New fast and stable Lagrangean method for image segmentation

Karol Mikula
Slovak University of Technology Bratislava,
Slovakia

Jozef Urbán
Slovak University of Technology Bratislava,
Slovakia

***Abstract*—In this paper we present new fast and stable Lagrangean approach to medical image segmentation. The Lagrangean approach consists in discretization of intrinsic partial differential equation for the evolving curve position vector. Since only a curve discretization by grid points is used it can be very fast provided that topological changes which may occur during the curve evolution from the initial guess to a final segmentation result are also resolved in a very fast way. The curve evolution model which we use for the Lagrangean segmentation includes expanding force in the normal direction, the advective term driving the curve from both sides to an edge and the curvature regularization. The numerical procedures are based on stable semi-implicit scheme in curvature part and on inflow-implicit/outflow explicit method in advective part which corresponds to tangential redistribution of grid points. The tangential velocity which is used to stabilize Lagrangean computations keeps the evolving curve uniformly discretized and this fact allows the fast O(n) solution of the topological changes. We present all deatails of our model, numerical procedures and we show their behaviour in medical image segmentation. In our model we do not need to take any special care of initial condition and the segmentation is done in less that 1 second.**

I. Introduction and continuous Lagrangean curve evolution model

In this paper we present new Lagrangean approach to segmentation of objects in medical images. The medical image segmentation is very important for diagnosis and planing of patient treatment. The segmentation results thus should be precise and reliable, as automatic as possible and also fast procedures are highly desirable.

We consider the evolution of segmentation curve driven by the following equation

$$\frac{\partial \mathbf{x}}{\partial t} = \mathbf{V}(\mathbf{x}, t) \ , \tag{1}$$

where $\mathbf{x}$ denotes curve position vector and

$$\mathbf{V}(\mathbf{x}, t) = \mu_1((1-\lambda)g_2\mathbf{N} - \lambda\nabla g_1) + \mu_2 g_3 k\mathbf{N} \ , \tag{2}$$

denotes our velocity vector field where $\mathbf{N}$ denotes the outer unit normal vector of the evolving curve and $k\mathbf{N}$ represents the curvature vector. In the vector field there are also three functions g_1, g_2, g_3 which are used to drive the segmentation curve in the normal direction ($g_2\mathbf{N}$) by the suitable edge attracting advective vector field ($-\nabla g_1$) and by curvature term ($g_3 k\mathbf{N}$). Moreover, we introduced parameters $\mu_1, \mu_2 > 0$ which weight the advection and curvature terms and $\lambda \in [0, 1]$ which weights the edge attracting vector field and the velocity in the normal direction.

The first important term in (2) is $-\nabla g_1$ which attracts the evolved curve to the edge in the image provided that the curve is in the edge local neighbourhood. The function $g_1 = g_1(|\nabla G_\sigma * I^0|, k_1)$

$$g_1(s, k) = \frac{1}{1 + k_1 s^2} \ . \tag{3}$$

depends on a scaling parameter k_1 and on the term $|\nabla G_\sigma * I^0|$ which represents the norm of the image intensity I^0 gradient smoothed by a convolution with a kernel G_σ. The importance of this edge attracting vector field was emphasized in [1], [3] where the so-called geodesic active contours were introduced in the level set formulation, and since then it has been used in many successful image processing models like subjective surfaces and their generalized versions [11]. The regularization by curvature is used to get the final curve more smooth and also with a goal to close the gaps if they arise in the boundaries of segmented objects. If we considered only the image intensity gradient and the curvature terms as above and placed the initial segmentation curve to the region where the vector field is weak, that evolving curve did not move or even worse it could shrink to a point because of influence of the curvature. This phenomenon must be overcame because we want to start the segmentation process just from one user given point inside the segmented object. To that goal, let us denote by ρ the value representing the greylevel intensity of the segmented object. This can be defined as the average of the image intensity values inside initial segmentation curve represented by a small circle around the user specified point. Then we add new term with the function $g_2 = g_2(I^0, \rho, k_2)$ defined as follows

$$g_2(s, \rho, k_2) = \frac{1}{1 + k_2(s-\rho)^2} \tag{4}$$

where k_2 is again a scaling parameter. This term takes bigger values in the regions with intensity values similar to ρ and lower values in other regions with the image intensity different from the indicated mean intensity of the segmented object. By this term we introduced the influence of the image intensity values of the segmented object into our model and evolve the segmentation curve in the normal direction by the speed g_2. Thanks to that we obtain very good results by using the

978-1-4673-0963-9/10/$26.00 ©2012 IEEE

Lagrangean approach although the initial curve is far away from the segmented object edges in the image. It also helps to speed up the segmentation process and approaching the final segmentation curve position.

Since in the continuous Lagrangean approach only the normal component of the movement gives the shape of the evolving curve, it is natural to ignore the tangential component of the edge attracting vector field $-\nabla g_1$ and we can define the curve evolution model (1)-(2) in the form

$$\frac{\partial \mathbf{x}}{\partial t} = \beta \mathbf{N} \ , \tag{5}$$

where the normal velocity β is defined by

$$\beta = \mu_1((1-\lambda)g_2 - \lambda \nabla g_1 \cdot \mathbf{N}) + \mu_2 g_3 k \ . \tag{6}$$

On the other hand, from the numerical point of view it is necessary to add suitable velocity of grid points in tangential direction in order to prevent the evolving curve from selfintersections and to allow fast detection and solution of topological changes. In the next section we present such tangential velocity and derive the discrete stable and fast Lagrangean algorithm which includes detection and solution of the topological changes during the curve evolution.

II. Fast and stable discrete Lagrangean algorithm for image segmentation

In this section we present all details of our Lagrangean approach to segmentation of 2D images by means of plane curve evolution. First we introduce definitions and notations necessary to deal with the evolving curves in Lagrangean formulation and we define our final model including tangential velocity. Then we define asymptotically uniform tangential redistribution of curve representing grid points and derive the discrete numerical scheme for solution of our model which has no restriction on time step regarding solvability of the arising linear systems. Finally we describe how to solve topological changes in evolving curve(s) with O(n) complexity where n is the number of discrete points representing the segmentation curve.

A. Plane curve evolution

Let Γ be the plane curve, $\Gamma : S^1 \to \mathbf{R}^2$, parametrized by $u \in S^1$, where S^1 is a circle with unit length, thus $u \in [0,1]$ and $\Gamma = \{\mathbf{x}(u), u \in S^1\}$, where $\mathbf{x}(u)$ is position vector of the curve Γ for parameter u.

Example of a closed curve discretization is displayed in Figure 1, where $\mathbf{x}_0, \mathbf{x}_1, ..., \mathbf{x}_n$ are discrete curve points which correspond to the uniform discretization of the unit circle with step $h = 1/n$ and $\mathbf{x}_0 = \mathbf{x}_n$. Let $|\mathbf{x}_u| > 0$, where $\mathbf{x}_u = \left(\frac{d\mathbf{x}_1}{du}, \frac{d\mathbf{x}_2}{du}\right)$ and $g = |\mathbf{x}_u| = \sqrt{\left(\frac{d\mathbf{x}_1}{du}\right)^2 + \left(\frac{d\mathbf{x}_2}{du}\right)^2}$. Let us denote by s the unit arc-length parametrization of the curve Γ. Then $ds = |\mathbf{x}_u|du = gdu$ and $du = \frac{1}{g}ds$. The unique definition of the unit tangent $\mathbf{T}$ and normal $\mathbf{N}$ vectors to the plane curve Γ can be done as follows: $\mathbf{T} = \frac{\partial \mathbf{x}}{\partial s}(= \partial_s \mathbf{x} = \mathbf{x}_s)$, $\mathbf{N} = \mathbf{x}_s^{\perp}$ and $\mathbf{T} \wedge \mathbf{N} = 1$, where $\mathbf{T} \wedge \mathbf{N}$ denotes the determinant of the matrix with columns $\mathbf{T}$ and $\mathbf{N}$. From the Frenet-Serret formulas we have $\mathbf{T}_s = k\mathbf{N}$ and $\mathbf{N}_s = -k\mathbf{T}$, where k is the curvature. From there it follows that $k\mathbf{N} = \mathbf{T}_s = (\mathbf{x}_s)_s = \mathbf{x}_{ss}$.

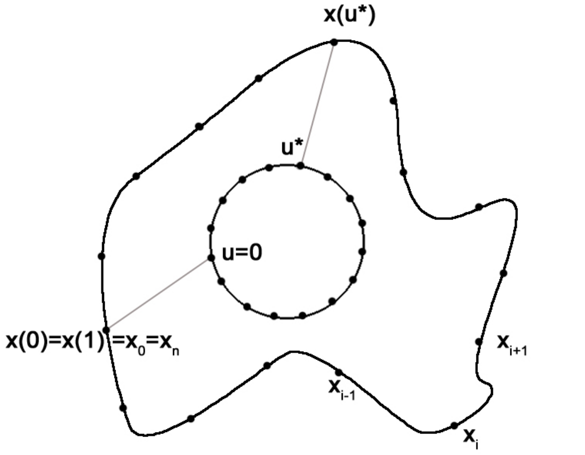

Fig. 1. Curve discretization corresponding to unit circle distribution

In our approach, the curve Γ is given by its position vector $\mathbf{x}$, so its evolution can be described by the evolution of this vector in time. The motion of any point on the curve can be decomposed into the normal and tangential directions. Although it is well-known that the tangential motion does not change the shape of the evolving curve, on the other hand we know that it is helpful in stabilization of the numerical algorithms based on the Lagrangean approaches. So we consider a general form of the curve evolution in the form

$$\mathbf{x}_t = \beta \mathbf{N} + \alpha \mathbf{T}. \tag{7}$$

where $\beta = w + \varepsilon k$, where $w = \mu_1((1-\lambda)g_2 - \lambda \nabla g_1 \cdot \mathbf{N})$, $\varepsilon = \mu_2 g_3$ and α is a "free" parameter, which can be suitably chosen as discussed in the next subsection. By using the above definitions and considerations we get our curve evolution model in the form of intrinsic partial differential equation for the curve position vector $\mathbf{x}$

$$\begin{aligned} \mathbf{x}_t &= (w + \varepsilon k)\mathbf{N} + \alpha \mathbf{T} = w\mathbf{N} + \varepsilon k \mathbf{N} + \alpha \mathbf{T} = \\ & \varepsilon \mathbf{x}_{ss} + \alpha \mathbf{x}_s + w \mathbf{x}_s^{\perp}, \end{aligned} \tag{8}$$

where $\mathbf{x} = (x_1, x_2)$, and therefore (8) is a system of two equations for the components of the position vector x_1 and x_2. These equations are coupled together by the derivative with respect to s, because both components of the position vector $\mathbf{x}$ occur in the term ds. The curvature term represents the intrinsic diffusion along the curve, the tangential velocity term represents the intrinsic advection along the curve and the external driving forces in the normal direction are given by the third term of the right hand side of (8).

B. Suitable choice of tangential velocity α

In this subsection we present a derivation of the tangential component of the velocity that causes asymptotically uniform redistribution of curve grid points which stabilizes the discrete

Lagrangean numerical scheme and allows fast detection and solution of topological changes in the Lagrangean approach.

To that goal, we derive an equation for the evolution of $g = |\mathbf{x}_u|$ which has in discrete scheme an important relation to distances between neighbouring curve grid points. For its time evolution we can write

$$g_t = |\mathbf{x}_u|_t = \frac{\mathbf{x}_u}{|\mathbf{x}_u|} \cdot (\mathbf{x}_u)_t. \tag{9}$$

For a smooth curve evolution, by using the general curve evolution equation (7), we obtain

$$(\mathbf{x}_u)_t = (\mathbf{x}_t)_u = (\beta \mathbf{N} + \alpha \mathbf{T})_u = g(\beta \mathbf{N} + \alpha \mathbf{T})_s.$$

The Frenet-Serret formulas yield

$$\begin{aligned} g(\beta \mathbf{N} + \alpha \mathbf{T})_s &= g(\beta_s \mathbf{N} + \beta \mathbf{N}_s + \alpha_s \mathbf{T} + \alpha \mathbf{T}_s) = \\ & \quad g((\beta_s + \alpha k)\mathbf{N} + (-\beta k + \alpha_s)\mathbf{T}). \end{aligned}$$

Since

$$\mathbf{x}_u = g\mathbf{x}_s = g\mathbf{T}. \tag{10}$$

we can derive the evolution equation for g as follows

$$\begin{aligned} g_t &= \frac{\mathbf{x}_u \cdot (\mathbf{x}_u)_t}{|\mathbf{x}_u|} = g\mathbf{T} \cdot \frac{g((\beta_s + \alpha k)\mathbf{N} + (-\beta k + \alpha_s)\mathbf{T})}{g} = \\ & \quad -gk\beta + g\alpha_s = -gk\beta + g\frac{d\alpha}{ds} = \\ & \quad -gk\beta + \frac{d\alpha}{\frac{1}{g}ds} = -gk\beta + \frac{d\alpha}{du} = -gk\beta + \alpha_u. \end{aligned}$$

Thus, for any evolving curve satisfying (7) with arbitrary parameters α and β we get the formula which describes the evolution of its local length g

$$g_t = -gk\beta + g\alpha_s = -gk\beta + \alpha_u. \tag{11}$$

From there we can see that for $\alpha = 0$ we get the simple differential equation $g_t = -rg$, where $r = k\beta$ with the solution

$$g(t) = g(0)e^{-rt}, \tag{12}$$

at any $u \in S^1$. Such exponential decrease of the local lengths causes accumulation of numerical grid points in places with high curvature k and normal velocity β or it can even cause a formation of swallow-tails and curve selfintersections in a naive Lagrangean numerical implementations.

On the other hand, by integration of the equation (11) along the curve we get equation for evolution of the total curve length

$$\int_{S^1} g_t du = \frac{d}{dt}\int_{S^1} g du = -\int_{S^1} gk\beta du + \int_{S^1} \alpha_u du, \tag{13}$$

so for any curve Γ we have

$$\frac{d}{dt}\int_{\Gamma} ds = -\int_{\Gamma} k\beta ds + \alpha(1) - \alpha(0). \tag{14}$$

Let us denote by $L = \int_\Gamma ds$ the total length of the curve and by $\langle k\beta \rangle_\Gamma = \frac{1}{L}\int_\Gamma k\beta ds$ the mean value of $k\beta$. Since $\alpha(1) - \alpha(0) = 0$ for closed curves, we get the ordinary differential equation for the total curve length

$$\frac{dL}{dt} = -L\langle k\beta \rangle_\Gamma. \tag{15}$$

Let us consider a numerical discretization of the ratio $\frac{g}{L}$ with the approximation of $g = \frac{|\mathbf{x}_i - \mathbf{x}_{i-1}|}{h}$. Then we get

$$\frac{g}{L} \approx \frac{\frac{|\mathbf{x}_i - \mathbf{x}_{i-1}|}{h}}{L} = \frac{|\mathbf{x}_i - \mathbf{x}_{i-1}|}{Lh} = \frac{|\mathbf{x}_i - \mathbf{x}_{i-1}|}{\frac{L}{n}}, \tag{16}$$

where the numerator denotes distance between two neighbouring points and denominator a distance of neighbouring points if the curve would be uniformly discretized (since n denotes the total number of curve points and its segments). We can simply see that one can get the curve with uniformly redistributed discrete grid points if such ratio $\frac{|\mathbf{x}_i - \mathbf{x}_{i-1}|}{\frac{L}{n}} \to 1$ for all discrete segments representing distances between neighbouring points. In the continuous formulation we should have that $\frac{g}{L} \to 1$ with increasing time and if we define a quantity $\theta = \ln(\frac{g}{L})$ we need that $\theta \to 0$ with increasing time. Using the above relations we get evolution equation for θ

$$\begin{aligned} \theta_t &= (\ln(\frac{g}{L}))_t = \frac{L}{g}\frac{g_t L - gL_t}{L^2} = \\ & \quad \frac{(-gk\beta + g\alpha_s)L + gL\langle k\beta \rangle_\Gamma}{gL} = \\ & \quad -k\beta + \langle k\beta \rangle_\Gamma + \alpha_s \end{aligned}$$

which can be controlled by the choice of α, or better say by the choice of α_s. If we choose

$$\alpha_s = k\beta - \langle k\beta \rangle_\Gamma, \tag{17}$$

then $\theta_t = 0$ and the initial curve points distribution will be preserved during the evolution, i.e. the tangential velocity α preserves the initial distribution of points [2], [4], [7]. On the other hand, if we choose

$$\alpha_s = k\beta - \langle k\beta \rangle_\Gamma + \omega(e^{-\theta} - 1), \tag{18}$$

then θ fulfils differential equation $\theta_t = \omega(e^{-\theta} - 1)$ and for solution of such differential equation we get $\theta \to 0$, for $t \to \infty$. The relaxation parameter ω controls redistribution velocity and such tangential velocity is called the asymtotically uniform redistribution [8], [9], [10]. Thus, the suitable choice of α is given by the formula

$$\begin{aligned} \alpha_s &= k\beta - \langle k\beta \rangle_\Gamma + \omega(e^{-\ln(\frac{g}{L})} - 1) = \\ &= k\beta - \langle k\beta \rangle_\Gamma + \omega(\frac{L}{g} - 1). \end{aligned} \tag{19}$$

C. Numerical discretization of intrinsic PDE

Let us consider our general intrinsic differential equation (8)

$$\mathbf{x}_t = \varepsilon \mathbf{x}_{ss} + \alpha \mathbf{x}_s + w\mathbf{x}_s^{\perp}. \tag{20}$$

and integrate it on the segment $[\mathbf{x}_{i-\frac{1}{2}}, \mathbf{x}_{i+\frac{1}{2}}]$, where $\mathbf{x}_{i-\frac{1}{2}}$ denotes the middle point between the points $\mathbf{x}_{i-1}$ and $\mathbf{x}_i$, i.e.

$\mathbf{x}_{i-\frac{1}{2}} = \frac{\mathbf{x}_{i-1}+\mathbf{x}_i}{2}$:

$$\int_{\mathbf{x}_{i-\frac{1}{2}}}^{\mathbf{x}_{i+\frac{1}{2}}} \mathbf{x}_t ds = \varepsilon \int_{\mathbf{x}_{i-\frac{1}{2}}}^{\mathbf{x}_{i+\frac{1}{2}}} \mathbf{x}_{ss} ds + \int_{\mathbf{x}_{i-\frac{1}{2}}}^{\mathbf{x}_{i+\frac{1}{2}}} \alpha \mathbf{x}_s ds + \int_{\mathbf{x}_{i-\frac{1}{2}}}^{\mathbf{x}_{i+\frac{1}{2}}} w \mathbf{x}_s^{\perp} ds.$$

Let us denote by $h_i = |\mathbf{x}_i - \mathbf{x}_{i-1}|$ the length of the linear approximation of the i-th discrete curve segment. Then we have $|\mathbf{x}_{i-\frac{1}{2}} - \mathbf{x}_{i+\frac{1}{2}}| = \frac{h_i+h_{i+1}}{2}$. Let us consider that ε, α and w are given by constant values ε_i, α_i and w_i in the curve segment around the point $\mathbf{x}_i$. Let us denote by m the time step numbering and by τ the length of discrete time step. Let us approximate the time derivative by the finite difference. Using the Newton-Leibniz formula and semi-implicit approach we get

$$\frac{h_i^m + h_{i+1}^m}{2} \frac{\mathbf{x}_i^{m+1} - \mathbf{x}_i^m}{\tau} = \varepsilon_i^m [\mathbf{x}_s^{m+1}]_{\mathbf{x}_{i-\frac{1}{2}}}^{\mathbf{x}_{i+\frac{1}{2}}} + \alpha_i^m [\mathbf{x}^{m+1}]_{\mathbf{x}_{i-\frac{1}{2}}}^{\mathbf{x}_{i+\frac{1}{2}}} + w_i^m ([\mathbf{x}^m]_{\mathbf{x}_{i-\frac{1}{2}}}^{\mathbf{x}_{i+\frac{1}{2}}})^{\perp}.$$

from where

$$\frac{h_i^m + h_{i+1}^m}{2} \frac{\mathbf{x}_i^{m+1} - \mathbf{x}_i^m}{\tau} = \varepsilon_i^m [\mathbf{x}_s^{m+1}]_{\mathbf{x}_{i-\frac{1}{2}}}^{\mathbf{x}_{i+\frac{1}{2}}} + \alpha_i^m (\mathbf{x}_{i+\frac{1}{2}}^{m+1} - \mathbf{x}_{i-\frac{1}{2}}^{m+1}) + w_i^m (\mathbf{x}_{i+\frac{1}{2}}^m - \mathbf{x}_{i-\frac{1}{2}}^m)^{\perp}$$

and by approximating the arc-length derivative in the first bracket on the right hand side by the finite difference we obtain the most simple semi-implicit scheme, cf. [9],

$$\frac{h_i^m + h_{i+1}^m}{2} \frac{\mathbf{x}_i^{m+1} - \mathbf{x}_i^m}{\tau} = \varepsilon_i^m \left(\frac{\mathbf{x}_{i+1}^{m+1} - \mathbf{x}_i^{m+1}}{h_{i+1}^m} - \frac{\mathbf{x}_i^{m+1} - \mathbf{x}_{i-1}^{m+1}}{h_i^m} \right) + \alpha_i^m \left(\frac{\mathbf{x}_{i+1}^{m+1} - \mathbf{x}_{i-1}^{m+1}}{2} \right) + w_i^m \left(\frac{\mathbf{x}_{i+1}^m - \mathbf{x}_{i-1}^m}{2} \right)^{\perp}. \tag{21}$$

As one can see, we have obtained two cyclic tridiagonal systems of linear equations for position vector components $\mathbf{x}_i^{m+1} = ((\mathbf{x}_i^{m+1})_1, (\mathbf{x}_i^{m+1})_2)$, where $i = 1, ..., n$ and $\mathbf{x}_0^{m+1} = \mathbf{x}_n^{m+1}$ resp. $\mathbf{x}_{n+1}^{m+1} = \mathbf{x}_1^{m+1}$.

The tangential velocity α_i^m is computed as follows. We set $\alpha_0^m = 0$, i.e. the point $\mathbf{x}_0$ will not be moving in tangential direction, but only in the normal direction. Then we get disretization of (19):

$$\frac{\alpha_i^m - \alpha_{i-1}^m}{h_i^m} = k_i^m \beta_i^m - \langle k\beta \rangle_\Gamma^m + \omega \left(\frac{L^m}{n h_i^m} - 1 \right) \tag{22}$$

from where we obtain

$$\alpha_i^m = \alpha_{i-1}^m + h_i^m k_i^m \beta_i^m - \langle k\beta \rangle_\Gamma^m h_i^m + \omega \left(\frac{L^m}{n} - h_i^m \right) \tag{23}$$

where the curvature k_i^m, the normal component of velocity β_i^m, for $i = 1, ..., n$, the mean value $\langle k\beta \rangle_\Gamma^m$ and the total length L^m are computed by using the following formulas (see [9]):

$$\begin{aligned} k_i^m &= \operatorname{sgn}(R_{i-1} \wedge R_{i+1}) \frac{1}{2h_i^m} \arccos \left(\frac{R_{i+1} . R_{i-1}}{h_{i+1}^m h_{i-1}^m} \right), \\ \beta_i^m &= \frac{\varepsilon_{i-1}^m + \varepsilon_i^m}{2} k_i^m + \frac{w_{i-1}^m + w_i^m}{2}, \\ \langle k\beta \rangle_\Gamma^m &\approx \frac{1}{L^m} \sum_{l=1}^{n} h_l^m k_l^m \beta_l^m, \\ L^m &= \sum_{l=1}^{n} h_l^m, \end{aligned}$$

where $R_i = ((R_i)_1, (R_i)_2)^T = \mathbf{x}_i^{m-1} - \mathbf{x}_{i-1}^{m-1}$.

The above constructed system (21) is cyclic tridiagonal and can be written in the following form

$$-\left(-\frac{\alpha_i^m}{2} + \frac{\varepsilon_i^m}{h_i^m} \right) \mathbf{x}_{i-1}^{m+1} + \left(\frac{h_{i+1}^m + h_i^m}{2\tau} + \frac{\varepsilon_i^m}{h_{i+1}^m} + \frac{\varepsilon_i^m}{h_i^m} \right) \mathbf{x}_i^{m+1} - \left(\frac{\alpha_i^m}{2} + \frac{\varepsilon_i^m}{h_{i+1}^m} \right) \mathbf{x}_{i+1}^{m+1} = \frac{h_{i+1}^m + h_i^m}{2\tau} \mathbf{x}_i^m + w_i^m \left(\frac{\mathbf{x}_{i+1}^m - \mathbf{x}_{i-1}^m}{2} \right)^{\perp} \tag{24}$$

It can be solved by the cyclic tridiagonal solver which uses Sherman-Morrison formula in order to generalize the classical Thomas algorithm to the matrices with non-zero elements in matrix corners. As in the case of the Thomas algorithm, the solvability is guaranteed by the diagonal dominance of the system matrix. In case of (24) it means

$$|a_{ii}| \geq \sum_{i \neq j} |a_{ij}|, \tag{25}$$

where a_{ij} denotes matrix element in the i-th row and j-th column. We can rewrite the system (24) to the simple form

$$-A_i^m \mathbf{x}_{i-1}^{m+1} + B_i^m \mathbf{x}_i^{m+1} - C_i^m \mathbf{x}_{i+1}^{m+1} = D_i^m, \tag{26}$$

where D_i^m is right hand side and $A_i^m = -\frac{\alpha_i^m}{2} + \frac{\varepsilon_i^m}{h_i^m}$, $C_i^m = \frac{\alpha_i^m}{2} + \frac{\varepsilon_i^m}{h_{i+1}^m}$ and $B_i^m = (H_i^m + A_i^m + C_i^m)$, where $H_i^m = \frac{h_{i+1}^m + h_i^m}{2\tau}$. The diagonal dominance (25) yields the condition $|B_i^m| \geq |-A_i^m| + |-C_i^m| = |A_i^m| + |C_i^m|$. Since in the semi-implicit approach the system parameters are fixed from the previous time step this condition can be fulfilled only by the proper choice of the time step τ according to the next condition

$$\tau \leq \frac{1}{2} \frac{h_{i+1}^m + h_i^m}{|\frac{\varepsilon_i^m}{h_i^m} - \frac{\alpha_i^m}{2}| + |\frac{\varepsilon_i^m}{h_{i+1}^m} + \frac{\alpha_i^m}{2}| - (\frac{\varepsilon_i^m}{h_i^m} + \frac{\varepsilon_i^m}{h_{i+1}^m})}, \tag{27}$$

which must be tested for all $i = 1, .., n$ in every time step. Such test is timeconsuming itself and moreover it can give different values of τ for different curves arising e.g. after a topological change. The smallest τ then must be chosen which again slowed down the speed of this semi-implicit scheme. For this reason it is usefull to modify the scheme, in order to get a method which is not constrained by the choice of the length of discrete time step.

D. Numerical scheme without restriction on time step

In this subsection we design new scheme for the Lagrangean curve evolution which is not constrained by the choice of the time step. The presented scheme is motivated by [5], [6]. If A_i^m and C_i^m in (26) were positive, the system would always be diagonally dominant and thus solvable. A negative values of A_i^m and C_i^m can be caused by the intrinsic advection term of the model which corresponds to the tangential velocity α. We will show how to solve this problem by using the so-called inflow-implicit/outflow-explicit scheme I^{2}OE suggested in [5], [6]. The main idea is a splitting of the advection to the inflow and outflow parts. The inflow part contributes by a good sign to the elements out of the matrix diagonal. So we consider this part implicitly. On the other hand, the outflow part we take explicitly. This approach has the second order accuracy in space and time in case of the scalar advection equation and, moreover, for the curve evolution it keeps the grid points on the circle for the constant tangential velocity. Thanks to such scheme we get always diagonally dominant M-matrix and the solvability of the system is not restricted by the choice of time step length τ.

Let us consider the advection equation

$$q_t + v q_x = 0, \tag{28}$$

where $q : \Omega \times [0,T] \to \mathbf{R}$ is an unknow function and $v(x)$ is a scalar velocity field. We solve the equation (28) in a domain $\Omega \subset \mathbf{R}$ and in time interval $[0,T]$. Let us denote by p_i a finite volume (in our case it will be a segment of the curve$[\mathbf{x}_{i-\frac{1}{2}}, \mathbf{x}_{i+\frac{1}{2}}]$ around the point $\mathbf{x}_i$), with length h. The equation (28) can be rewriten as

$$q_t + (vq)_x - q v_x = 0. \tag{29}$$

If we integrate (29) on the finite volume p_i, we get

$$h q_t + v_{i+\frac{1}{2}} \bar{q}_{i+\frac{1}{2}} - v_{i-\frac{1}{2}} \bar{q}_{i-\frac{1}{2}} - \bar{q}_i (v_{i+\frac{1}{2}} - v_{i-\frac{1}{2}}) = 0, \tag{30}$$

$$h q_t + v_{i-\frac{1}{2}} (\bar{q}_i - \bar{q}_{i-\frac{1}{2}}) + (-v_{i+\frac{1}{2}})(\bar{q}_i - \bar{q}_{i+\frac{1}{2}}) = 0, \tag{31}$$

where $v_i = v(x_i)$, $v_{i-\frac{1}{2}} = v(x_{i-\frac{1}{2}})$, $v_{i+\frac{1}{2}} = v(x_{i+\frac{1}{2}})$, and $\bar{q}_i$, $\bar{q}_{i-\frac{1}{2}}$ and $\bar{q}_{i+\frac{1}{2}}$ are representative values of solution inside and on the boundaries of the finite volume p_i. We get these values by a reconstruction from the numerical solution q_i for $i = 1, ...n$, obtained by the scheme.

If $v_{i-\frac{1}{2}} > 0$, it represents the inflow into the finite volume from the left side. If $v_{i+\frac{1}{2}} < 0$, it represents the inflow to the finite volume from the right side. If the signs are opposite, they represent outflows from the finite volume. Let us define

$$\begin{aligned} b^{in}_{i-\frac{1}{2}} &= \max(v_{i-\frac{1}{2}}, 0),\ b^{out}_{i-\frac{1}{2}} = \min(v_{i-\frac{1}{2}}, 0), \\ b^{in}_{i+\frac{1}{2}} &= \max(-v_{i+\frac{1}{2}}, 0),\ b^{out}_{i+\frac{1}{2}} = \min(-v_{i+\frac{1}{2}}, 0). \end{aligned} \tag{32}$$

Let approximate the time derivative by the finite difference and a simple reconstruction $\bar{q}_i^m = q_i^m$, $\bar{q}^m_{i-\frac{1}{2}} = \frac{q_i^m + q_{i-1}^m}{2}$ and $\bar{q}^m_{i+\frac{1}{2}} = \frac{q_i^m + q_{i+1}^m}{2}$. If we take the inflow implicitly and the outflow explicitly we get the I^{2}OE scheme:

$$\tfrac{h}{\tau} q_i^{m+1} + \tfrac{1}{2} b^{in}_{i-\frac{1}{2}} (q_i^{m+1} - q_{i-1}^{m+1}) + \tfrac{1}{2} b^{in}_{i+\frac{1}{2}} (q_i^{m+1} - q_{i+1}^{m+1}) =$$
$$\tfrac{h}{\tau} q_i^m - \tfrac{1}{2} \left(b^{out}_{i-\frac{1}{2}} (q_i^m - q_{i-1}^m) + b^{out}_{i+\frac{1}{2}} (q_i^m - q_{i+1}^m) \right)$$

which can be written in the form

$$\begin{gathered} -\left(\tfrac{1}{2} b^{in}_{i-\frac{1}{2}}\right) q_{i-1}^{m+1} - \left(\tfrac{1}{2} b^{in}_{i+\frac{1}{2}}\right) q_{i+1}^{m+1} + \\ + \left(\tfrac{h}{\tau} + \tfrac{1}{2}\left(b^{in}_{i-\frac{1}{2}} + b^{in}_{i+\frac{1}{2}}\right)\right) q_i^{m+1} = \\ \tfrac{h}{\tau} q_i^m - \tfrac{1}{2}\left(b^{out}_{i-\frac{1}{2}} (q_i^m - q_{i-1}^m) + b^{out}_{i+\frac{1}{2}} (q_i^m - q_{i+1}^m)\right). \end{gathered} \tag{33}$$

Since we have the velocity α_i given in the centers of finite volumes (the points $\mathbf{x}_i$) and not on their boundaries we modify the basic I^{2}OE by the following definition of inflows and outflows

$$\begin{aligned} b^{in}_{i-\frac{1}{2}} &= \max(v_i, 0),\ b^{out}_{i-\frac{1}{2}} = \min(v_i, 0), \\ b^{in}_{i+\frac{1}{2}} &= \max(-v_i, 0),\ b^{out}_{i+\frac{1}{2}} = \min(-v_i, 0). \end{aligned} \tag{34}$$

In the case of curve evolution equation (8), $v_i = -\alpha_i$ and the unknowns are components of the position vector $\mathbf{x}_i$ so we get our final numerical scheme:

$$\begin{gathered} -\left(\frac{\varepsilon_i^m}{h_i^m} + \frac{1}{2} b^{in}_{i-\frac{1}{2}}\right) \mathbf{x}_{i-1}^{m+1} - \left(\frac{\varepsilon_i^m}{h_{i+1}^m} + \frac{1}{2} b^{in}_{i+\frac{1}{2}}\right) \mathbf{x}_{i+1}^{m+1} + \\ \left(\frac{h_{i+1}^m + h_i^m}{2\tau} + \frac{\varepsilon_i^m}{h_{i+1}^m} + \frac{\varepsilon_i^m}{h_i^m} + \frac{1}{2} b^{in}_{i-\frac{1}{2}} + \frac{1}{2} b^{in}_{i+\frac{1}{2}}\right) \mathbf{x}_i^{m+1} = \\ \frac{h_{i+1}^m + h_i^m}{2\tau} \mathbf{x}_i^m - \frac{1}{2} b^{out}_{i-\frac{1}{2}} (\mathbf{x}_i^m - \mathbf{x}_{i-1}^m) - \\ \frac{1}{2} b^{out}_{i+\frac{1}{2}} (\mathbf{x}_i^m - \mathbf{x}_{i+1}^m) + w_i^m \left(\frac{\mathbf{x}_{i+1}^m - \mathbf{x}_{i-1}^m}{2}\right)^{\perp}. \end{gathered} \tag{35}$$

E. Topological changes

By the topological change we mean splitting of the evolving segmentation curve into several separate parts. Such situations can occur during the evolution mainly due to a noise or other spurious inner structures inside the segmented object. To solve the topological changes in the Lagrangean approach is usually very time consuming because the standard approaches has computational complexity O(n^2) where n is the number of curve grid points. Such complexity is due to a standard strategy for the topological change detection which consists in computing distances between all grid points of the curve. Then, if the smallest computed distance is realized not among the neighbouring grid points and it is below a specified threshold, it indicates that the curve should be split to two curves in those points. The number of operations in such approach is $\sum_{i=2}^{n-2} (n-i) = \frac{n^2-3n}{2}$ and it slowes down computing time significantly. The advantage of the Lagrangean approach (in comparison to level-set methods) that the number of unknowns is proportional only to n (in the level-set method it is proportional to a dimension of image, i.e. it is much higher) is lost in evolutions where topological changes may occur.

Our goal is to develop new method for the topological changes detection with much lower complexity. We follow the same strategy to find two not-neighbouring points with the smallest distance below some threshold. But we find this couple in completely different and fast way. As it was emphasized in the previous sections, our curve is asymptotically uniformly discretized which means that all distances are around their mean value. The global length is increasing or decreasing (after a topological change) during the evolution. If the mean distance is greater than 1 (the size of one pixel) we densify the curve (i.e. we put new point in the middle of curve segment and thus the number of points is doubled). On the other hand, if the mean distance is less than 0.4 we coarse the discretization and remove half of the grid points. By this procedure we guarantee that generically there are maximally 3 grid points in one pixel.

Our main new idea is to create a narrow strip (with thickness 1) of pixels along the curve. If there is no topological change occuring, this strip should not be crossed by two distant pieces of the curve. On the other hand, if there are two distant points inside one pixel of this strip, it indicates the topological change. So our algorithm is as follows:

1. we traverse all curve grid points and mark the pixels in which they lie by $j = 0$.

2. we traverse again subsequently all points and ask whether the pixel value j where the point belongs is equal to 0. If yes, set it to i, where i is the number of grid point. If it is not 0, it means that the value in pixel was set by another point j. If $i - j \leq 2$ then go to another point. If $i - j > 2$ then there are more than two points between the points i and j, and i and j belong to one pixel. It is clear that such situation indicates the splitting of the curve.

3. if such splitting was detected, we do a test of distances between the sets of points $\{i-2, i-1, i, i+1, i+2\}$ and $\{j-2, j-1, j, j+1, j+2\}$. If the smallest distance is less than a given threshold (in our case the mean distance) then the curve is split in two points where this smallest distance was computed.

From the above description it is clear that the number of operations is proportional only to number of grid points and thus our algorithm has complexity O(n). Our procedure is illustrated by figure 2.

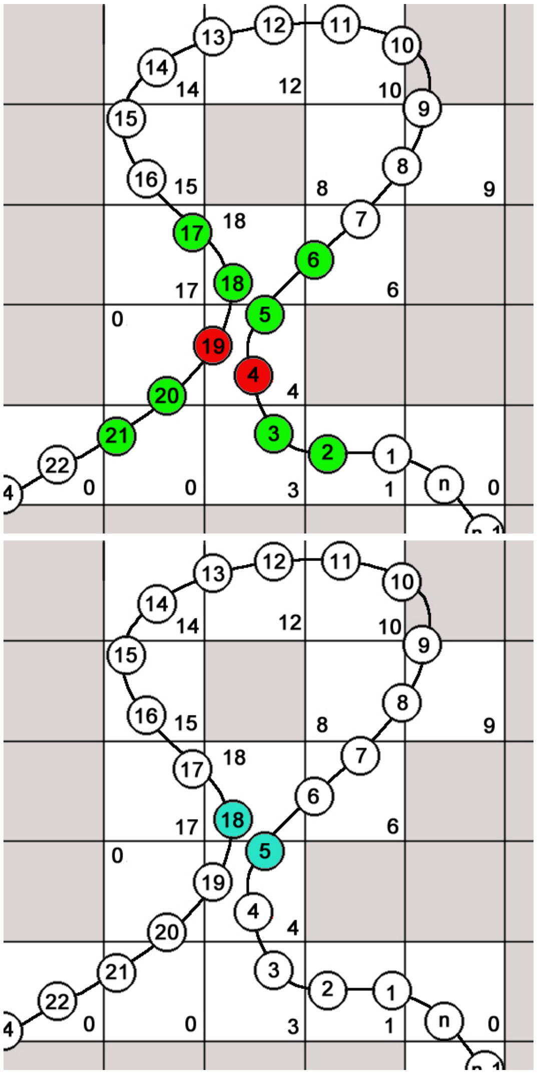

Fig. 2. Detection of topological changes. In the upper figure, the points x_4 and x_{19} are detected as indicators of the topological change, they are distant but in one pixel. The smallest distance in their local neighbourhood is then computed in the points x_5 and x_{18} where the topological change is realized and two new curves are created each of which contains x_5 and x_{18}.

III. Numerical Experiments

In this section we present the results of segmentation of medical images by the proposed mathematical model and numerical scheme. The images are filtered by a few steps of linear diffusion (which corresponds to the Gaussian convolution) and the segmentation is stopped when the curve motion is very small. The coefficients k_1, k_2, μ_1, μ_2 were chosen according to data which are segmented, $g_3 = 1$, and they are not changed in one segmentation process. After the stopping of the curve with one set of parameters, it is possible to run the segmentation process again with modified parameters ε and λ , e.g. one can increase the influence of the curvature or influence of the vector field $-\nabla g_1$. Usually we run first the segmentation with $\varepsilon = 0.001$ and $\lambda = 0.5$. After the first stopping we usually increase ε to 0.1 and then also λ is set to 1.

In the first experiments we have segmented parts of brain in the image with dimensions 512 x 512 pixels.

In figure 3 the segmentation of truncus encepalicus is presented. This segmentation took approximately 0.30 second with $\tau = 0.25$ and we used with 2200 time steps until the final segmentation result was obtained.

In figure 4 we present the segmentation of cerebellum in more detailed way. One can see that during the segmentation process the evolving curve has a complicated shape due to a noise and also that topological changes occur (see the first and third image in the second raw and the first image in the third raw). The small curves arising after the topological changes are shinked to a point due to the curvature term and orientation of normal. Only the largest one stays as the result of segmentation. The segmentation process with one set of model parameters ($\varepsilon = 0.001, \lambda = 0.5$) was stopped when the curve did not move longer in the shape similar to the one from the left image in the third raw. Then new set of parameters was chosen to obtain the final result presented on the right side of the third raw ($\varepsilon = 0.1$). The overal number of time steps with size $\tau = 0.2$ was 2500 and the computations took approximately 0.42 seconds.

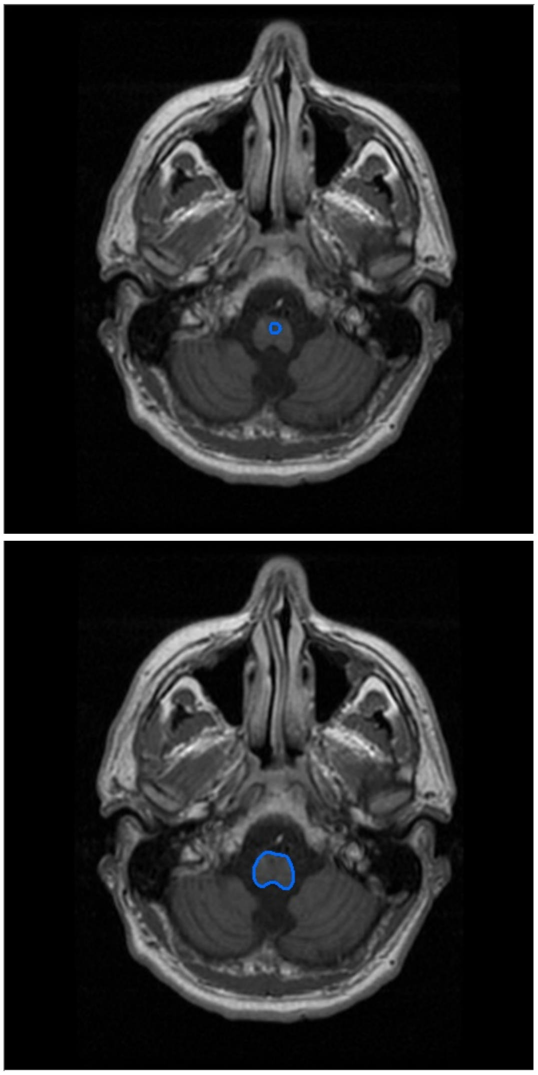

Fig. 3. The initial condition for the segmentation of truncus encepalicus (up) and the result of the segmentation (down).

Next we perform segmentation of tumor of lobus occipitalis lateris dextri telencephali in the image with dimensions 256 x 256 pixels and the result is presented in figure 5. One can see that the segmentation algorithm has reached the complicated tumor shape. This segmentation took 0.15 second ($\tau = 0.25$ and 900 time steps).

Next experiment represents the segmentation of prostate in the image with dimensions 168x168 pixels (figure 6). This segmentation took approximately 0.6 second ($\tau = 0.2$ and 3500 timesteps).

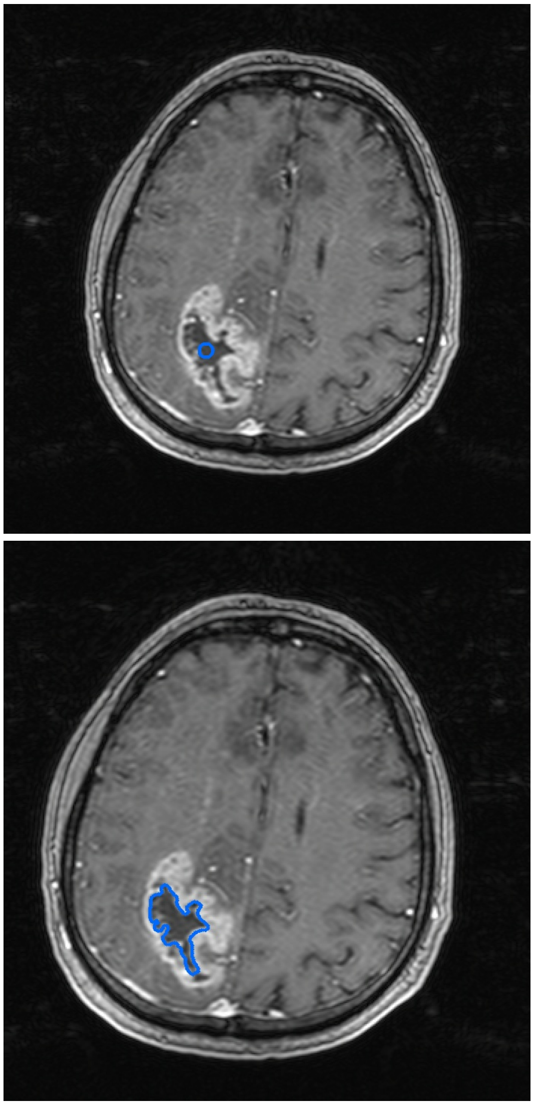

Fig. 5. The initial condition for the segmentation of tumor of lobus occipitalis lateris dextri telencephali (up) and the result of segmentation (down).

In the last experiment we present the segmentation process for the bladder (figure 7), again one can see that topological changes occur. This segmentation took 0.95 seconds ($\tau = 0.2$ and 3500 time steps).

IV. Conclusion

In this paper we presented new approach to medical image segmentation. We developed fast algorithm based on evolving curves in the Lagrangean formulation. Our method allows fast detection of topological changes which is necessary step in order to make the Lagrangean approach efficient. The results were documented on real images where CPU times were always less than 1 second.

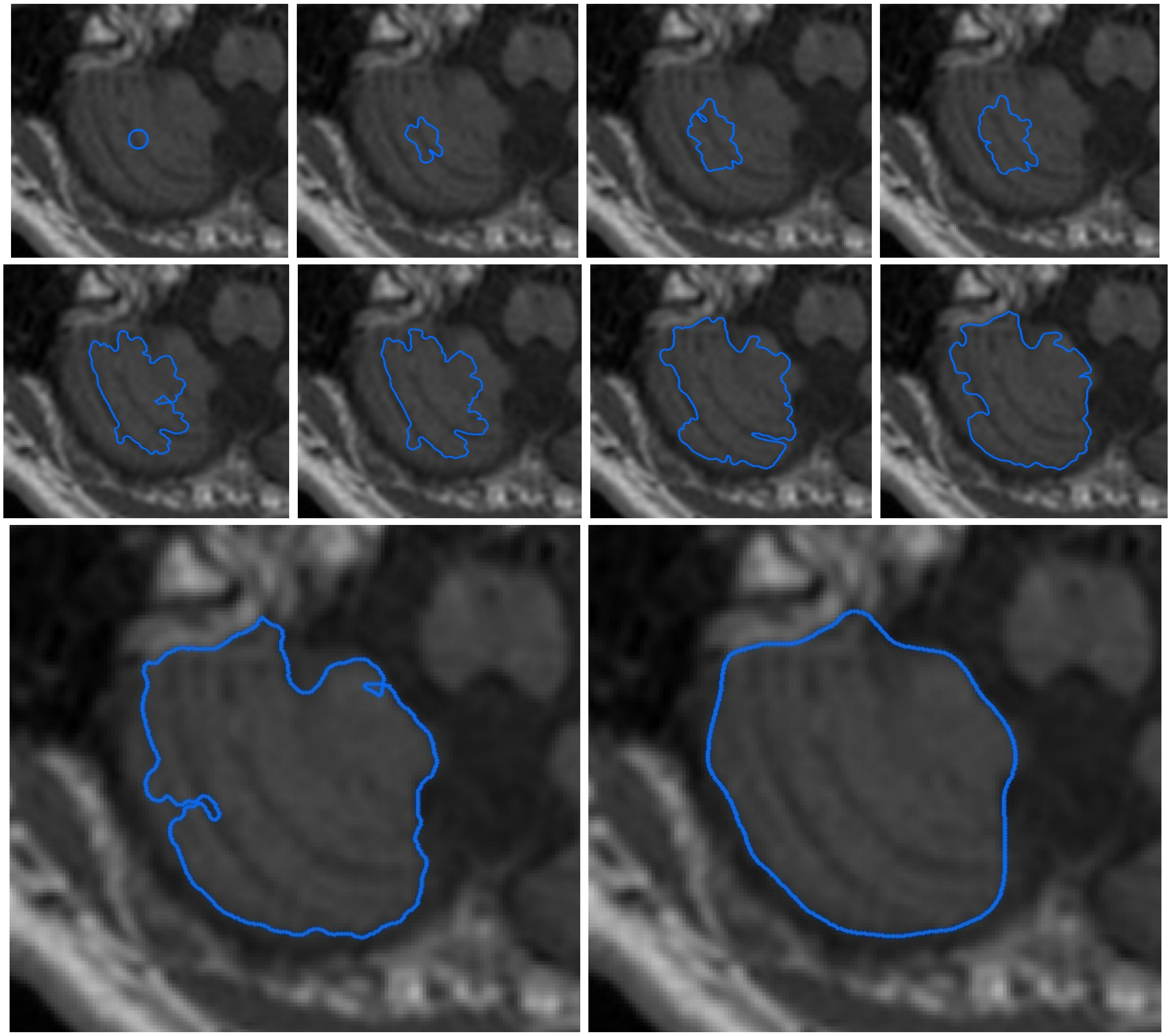

Fig. 4. The initial condition and evolved curve during segmentation of cerebellum.

Acknowledgment

This work was supported by the projects APVV-0184-10 and VEGA 1/0269/09.

References

[1] V. Caselles, R. Kimmel, G. Sapiro, *Geodesic active contours*, International Journal of Computer Vision 22 (1997), 61–79.

[2] T.Y. Hou, J. Lowengrub, M. Shelley, *Removing the stiffness from interfacial flows and surface tension*, J. Comput. Phys., 114 (1994), pp. 312–338.

[3] S. Kichenassamy, A. Kumar, P. Olver, A. Tannenbaum, A. Yezzi, *Conformal curvature flows: from phase transitions to active vision*, Arch. Rational Mech. Anal. 134 (1996), 275–301.

[4] M. Kimura, *Numerical analysis for moving boundary problems using the boundary tracking method*, Japan J. Indust. Appl. Math., 14 (1997), pp. 373–398.

[5] K.Mikula, M.Ohlberger, *Inflow-Implicit/Outflow-Explicit Scheme for Solving Advection Equations*, in Finite Volumes in Complex Applications VI, Problems and Perspectives, Eds. J.Fořt et al. (Proceedings of the Sixth International Conference on Finite Volumes in Complex Applications, Prague, June 6-10, 2011), Springer Verlag, 2011, pp. 683-692

[6] K.Mikula, M.Ohlberger, J.Urbán, *Inflow-Implicit/Outflow-Explicit Finite Volume Methods for Solving Advection Equations*, Technical Report 01/12 - N , FB 10 , Universitaet Muenster, Number 01/12 - N - February 2012

[7] K. Mikula, D. Ševčovič, *Evolution of plane curves driven by a nonlinear function of curvature and anisotropy*, SIAM J. Appl. Math., 61 (2001), pp. 1473–1501.

[8] K. Mikula, D. Ševčovič, *A direct method for solving an anisotropic mean curvature flow of planar curve with an external force*, Mathematical Methods in Applied Sciences, 27(13) (2004) pp. 1545-1565.

[9] K. Mikula, D. Ševčovič, M.Balažovjech, *A simple, fast and stabilized flowing finite volume method for solving general curve evolution equations*, Communications in Computational Physics, Vol. 7, No. 1 (2010) pp. 195-211

[10] K. Mikula, J. Urbán, *3D curve evolution algorithm with tangential redistribution for a fully automatic finding of an ideal camera path in virtual colonoscopy*, Lecture Notes in Computer Science 6667, Springer, 2011

[11] A. Sarti, R. Malladi, J. A. Sethian: *Subjective Surfaces: A Method for Completing Missing Boundaries*, Proceedings of the National Academy of Sciences of the United States of America 12 (97) (2000), 6258–6263.

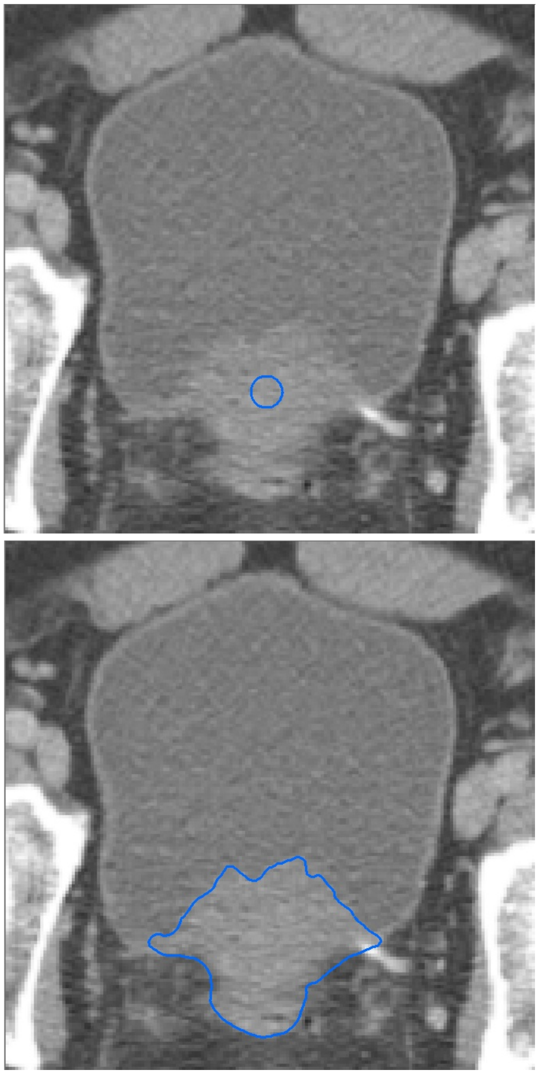

Fig. 6. Segmentation of the prostate.

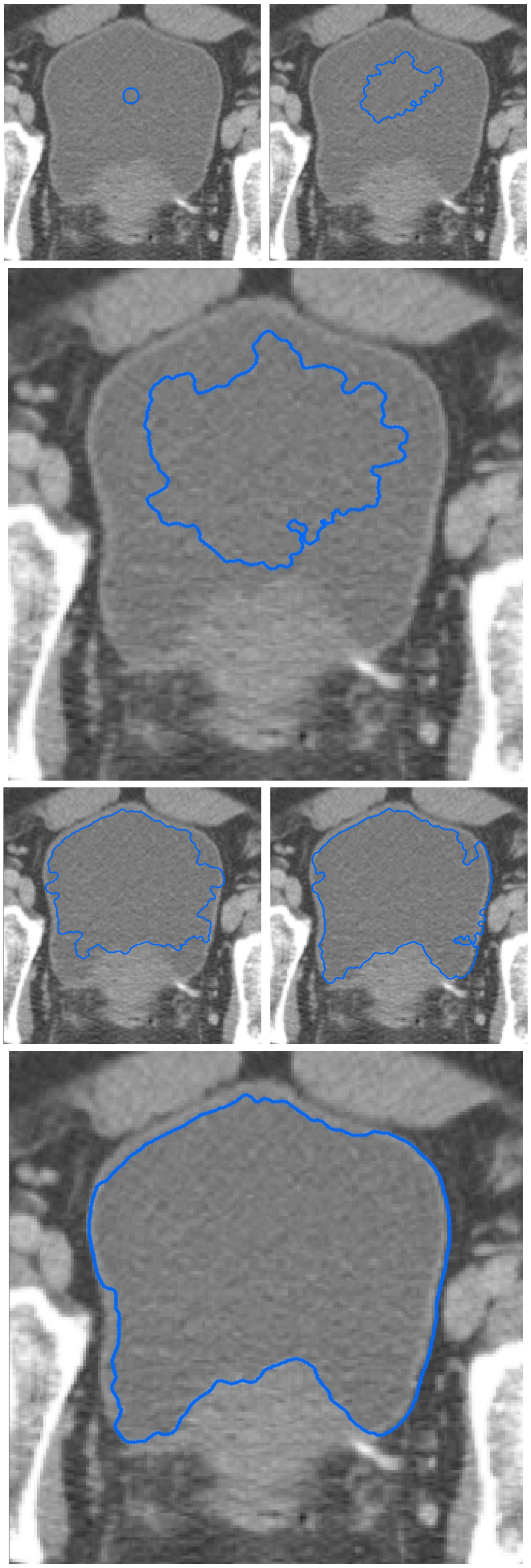

Fig. 7. Segmentation of the bladder.