3D Curve Evolution Algorithm with Tangential Redistribution for a Fully Automatic Finding of an Ideal Camera Path in Virtual Colonoscopy

Karol Mikula and Jozef Urbán

Slovak University of Technology,
Department of Mathematics and Descriptive Geometry
Radlinského 11, 813 68 Bratislava, Slovakia
mikula@math.sk,jozo.urban@gmail.com
http://www.math.sk

Abstract. In this paper we develop new method, based on 3D evolving curves, for finding the optimal trajectory of the camera in the virtual colonoscopy - the medical technology dealing with colon diagnoses by computer. The proposed method consists of three steps: 3D segmentation of the colon from CT images, finding an initial trajectory guess inside the segmented 3D subvolumes, and driving the initial 3D curve to its optimal position. To that goal, the new fast and stable 3D curve evolution algorithm is developed in which the initial curve is driven by the velocity field in the plane normal to the evolving curve, the evolution is regularized by curvature and accompanied by the suitable choice of tangential velocity. Thanks to the asymtotically uniform tangential redistribution of grid points, originally introduced in this paper for 3D evolving curves, and to the fast and stable semi-implicit scheme for solving our proposed intrinsic advection-diffusion PDE, we end up in fast and robust way with the smooth uniformly discretized 3D curve representing the ideal path of the camera in virtual colonoscopy.

Keywords: Virtual colonoscopy, evolving 3D curves, tangential velocity, asymptotically uniform redistribution, distance function, segmentation.

1 Introduction

According to the official evidence from 2007, the colon cancer is the third most spread cancer desease (after the breast and lung cancers) in the countries included in the World Health Organization.

A classical optical colonoscopy is an examination of the colon (large intestine) which can successfully detect colon polyps and colorectal tumours. The examination takes 15-60 minutes and it is performed by a colonoscope which is a flexible tube with a miniature camera and which may also provide a tool for removing a tissue. The colonoscope is introduced into the colon through the rectum, it moves along the colon and a physician can see the situation in the colon on the screen. Because this examination is uncomfortable and painful,

the patient receives medication absorbing the pain or it is done under the general anesthesia. On the other hand, the virtual colonoscopy, introduced in [5], is an examination performed by using the computed tomography (CT). The colon is inflated (with air or CO_2) and then the patient is scanned in two positions (on the abdomen and on the back) by CT. The colon can be viewed similarly to the classical optical colonoscopy, but the physician controls the so-called virtual camera by using its computed "ideal" trajectory. The computation of the ideal path is the important part of the process and new method for its finding is proposed in this paper. It is worth to note that the results achieved by the virtual colonoscopy are comparable to the classical approach [8]. Moreover, the images can be viewed at any time, it provides the option to view panoramas of the colon surface, to make its unfolding etc. In addition to these benefits, the virtual colonoscopy allows to examine colon parts impassable for colonoscope and it avoids risk of a perforation of the colon. A disadvantage is the radiation during CT examination and the fact that if the physician has found a polyp or tumour, it cannot be removed by the virtual approach.

The goal of this paper is to develop new fast and robust method for the fully automatic extraction of the ideal path of the camera in virtual colonoscopy. Here, we understand the ideal path as the 3D curve which passes along the centerline of the colon, which is smooth and which discrete point representation is uniformly distributed. Our proposed method consists of three basic steps. First, the 3D segmentation of the colon from CT images is performed. Due to the quality of CT data, the classical approaches like the thresholding and the region growing are used, see e.g. [1]. As the result we get all simply-connected parts of the large and small intestines filled with the gas. The next step consists in finding an initial guess for the camera trajectory in every simply-connected segmented subvolume of the intestine. Such initial guess is obtained by using the Dijkstra algorithm [3] for computing approximate distance from point sources inside the segmented subvolumes followed by the backtracking in steepest descent direction [13]. The third step is the core of our approach. It consists in driving the initial guess to its optimal position in smooth and stable way. To that goal we use a vector field given by the gradient of distance function to the segmented intestine borders which is computed by a 3D generalization of the approach from [2] based on the numerical solution of the time relaxed eikonal equation. Then, new 3D curve evolution algorithm is developed in which the initial curve is driven by the velocity given by the projection of the computed vector field to the plane normal to the evolving curve, the evolution is regularized by curvature, which makes it smooth, and it is accompanied by the suitable choice of the tangential velocity which makes the curve uniformly discretized during the evolution. In this paper, we develop new asymptotically uniform tangential grid point redistribution method for 3D evolving curves in parametric representation. Our new method is based on ideas from [7], where the authors used special $\kappa_1 - \kappa_2 - \omega - L$ 3D curve evolution formulation. The new method can be also understood as nontrivial generalization of 2D approaches from [9,10,11]. Our final 3D curve evolution model, in the form of an intrinsic advection-diffusion partial differential

equation with a driving force, is solved by the fast and stable semi-implicit scheme and we end up with the smooth uniformly discretized 3D curve representing the ideal trajectory of the camera in virtual colonoscopy. It is worth to note that the overall CPU for all steps in our approach is about 8 seconds on standard PC. Thus the method is highly competitive and it is being implemented into the medical software of the TatraMed spol s r.o., Bratislava company.

2   The Colon Segmentation and the Initial Trajectory Guess

The 3D colon image data sets obtained by CT are given by a sequence of 2D slices (512 × 512 pixels) with a typical slice thickness about 0.75 mm. For our further goals, it is sufficient to subsample data and work with 3D images of the typical size $256 \times 256 \times 400$ voxels. First, we use thresholding corresponding to the air (about -1000 HU) in CT scans and detect all subvolumes filled by the gas. All voxels of these subvolumes get value 1, the others were set to 0. Next, we apply the region growing method in order to find all simply connected parts of the large and small intestines. The first seed is put to a corner of the 3D image and the region growing algorithm finds all voxels outside the body, their value is put to 0, so this subvolume is ignored. Next, we go subsequently through the whole 3D image and the seed for the next region growing is the next voxel found with the value 1. This seed and all voxels found by the region growing get number 2 which is set also as the number of this first inner body subvolume. We continue such procedure until all seeds for the next region growings are found. During the current region growing all detected voxel values are set to the number of the currently segmented subvolume which is given by the increment of the previously detected subvolume number. We also count the number of detected voxels in each subvolume which gives us the approximate size of the segmented structures. The last segmentation step consists in removing all spuriously detected subvolumes inside the body. By the checking of the size, we remove small inner structures filled by the gas (detected e.g. in lungs). Then we compute the distance function of all inner voxels to the border of the segmented subvolume (by the method from section 3) and if the global maximum of the distance function (maximal thickness of the structure) is less then a prescribed threshold we ignore such subvolume (representing e.g. the gas between the body and the CT desk). In such way we end up with one (rarely) or several simply connected subregions of the colon (and also of the small intestine) for which we find then the optimal virtual camera trajectories. The visualization of our segmentation result is presented in Figure 1 left. In order to illustrate our final results, on the right we show the camera trajectories inside segmented subvolumes of both large and small intestines found by our method described in the sequel.

   The initial trajectory guess in any colon subvolume is constructed by computing a distance from a point source by the Dijkstra algorithm (in which the graph edges connecting neighbouring voxels have value 1) followed by the backtracking. First, we take any point of the subvolume and fix the distance at this point to
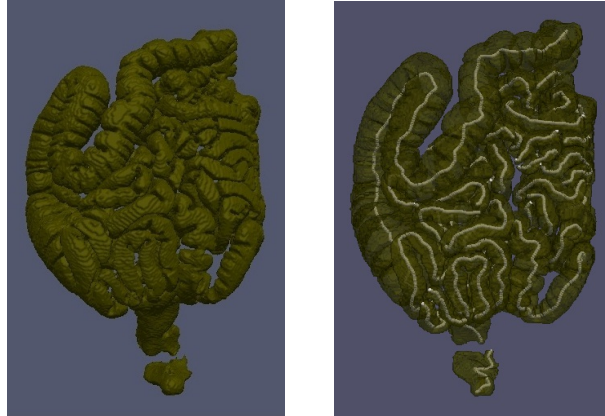
Fig. 1. The segmentation of the large and small intestine (left) and the virtual camera paths visualization for all segmented subvolumes (right)
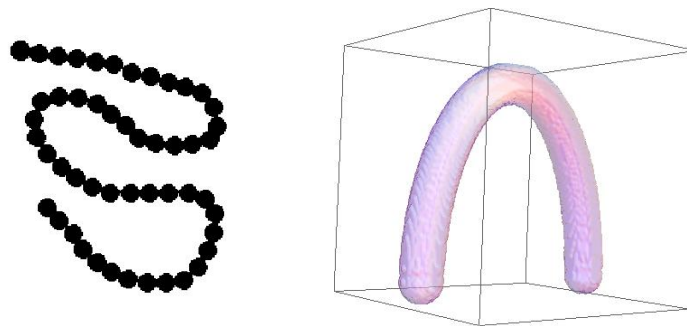

Fig. 2. The 2D and 3D image data sets used for testing the proposed method

0. Then we compute distance in the sense of the Dijkstra algorithm to this fixed point for all voxels inside the subvolume. Since the colon is ablong organ we take a point with the maximal distance as the first endpoint of the segmented subvolume. Now, we fix the zero value at this point and use the Dijkstra algorithm again, a point with the maximal distance will represent the second endpoint of the subvolume. From this second endpoint we start the backtracking of computed distances in the steepest descent direction, we end at the first endpoint of the subvolume. The voxel coordinates of such path represent the parametric 3D curve, the initial guess of the trajectory inside the subvolume. It is worth to note that, for our approach, the only important issue is that the extracted path is a parametric 3D curve inside subvolume which connects two endpoints, its exact localization or smoothness is not important, because it will be improved later by the suitable 3D curve evolution. Also, the very precise localization of the two endpoints does not play any crucial role, because the virtual camera does not really touch the subvolume border in the first and last point. In order to illustrate and test the particular steps of our method, we constructed 2D and

3D artificial data shown in Figure 2, the connected circles on the left mimic the typically alternating very thin and thick colon parts (mostly problematic for the algorithms) and U-like volume on the right mimics an overal colon shape. As one can see in Figures 3-4, the initial trajectory guess is nor smooth nor centered, it touches very often the boundary of segmented volume. On the other hand, it gives the first parametric representation of the 3D curve which can be evolved to the optimal position by the approach discussed in the next section.
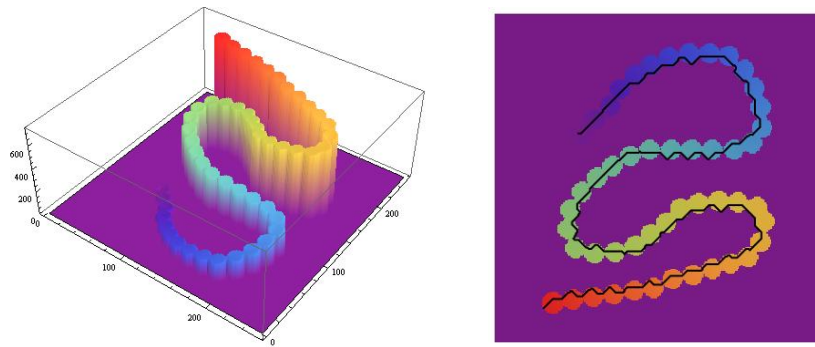
Fig. 3. The graph of the distances (left) and the initial trajectory guess (right)
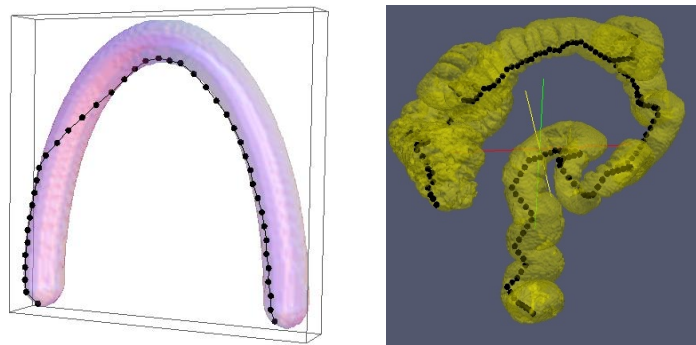
Fig. 4. Initial 3D curve in the test data (left) and in the real segmented colon (right)

3   Finding the Optimal Camera Trajectory

In this section we discuss important issues leading to suitable 3D curve evolution model which will drive the initial curve to its optimal position. Our model will be based on carefull construction of the velocity in normal direction, on the regularization of the motion by curvature and on the suitable tangential velocity yielding the uniform discretization of the evolving curve. We show that all these issues are necessary ingredients in order to get smooth and correctly centered virtual camera trajectory in real colon data of complicated shape.

In order to get the velocity field by which the 3D curve will be moving to its optimal position we solve the eikonal equation with the zero fixed values at the boundary voxels of the segmented subvolume. Its solution is a distance function which has a ridge along the centerline of the segmented subvolume and the gradient of such distance function points towards the ridge. The initial 3D curve should be driven in a smooth way into that ridge position. Our method for finding the distance function, which gives the above mentioned vector field, is based on the numerical solution of the time relaxed eikonal equation $d_t + |\nabla d| = 1$ by the so-called Roy-Tourin scheme [12,2]. Let us denote by d_{ijk}^n approximate solution d in time step n in the middle of voxel with spatial coordinates (i, j, k), τ_D the length of the time step and h_D the size of the voxel. Let us define expressions M_{ijk}^{pqr}, where $p, q, r \in \{-1, 0, 1\}, |p| + |q| + |r| = 1$, by

$$M_{ijk}^{pqr} = (\min(d_{i+p,j+q,k+r}^n - d_{ijk}^n, 0))^2. \tag{1}$$

Then the scheme is given by

$$d_{ijk}^{n+1} = d_{ijk}^n + \tau_D - \tag{2}$$
$$\frac{\tau_D}{h_D}\sqrt{\max(M_{ijk}^{-1,0,0}, M_{ijk}^{1,0,0}) + \max(M_{ijk}^{0,-1,0}, M_{ijk}^{0,1,0}) + \max(M_{ijk}^{0,0,-1}, M_{ijk}^{0,0,1})}.$$

The values at specified points to which the distance is computed numerically are fixed to zero. In all other points the numerical values are increasing monotonically and if they become changeless we can fix them on the fly [2]. Since the colon is an ablong organ, the method (2) is sufficiently fast and easily implementable and applicable to any complicated shape. After computing the distance function we compute the vector field $\mathbf{v} = \nabla d$ by using the central finite difference approximation of the partial derivatives. In Figures 5-6 we show visualization of the computed 2D distance function and the associated vector field.
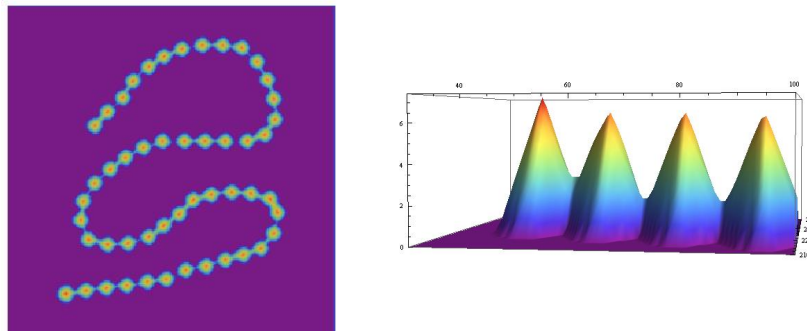
Fig. 5. The distance function to the boundary of 2D testing shape (left) and its detailed graph (right)
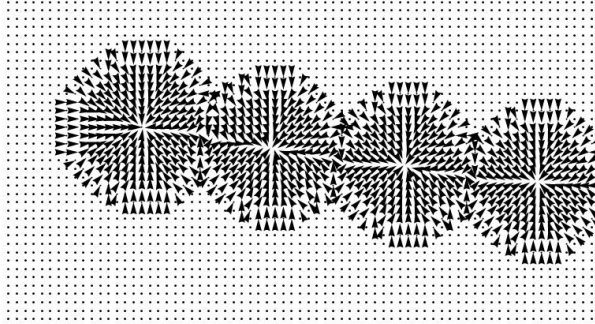
Fig. 6. The detail of the vector field given by the gradient of the distance function

In our parametric approach, the evolving 3D curve is represented by discrete points $\mathbf{r}_i^n = (x_i^n, y_i^n, z_i^n)$, where $i = 0, \ldots, m$, denotes the grid point number and n represents the discrete time stepping. We consider that the endpoints of the curve (i.e. the 0-th and the m-th point) are fixed. The simplest model for the motion of the curve in the vector field $\mathbf{v}$ is given by $\partial_t \mathbf{r} = \mathbf{v}(\mathbf{r})$, numerical discretization of which can be written as $\mathbf{r}_i^{n+1} = \mathbf{r}_i^n + \tau \mathbf{v}(\mathbf{r}_i^n)$, where τ being a discrete time step. The results achieved by this approach can be seen in Figure 7 where all the grid points were moved into the ridge position, but due to the specific direction and length of the velocity field (which is nonzero also on the ridge), they are packed together and thus it is difficult to get smooth virtual camera path by using such final curve state. A group of points may contain curve self-intersections (due to numerics) and their distances are irregular so there is no guarantee that the curve would not cross the edge of the colon on the way between the far-distant points. Such situation is not rare in the real data where the colon has complicated structure similar to our connected circles testing example. We note that many standard approaches to virtual colonoscopy, see e.g. [6,14], uses a combination (e.g. a weighted sum) of two distance functions, the one constructed in the previous section (distance to one fixed endpoint) and the one computed here (distance to the colon borders), followed by a minimization procedure. As we can see, it can lead to a serious troubles in trajectory representation or to a stacking in local minima, which are then solved by some heuristic and/or semi-automatic approaches.

The main difficulty of the above simple approach is given by the fact that the grid points just moved independently on each other by numerical discretization of ODE in direction of the basic velocity field $\mathbf{v}$. There is no mechanism by which the neighbouring points influence each other and thus move smoothly without degeneracy of their distances. All these problems will be solved, without any heuristic, by our new approach described below.

We know that the motion of the curve can be decomposed into the movement in tangential and normal directions and that the overall shape of the evolving curve with the fixed endpoints is determined only by the normal component of the velocity. The tangential velocity influences the redistribution of points along
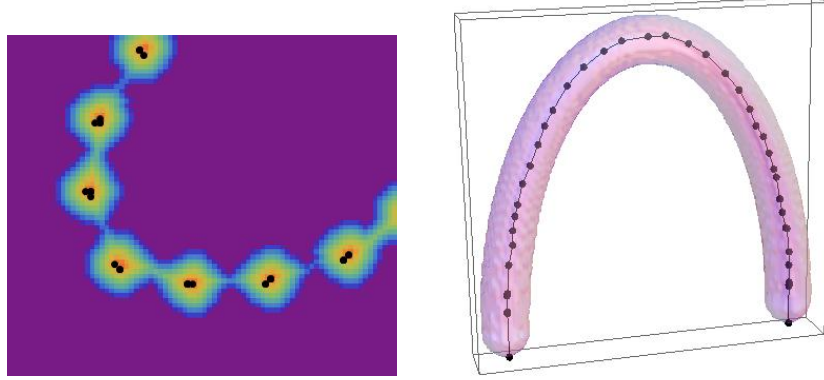
Fig. 7. The results obtained using the velocity field given by the gradient of the distance to the boundary of segmented object in 2D (left) and 3D (right) test data

the curve, thus if it is not controlled, it can cause accumulation of grid points as in the above mentioned examples. As the first modification of the vector field $\mathbf{v}$ we shall consider its projection to the evolving curve normal plane. This makes the model nonlinear (because the curve normal plane depends on the current curve shape) but it greatly improves the result. Moreover, if we want that the evolving curve points are tied together we have to move from the ordinary to a partial differential equation. A natural intrinsic PDE arising in this case is the one obtained by adding the curvature regularization to the motion by using the curvature vector $k\mathbf{N}$ which is again in the curve normal plane. Let $\mathbf{T}$ be the unit tangent vector to the curve, the projection of vector field $\mathbf{v}$ to the curve normal plane is then defined by $\mathbf{N_v} = \mathbf{v} - (\mathbf{T}.\mathbf{v})\mathbf{T}$ and the regularized motion of the curve in the normal plane is given by

$$\partial_t \mathbf{r} = \mu \mathbf{N_v} + \epsilon k \mathbf{N}, \tag{3}$$

where μ and ϵ are the model parameters. Its explicit numerical disretization is

$$\frac{\mathbf{r}_i^{n+1} - \mathbf{r}_i^n}{\tau} = \mu \left(\mathbf{N_v}\right)_i^n + \epsilon \frac{2}{h_{i+1}^n + h_i^n} \left(\frac{\mathbf{r}_{i+1}^n - \mathbf{r}_i^n}{h_{i+1}^n} - \frac{\mathbf{r}_i^n - \mathbf{r}_{i-1}^n}{h_i^n} \right) \tag{4}$$

$i = 1, \ldots, m - 1$, where the second term on the right hand side represents the discretization of the curvature vector $k\mathbf{N}$, see e.g. [4], and the first term is the approximation of the vector $\mathbf{N_v}$ at the i-th curve grid point, both at the previous time step n. The distances between the grid points are given by the expressions

$$h_i^n = \sqrt{(x_i^n - x_{i-1}^n)^2 + (y_i^n - y_{i-1}^n)^2 + (z_i^n - z_{i-1}^n)^2}. \tag{5}$$

Since we removed the improper tangential component of the velocity and used the curvature regularization, the ridge in the testing data is found in much more regular way, see Figure 8. The only problem which is still remaining is the nonuniform distribution of the grid points at the final state and also during the

subsequent curve evolution which may cause problems during the motion inside complicated shapes. The uniform curve representation would guarantee that the properties of the projected vector field are taking into account uniformly and thus the motion of the curve is done in the most correct way. The uniform curve discretization will be controlled by adding a suitable tangential velocity into the mathematical model and its numerical discretization.
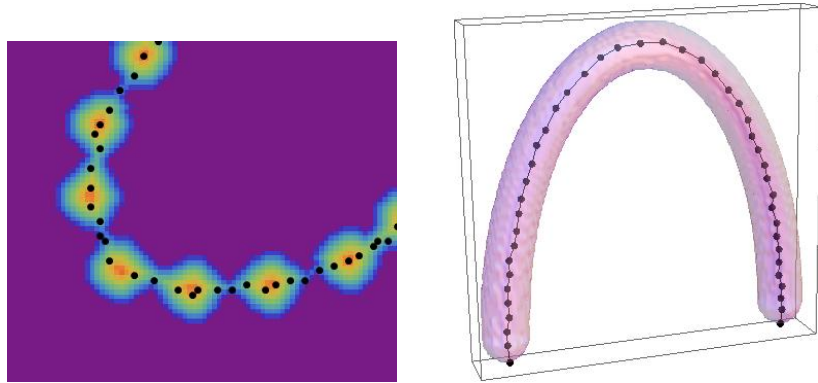
Fig. 8. The results obtained using the projection of the original vector field into the normal plane to the evolving 3D curve accompanied by the curvature regularization

In order to determine suitable tangential velocity we introduce the local orthogonal basis smoothly varying along the 3D curve, cf. [7]. It will consist of $\mathbf{T}$ and two orthogonal vectors in the normal plane defined as $\mathbf{N_1} = \frac{\mathbf{N_v}}{|\mathbf{N_v}|}$ and $\mathbf{N_2} = \mathbf{N_1} \times \mathbf{T}$ (if $|\mathbf{N_v}| = 0$ we redefine $\mathbf{N_1}$ due to the smoothness requirement, e.g. in discrete settings by the averaged value from the neighboring grid points). Let us define $k_1 = k\mathbf{N}.\mathbf{N}_1$ and $k_2 = k\mathbf{N}.\mathbf{N}_2$, the projections of the curvature vector onto $\mathbf{N_1}$ and $\mathbf{N_2}$. Then the curvature vector satisfies $k\mathbf{N} = k_1\mathbf{N_1} + k_2\mathbf{N_2}$ and the evolution equation (3) can be written as

$$\partial_t \mathbf{r} = U\mathbf{N_1} + V\mathbf{N_2} + \alpha\mathbf{T}, \tag{6}$$

with free parameter α representing the tangential component of the velocity, and with the normal components given by

$$U = \epsilon k_1 + \mu|\mathbf{N_v}|, \quad V = \epsilon k_2. \tag{7}$$

Let us consider the curve Γ with fixed endpoints parametrized by the position vector $\mathbf{r}$, and define its local lenght parameter $g = \left|\frac{\partial \mathbf{r}}{\partial u}\right|$, $u \in [0,1]$. In discrete settings it is approximated by $g \approx \frac{|\mathbf{r}_i - \mathbf{r}_{i-1}|}{h}$ with $h = \frac{1}{m}$. One can prove, cf. [7], that the time change of the local length of the evolving 3D curve is given by

$$\partial_t g = g\partial_s \alpha - g(Uk_1 + Vk_2) \tag{8}$$

where s is the arclength parameter. This relation gives us also the time evolution for the total length L

$$\frac{dL}{dt} = - < Uk_1 + Vk_2 >_\Gamma L \tag{9}$$

where $< Uk_1 + Vk_2 >_\Gamma$ denotes the averaged quantity along the curve. In the discrete settings it is computed by

$$< Uk_1 + Vk_2 >_\Gamma^n = \frac{1}{L^n} \sum_{l=1}^{m} h_l^n (U_l^n k_{1l}^n + V_l^n k_{2l}^n), \quad L^n = \sum_{l=1}^{m} h_l^n \tag{10}$$

where $U_l^n, k_{1l}^n, V_l^n, k_{2l}^n$ are approximations of the corresponding quantities on discrete curve segments, cf. (4). In order to define the tangential velocity leading to the asymptotically uniform grid points redistribution, it is worth to study the fraction $\frac{g}{L} \approx \frac{|\mathbf{r}_i - \mathbf{r}_{i-1}|}{Lh} = \frac{|\mathbf{r}_i - \mathbf{r}_{i-1}|}{\left(\frac{L}{m}\right)} = \frac{h_i}{\left(\frac{L}{m}\right)}$ representing in discrete settings the ratio of the actual and averaged lengths of the curve segments. The goal is to desing a model in which this ratio tends to 1 or such that the quantity $\theta = \ln\left(\frac{g}{L}\right)$ converges to 0. Using (8) and (9) we get for its time evolution the relation $\partial_t \theta = \partial_s \alpha - (Uk_1 + Vk_2) + < Uk_1 + Vk_2 >_\Gamma$. On the other hand, if we set $\partial_t \theta = (e^{-\theta} - 1)\omega_r$, where ω_r is a speed of redistribution process, we get that $\theta \to 0$ as $t \to \infty$ and we obtain equation for the tangential velocity α guarateeing the asymptotically uniform redistribution of 3D curve grid points

$$\partial_s \alpha = Uk_1 + Vk_2 - < Uk_1 + Vk_2 >_\Gamma + \left(\frac{L}{g} - 1\right)\omega_r. \tag{11}$$

Since $\mathbf{T} = \partial_s \mathbf{r}$ and $k\mathbf{N} = \partial_{ss}\mathbf{r}$ we get our final 3D curve evolution model in the form of the following intrinsic advection-diffusion PDE with driving force

$$\partial_t \mathbf{r} = \mu \, \mathbf{N_v} + \epsilon \, \partial_{ss}\mathbf{r} + \alpha \partial_s \mathbf{r} \tag{12}$$

with α given by (11) and accompanied by the Dirichlet boundary conditions (fixed endpoints of the curve). Our final step is the numerical discretization of (11)-(12). We proceed similarly to [9,11] and get the discrete tangential velocity

$$\alpha_i^n = \alpha_{i-1}^n + h_i^n (U_i^n k_{1i}^n + V_i^n k_{2i}^n) - h_i^n < Uk_1 + Vk_2 >_\Gamma^n + \left(\frac{L^n}{m} - h_i^n\right)\omega_r, \tag{13}$$

for $i = 1, \ldots, m - 1$, setting $\alpha_0^n = 0$ and getting $\alpha_m^n = 0$. The discretization of the equation (12) is performed by using the semi-implicit scheme and we get

$$\frac{h_{i+1}^n + h_i^n}{2} \frac{\mathbf{r}_i^{n+1} - \mathbf{r}_i^n}{\tau} = \mu \frac{h_{i+1}^n + h_i^n}{2} (\mathbf{N_v})_i^n + \tag{14}$$

$$\epsilon \left(\frac{\mathbf{r}_{i+1}^{n+1} - \mathbf{r}_i^{n+1}}{h_{i+1}^n} - \frac{\mathbf{r}_i^{n+1} - \mathbf{r}_{i-1}^{n+1}}{h_i^n} \right) + \frac{\alpha_i^n}{2}(\mathbf{r}_{i+1}^{n+1} - \mathbf{r}_{i-1}^{n+1}),$$

for $i = 1, \ldots, m-1$, with $\mathbf{r}_0^{n+1}$ and $\mathbf{r}_m^{n+1}$ prescribed. The scheme (14) represents three linear tri-diagonal systems for the x, y, z coordinates of the grid points representing new curve position. They can be written as

$$\mathcal{A}_i^n \mathbf{r}_{i-1}^{n+1} + \mathcal{B}_i^n \mathbf{r}_i^{n+1} + \mathcal{C}_i^n \mathbf{r}_{i+1}^{n+1} = \mathcal{F}_i^n \tag{15}$$

with coefficients given by

$$\mathcal{A}_i^n = -\frac{\epsilon}{h_i^n} + \frac{\alpha_i^n}{2}, \quad \mathcal{C}_i^n = -\frac{\epsilon}{h_{i+1}^n} - \frac{\alpha_i^n}{2}, \quad \mathcal{B}_i^n = \frac{h_i^n + h_{i+1}^n}{2\tau} - \mathcal{A}_i^n - \mathcal{C}_i^n,$$

$$\mathcal{F}_i^n = \frac{h_i^n + h_{i+1}^n}{2\tau} \mathbf{r}_i^n + \mu \left(\mathbf{N_v}\right)_i^n \frac{h_i^n + h_{i+1}^n}{2}.$$

The tri-diagonal systems are solved by the Thomas algorithm which is the fast procedure, numerically stable provided that the system matrix is strictly diagonally dominant. Such property can be simply fulfilled by chosing appropriately the time step, in practice (thanks to the close to uniform discretization) it can be proportional to the spatial discretization step and thus the numerical evolution to the steady state is realized in the fast and stable way. The results for our testing data as well as for the real virtual colonoscopy data are presented in Figures 9 and 11. In presented computations we used parameters $\mu = \epsilon = \tau = \omega_r = 1$ and the method is robust with respect to their choice. The illustrative 2D experiments were performed by the method from [11]. Figure 10 shows differences in grid point distances for our 3D test data (Figures 7-9 right), the red curve for the basic velocity field, the blue for the projected vector field plus curvature regularization and the violet curve for the final model (11)-(12). In the final model the grid point distances are uniform, the final curve is smoothly centered and can be definitely used for the virtual voyage inside the colon (which will be presented in the form of movie at the conference).
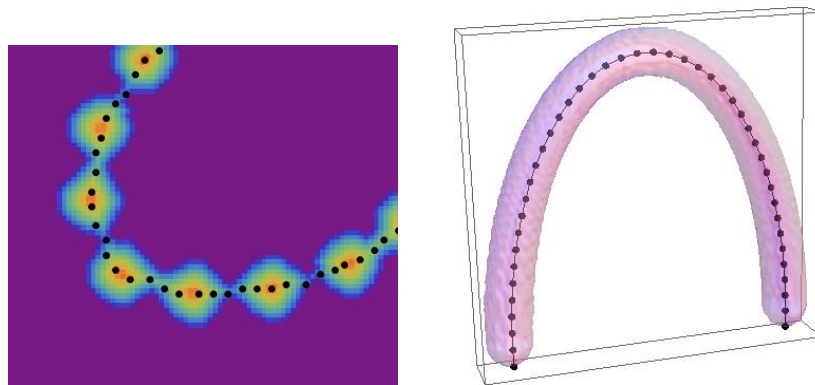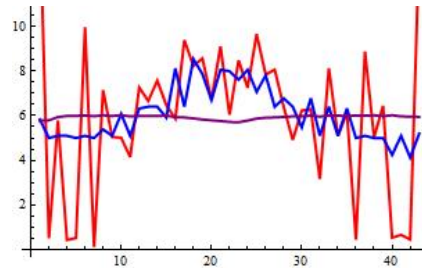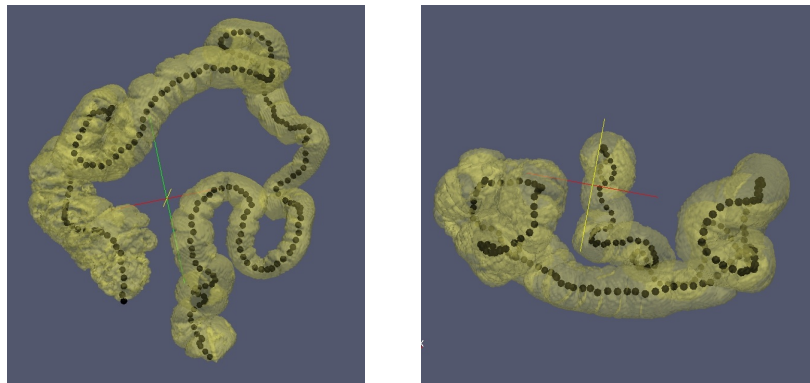

Fig. 9. The results for the test data obtained using tangential redistribution

Fig. 10. Comparison of the grid point distances: the basic vector field (red), the projected vector field plus curvature regularization (blue), the final model (11)-(12) (violet)

Fig. 11. The results for the real data obtained using the final model (11)-(12)

References

1. Bovik, A.: Handbook of Image and Video Processing. Academic Press, London (2000)
2. Bourgine, P., Frolkovič, P., Mikula, K., Peyriéras, N., Remešíková, M.: Extraction of the intercellular skeleton from 2D images of embryogenesis using eikonal equation and advective subjective surface method. In: Tai, X.-C., Mørken, K., Lysaker, M., Lie, K.-A. (eds.) SSVM 2009. LNCS, vol. 5567, pp. 38–49. Springer, Heidelberg (2009)
3. Dijkstra, E.W.: A note on two problems in connection with graphs. Numerische Mathematik 1, 269–271 (1959)
4. Dziuk, G.: Convergence of a semi discrete scheme for the curve shortening flow. Mathematical Models and Methods in Applied Sciences 4, 589–606 (1994)
5. Hong, L., Kaufman, A.E., Wei, Y., Viswambharan, A., Wax, M., Liang, Z.: 3D Virtual Coloscopy. In: IEEE Symposium on Biomedical Visualization, pp. 26–32 (1995)

6. Hong, L., Muraki, S., Kaufman, A.E., Bartz, D., He, T.: Virtual voyage: Interactive navigation in the human colon. In: ACM SIGGRAPH 1997, pp. 27–34 (1997)
7. Hou, T.Y., Klapper, I., Si, H.: Removing the stiffness of curvature in computing 3-D filaments. J. Comput. Physics 143, 628–664 (1998)
8. Lefere, P., Gryspeerdt, S. (eds.): Virtual colonoscopy: A practical guide. Springer, Heidelberg (2010) ISBN 978-3-540-79879-8
9. Mikula, K., Ševčovič, D.: Evolution of plane curves driven by a nonlinear function of curvature and anisotropy. SIAM J. Appl. Math. 61, 1473–1501 (2001)
10. Mikula, K., Ševčovič, D.: A direct method for solving an anisotropic mean curvature flow of planar curve with an external force. Mathematical Methods in Applied Sciences 27, 1545–1565 (2004)
11. Mikula, K., Ševčovič, D., Balažovjech, M.: A simple, fast and stabilized flowing finite volume method for solving general curve evolution equations. Comm. Comp. Physics 7(1), 195–211 (2010)
12. Rouy, E., Tourin, A.: Viscosity solutions approach to shape-from-shading. SIAM Journal on Numerical Analysis 29(3), 867–884 (1992)
13. Truyen, R., Deschamps, T., Cohen, L.D.: Clinical evaluation of an automatic path tracker for virtual colonoscopy. In: Niessen, W.J., Viergever, M.A. (eds.) MICCAI 2001. LNCS, vol. 2208, p. 169. Springer, Heidelberg (2001)
14. Wan, M., Dachille, F., Kaufman, A.E.: Distance-field based skeletons for virtual navigation. In: Proceedings of IEEE Visualization Conference 2001, pp. 239–245 (2001)