao J, Liu J, Petrick N, Van Uitert RL, Periaswamy S, and Summers RM. Registration of prone and supine CT colonography scans using correlation optimized warping and canonical correlation analysis. Medical Physics, 36(12):5595–5603, 2009. [PubMed: 20095272]

[32]. Wang Z, Liang Z, Li L, Li B, Eremina D, and Lu H. An improved electronic colon cleansing method for detection of colonic polyps by virtual colonoscopy. IEEE Transactions on Biomedical Engineering, 53(8):1635–46, 8 2006. [PubMed: 16916098]

[33]. Wei Z, Yao J, Wang S, and Summers RM. Teniae coli extraction in human colon for computed tomographic colonography images. Virtual Colonoscopy and Abdominal Imaging. Computational Challenges and Clinical Opportunities, pp. 98–104, 2010.

[34]. Yao J and Summers RM. Detection and segmentation of colonic polyps on haustral folds. IEEE International Symposium on Biomedical Imaging: From Nano to Macro, pp. 900–903, 2007.

[35]. Zeng W, Marino J, Gurijala KC, Gu X, and Kaufman A. Supine and prone colon registration using quasi-conformal mapping. IEEE Transactions on Visualization and Computer Graphics, 16(6):1348–1357, 2010. [PubMed: 20975175]

[36]. Zhao L, Botha CP, Bescos JO, Truyen R, Vos FM, and Post FH. Lines of curvature for polyp detection in virtual colonoscopy. IEEE Transactions on Visualization and Computer Graphics, 12(5):885–892, 9 2006. [PubMed: 17080813]

[37]. Zhu H, Barish M, Pickhardt P, and Liang Z. Haustral fold segmentation with curvature-guided level set evolution. IEEE Transactions on Biomedical Engineering, 60(2):321–331, 2013. [PubMed: 23193228]

[38]. Zhu H, Li L, Fan Y, Lin Q, Lu H, Gu X, and Liang Z. Automatic teniae coli detection for computed tomography colonography. SPIE Medical Imaging, 7963:79632N–1–79632N–7, 2011.
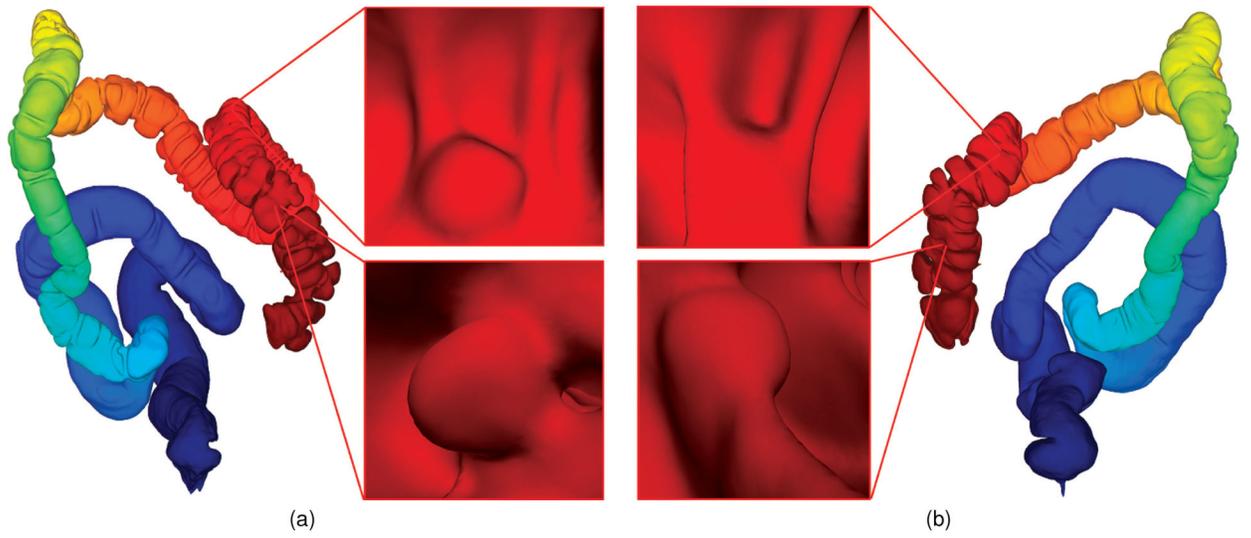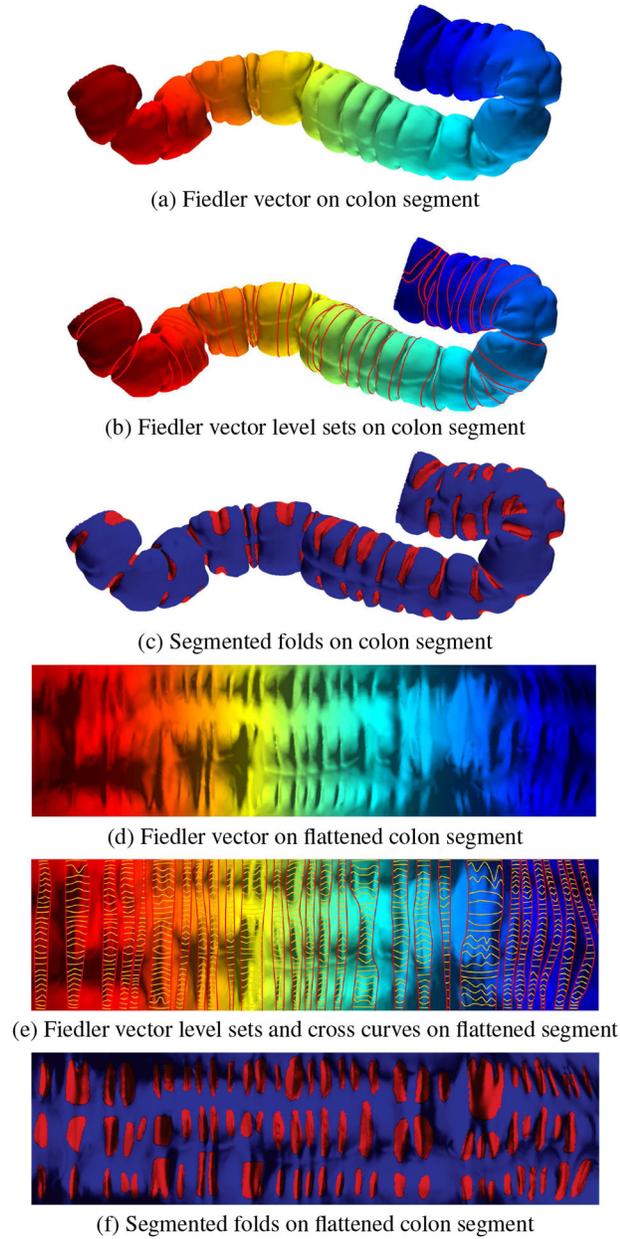
Fig. 1:
Corresponding supine and prone colon visualization. (a) A supine colon model with Fiedler vector representation and two polyps. (b) Registered prone model with corresponding polyps correlated in the endoluminal view with those in the supine model.
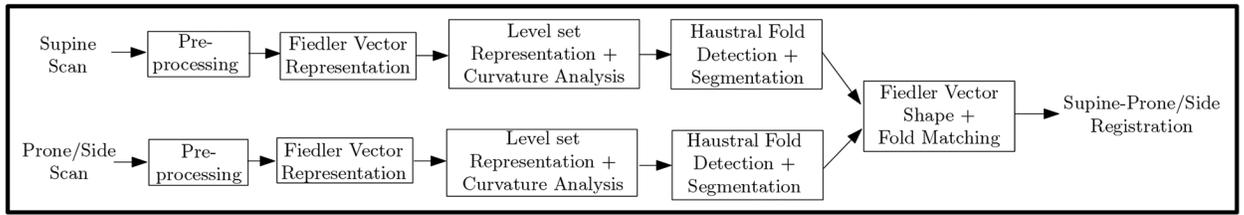
Fig. 2:
The pipeline for supine-prone/side registration.
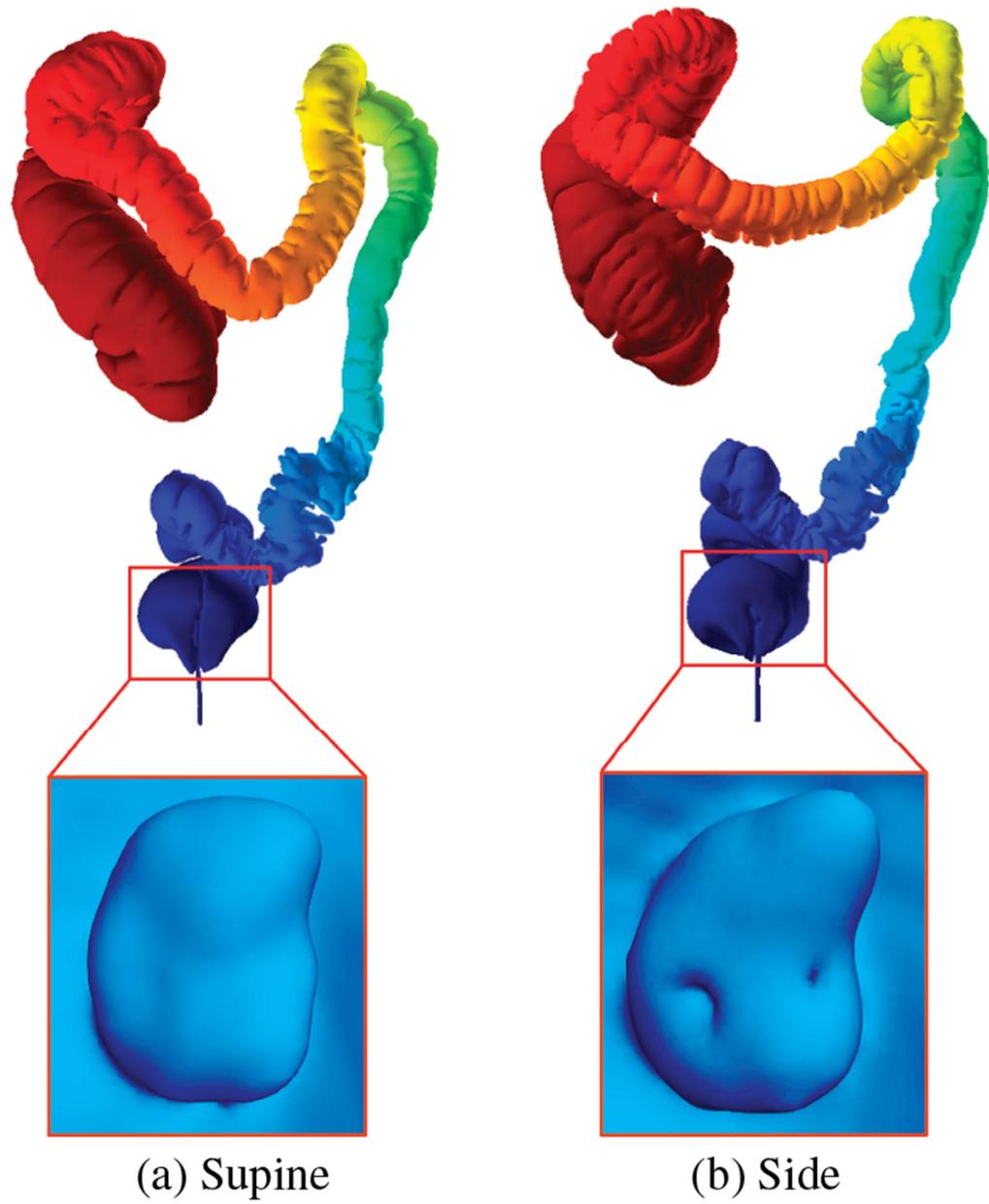
(a) Supine (b) Side

Fig. 3:
Fiedler vector representation for (a) supine and (b) side colon models with corresponding polyps detected in the rectum.

(a) Fiedler vector on colon segment

(b) Fiedler vector level sets on colon segment

(c) Segmented folds on colon segment

(d) Fiedler vector on flattened colon segment

(e) Fiedler vector level sets and cross curves on flattened segment

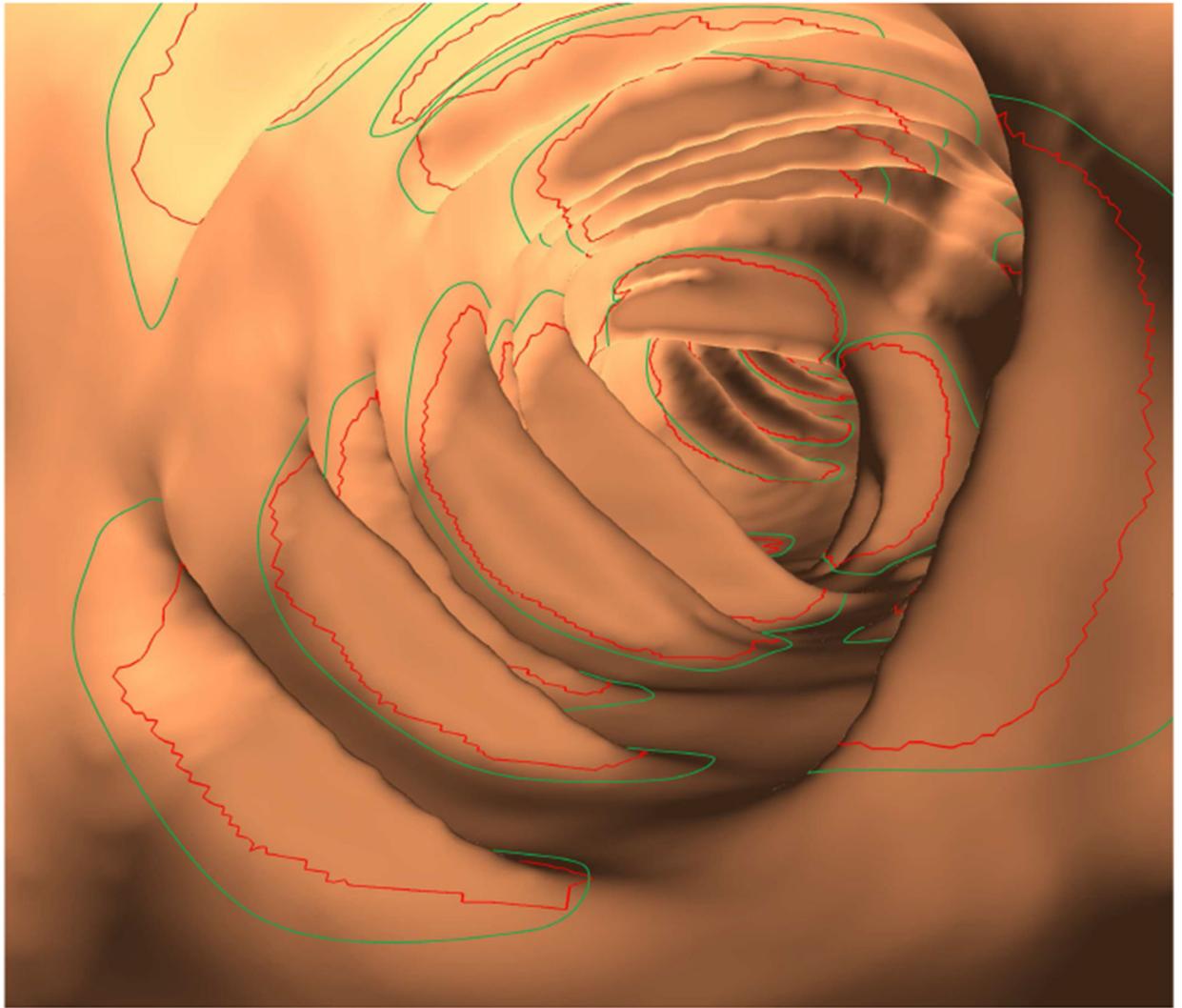(f) Segmented folds on flattened colon segment

Fig. 4:

Haustral fold detection and segmentation pipeline. The computed Fiedler vector is shown for a colon segment in the (a) 3D view and (d) corresponding flattened view. The 3D level sets, illustrated with red lines, are computed to detect fold contours and are shown in the (c) 3D view and together with cross curves, illustrated in yellow lines, in the (e) flattened view. Using these, the haustral folds are segmented, as shown in the (c) 3D view and (f) corresponding flattened view.

(a)   (b)   (c)
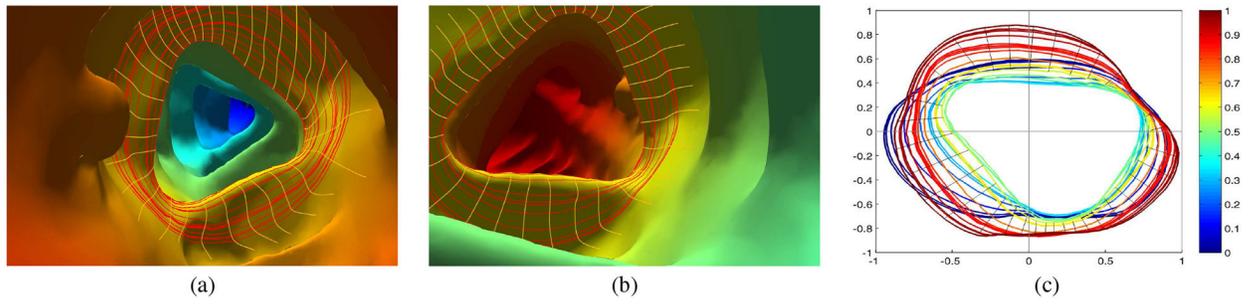
Fig. 5:

Haustral fold detection. (a) and (b) show two endoluminal views of the same colon segment with a 3D level set bundle (red loops) and corresponding cross curves (yellow lines). (c) Level sets projected on planes and normalized in 2D with the black lines representing the cross curves from an extreme level set in a bundle to the middle level set contour.

(a)   (b)   (c)   (d)   (e)

(f) Flattened colon Fiedler vector representation
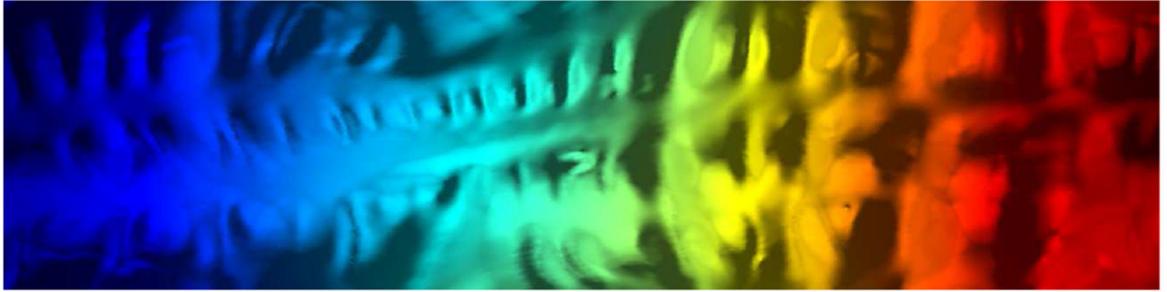
(g) Flattened fold segmentation
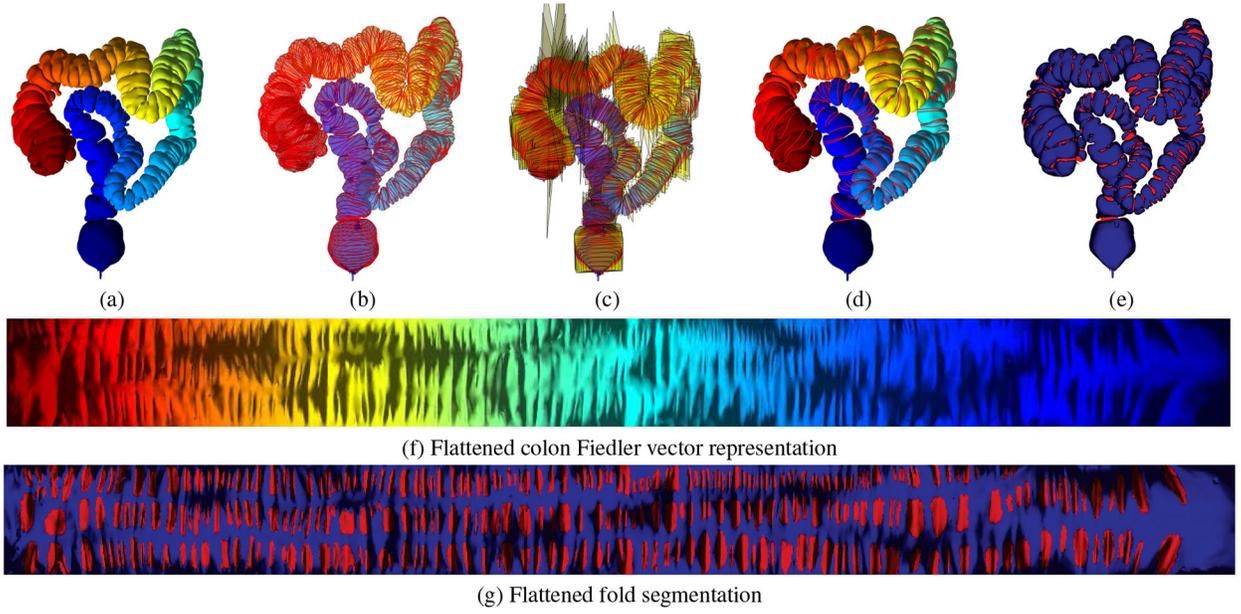
Fig. 6:

Haustral fold detection and segmentation for a complete colon dataset. (a) Fiedler vector computed for a colon dataset. (b) Equi-sampled level sets computed based on (a). (c) Planes fitted to the level sets in (b) with the level sets projected to the planes. (d) Haustral fold contours detected based on the normalization of the planes in 2D. (e) Haustral fold segmentation based on the cross curves shown in Figure 5. (f) Flattened Fiedler vector representation corresponding to the 3D view in (a). (g) Flattened haustral fold segmentation corresponding to the 3D view in (e).
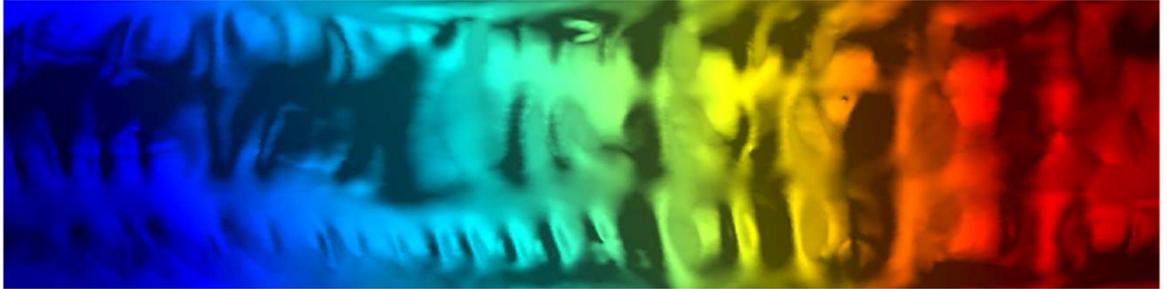
Fig. 7:
Endoluminal view of the haustral fold segmentation with red contours denoting the automatic segmentation and green contours denoting the manual segmentation.
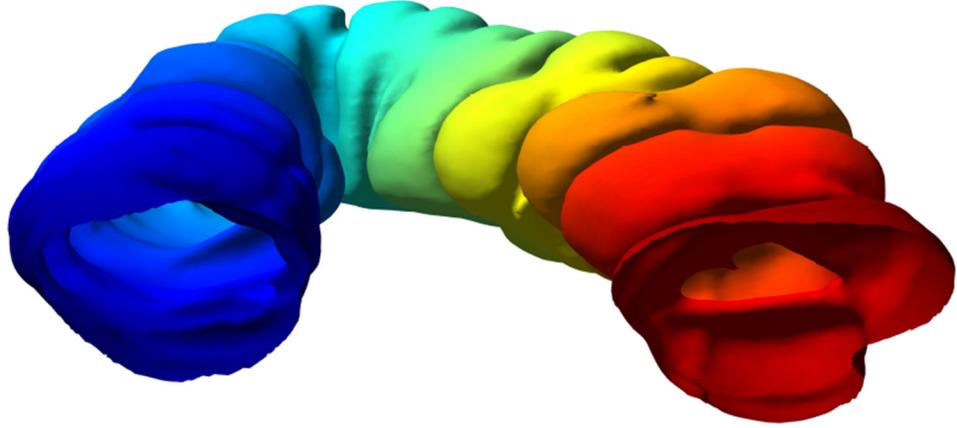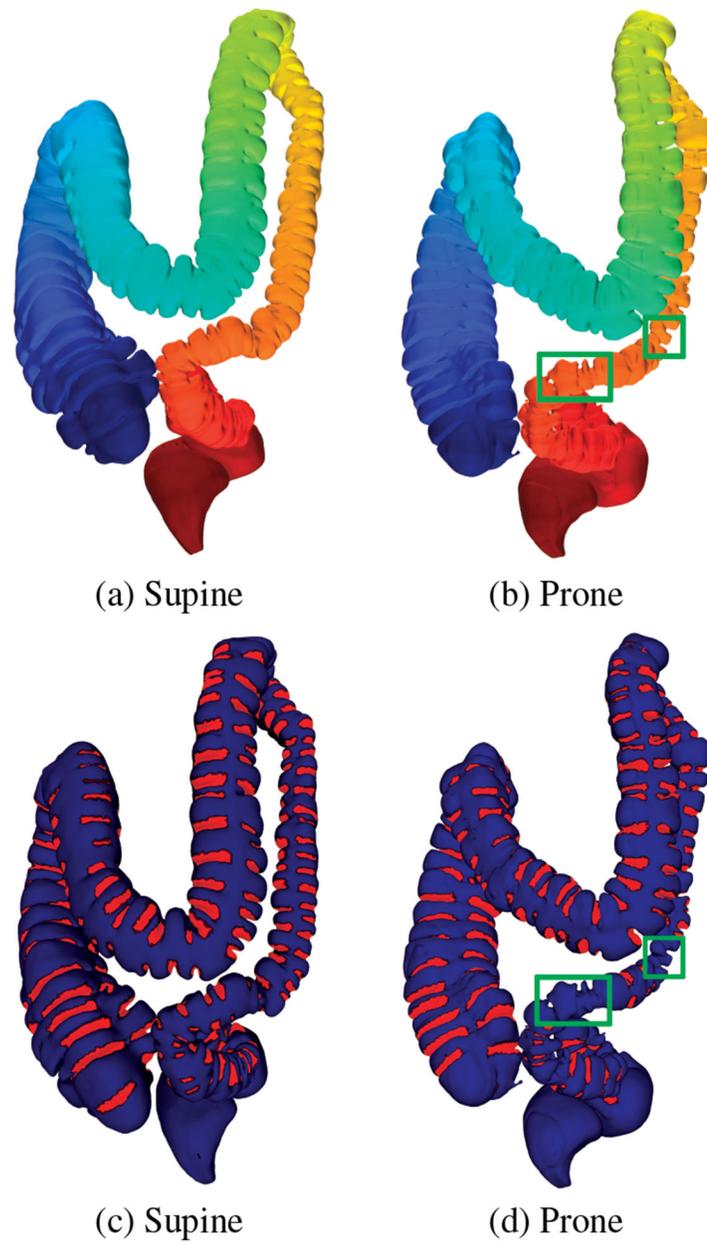
(a) Fiedler vector on colon segment

(b) Flattened colon segment with a cut via geodesic path

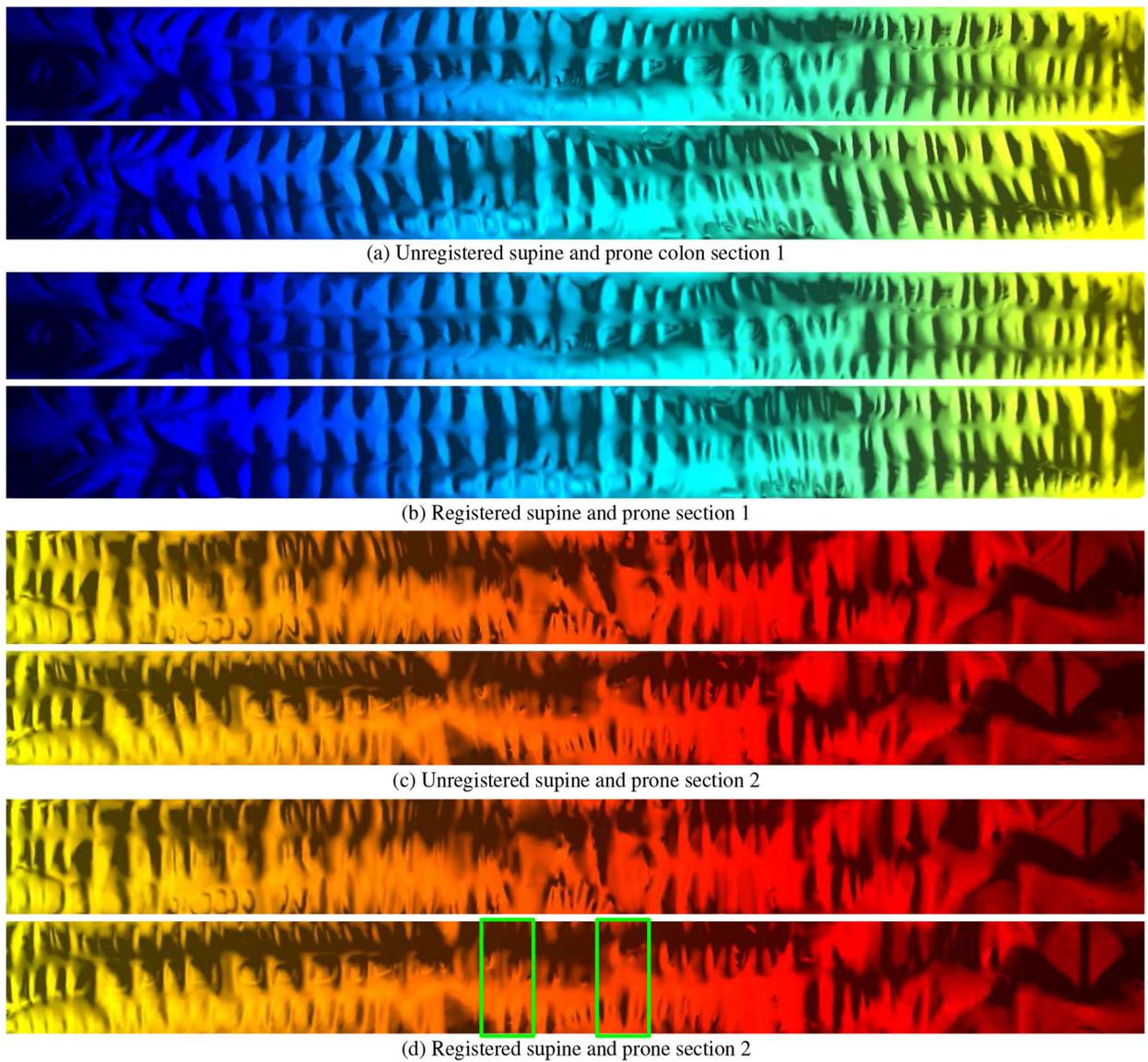(c) Flattened colon segment with a cut via our fold segmentation

Fig. 8:

Consistent cuts. (a) Fiedler vector representation on a colon segment. (b) For two given points on the boundaries, a geodesic path cuts through the folds and chops the folds on the boundaries while flattening. (c) Due to our accurate segmentation, we can trace consistent cuts automatically and keep the haustral folds intact during flattening.

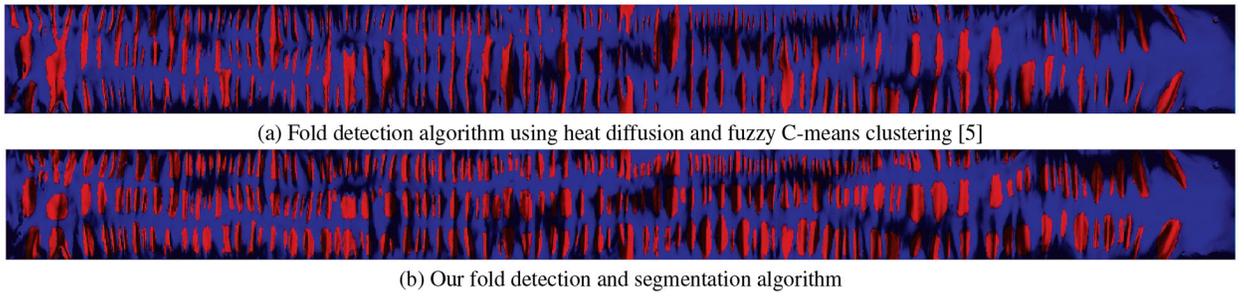(a) Supine   (b) Prone

(c) Supine   (d) Prone

Fig. 9:
Fiedler vector computation for (a) supine and (b) prone colons and the corresponding (c) and (d) fold segmentation results. The green boxes in (b) and (d) show the collapsed regions in the prone dataset.
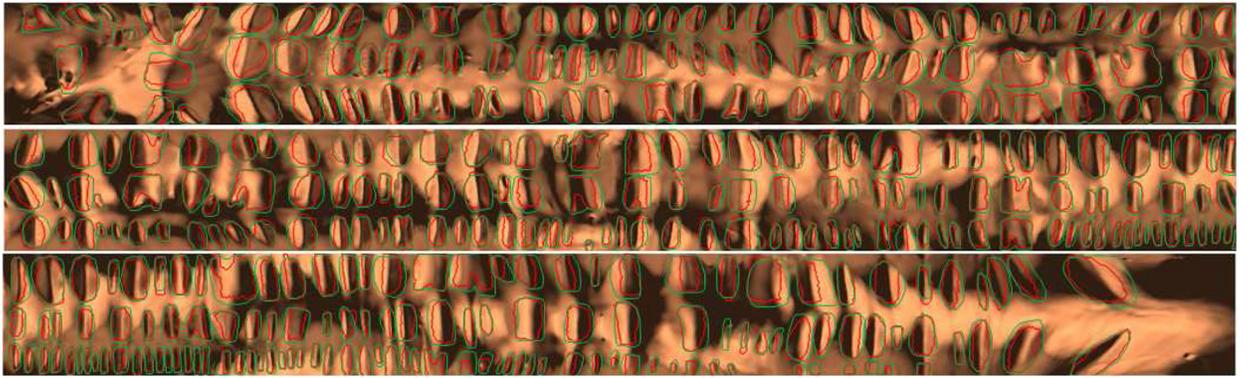
Fig. 10:
Automatic fold segmentation (red contours) and manual fold segmentation (green contours) of a complete colon.

(a) Unregistered supine and prone colon section 1

(b) Registered supine and prone section 1

(c) Unregistered supine and prone section 2

(d) Registered supine and prone section 2

Fig. 11:

Flattened visual registration verification of the supine and prone colons in Figure 9. (a) and (c) The unregistered supine and prone sections. (b) and (d) The registered supine and prone sections. The green boxes in (d) show the collapsed regions corresponding to the collapsed regions highlighted in Figures 9(b) and 9(d).

(a) Fold detection algorithm using heat diffusion and fuzzy C-means clustering [5]

(b) Our fold detection and segmentation algorithm

Fig. 12:
Fold detection evaluation on a flattened colon segment. (a) Fold detection algorithm using heat diffusion and fuzzy C-means clustering [5] and (b) our fold detection algorithm.

(a) Supine

(b) Prone

(c) Supine

(d) Prone

(e) Supine

(f) Side
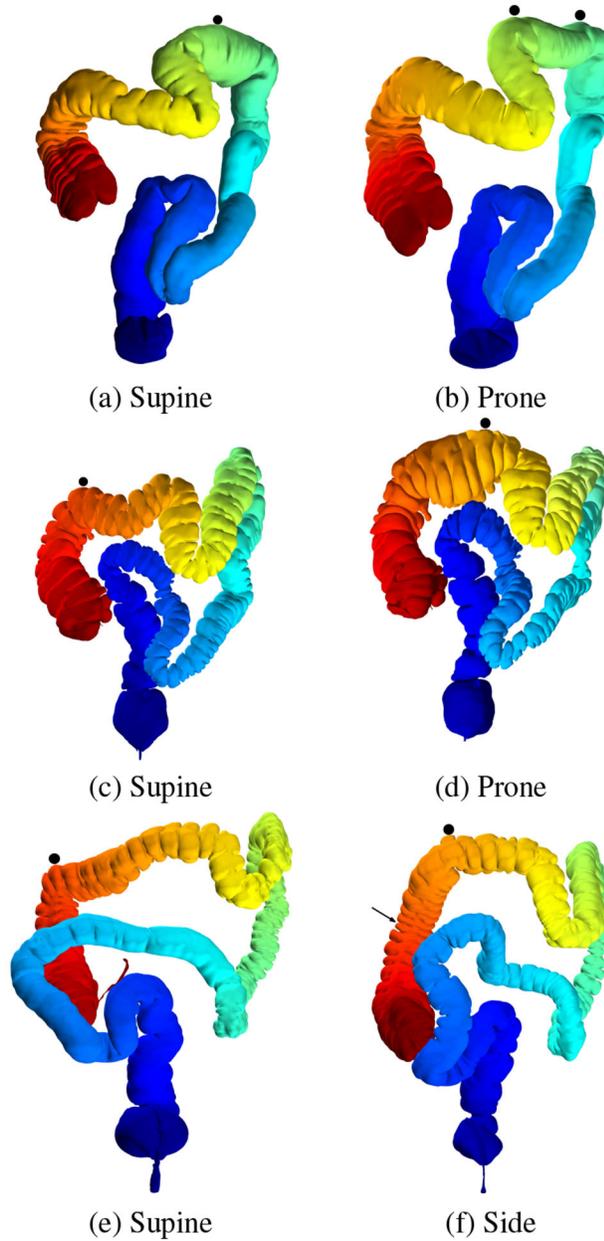
Fig. 13: Registration problems with the Lai et al. [16] algorithm.

Due to the dependence of the Lai et al. registration on correctly detecting flexures, the average distance registration error increases with any $\varepsilon > 0$ if the flexures are not detected correctly on the corresponding colon pairs. The black circles indicate the local maximum z-coordinates as per their approach, and we use $\varepsilon = 0.05$ to compute the corresponding points on the prone/side model. In (b), two local maximum z-coordinates are detected, instead of one (which is the Lai et al. assumption). In (f) the arrow shows the corresponding actual location on the side colon for the point detected in the supine model (e).

Table 1:

Quantitative results from the 20 patient scans. Columns 2–6 indicate for each scan the number of ground truth folds, sensitivity (# true positives divided by # ground truth positives), false negatives, false positives, and the segmented-area ratio. The last row denotes the averages by scan and standard deviations of columns 2–6.

Scan ID	# True	Sensitivity	#FNs	#FPs	SAR
1-Supine	285	0.94	18	0	0.82
1-Side	272	0.94	16	1	0.92
2-Supine	191	0.86	27	0	0.87
2-Side	169	0.83	28	0	0.86
3-Supine	293	0.99	2	0	0.93
3-Prone	268	0.97	7	1	0.81
4-Supine	274	0.93	18	0	0.91
4-Prone	251	0.90	24	0	0.83
5-Supine	165	0.84	26	1	0.86
5-Prone	152	0.82	27	0	0.76
6-Supine	223	0.94	13	0	0.86
6-Prone	202	0.88	25	1	0.76
7-Supine	196	0.88	24	1	0.91
7-Prone	161	0.94	9	0	0.78
8-Supine	247	0.97	8	0	0.94
8-Prone	210	0.91	19	0	0.86
9-Supine	187	0.82	33	1	0.84
9-Prone	158	0.91	15	0	0.75
10-Supine	173	0.87	22	0	0.82
10-Prone	142	0.92	11	0	0.89
Average	211.0 ± 49.4	0.90 ± 0.05	18.6 ± 8.36	0.30 ± 0.47	0.85 ± 0.06

Table 2:

Comparison of average distance error between our Fiedler vector approach (in bold) and other registration methods.

Method	Dist. Error
Fiedler Vector Representation + Fold matching	**5.24 mm**
Quasi-conformal mapping [35]	7.85 mm
Fiedler Vector Representation	**11.98 mm**
Centerline registration + statistical analysis [17]	12.66 mm
Linear stretching/shrinking of centerline [2]	13.20 mm
Centerline feature match + lumen deformation [29]	13.77 mm
Fiedler vector + piecewise registration [16]	14.19 mm
Centerline point correlation [7]	20.00 mm
Taeniae coli correlation [14]	23.33 mm