4D numerical schemes for cell image segmentation and tracking

K. Mikula, N. Peyriéras, M. Remešíková, M. Smíšek

Abstract The paper introduces techniques for space-time segmentation and tracking in time sequences of 3D images of zebrafish embryogenesis. Instead of treating each 3D image individually, we consider the whole time sequence as a single 4D image and perform the extraction of objects and cell tracking in four dimensions. The segmentation of the spatiotemporal objects corresponding to the time evolution of the individual cells is realized by using the generalized subjective surface model that is discretized by a 4D finite volume scheme. Afterwards, we use the distance functions to the borders of the segmented spatiotemporal objects and to the initial cell center positions in order to backtrack the cell trajectories that can be understood as 4D parametrized curves. The distance functions are obtained by numerical solution of the time relaxed eikonal equation.

1 Introduction

Cell tracking, i.e. finding the space-time trajectories and moments of divisions of the cells of a developing organism, is one of the most interesting challenges of modern biology. A reliable backward tracking can answer a lot of questions concerning the origin and formation of cell structures and organs, the global and local movement of the cells, the cell division rate and localization etc. They all are fundamental questions of developmental biology.

In this paper, we introduce the basic concepts of a technique that can be used for the cell tracking from time sequences of 3D images of embryogenesis. A cell

Karol Mikula, Mariana Remešíková, Michal Smíšek
Slovak University of Technology, Radlinského 11, 81368 Bratislava, Slovakia e-mail: mikula@math.sk, remesikova@math.sk, michal.smisek@gmail.com

Nadine Peyriéras
CNRS-DEPSN, Avenue de la Terasse, 91198, Gif sur Yvette, France e-mail: nadine.peyrieras@inaf.cnrs-gif.fr

can be represented by the surface of its nucleus or by its membrane, depending on the type of images we have at disposal. The time evolution of a cell can be seen as spatiotemporal tube whose cross-section by a chosen time hyperplane corresponds to the 3D representation of the cell at the selected time. This 4D tube is bifurcated in the time moments when the cell undergoes division. Thus, we get a tree-like object corresponding to any cell present at the beginning of the time sequence. In order to track a cell, we need to descend from its current position to the root of the tree in which it is situated. This implies that the tracking procedure consists in solving the following two problems:

1. Segmentation of the 4D cell evolution trees from the spatiotemporal image.
2. Finding the way to the root of a tree from any of its inner points.

In our paper, we discuss the solution of both of these problems. We test our methods on artificial data and on time sequences of 3D images corresponding to the zebrafish embryonic development obtained by a confocal microscope. In order to be able to apply the described methods, we need to have at disposal the approximate positions of cell centers for all cells visible in the images. For the artificial data, these points are known by construction and for the zebrafish images, the approximate cell centers are computed by a level set object detection technique [3].

In order to solve the first problem, we apply the generalized subjective surface model [1, 7]

$$u_t - w_a \nabla g \cdot \nabla u - w_c g |\nabla u| \nabla . \left(\frac{\nabla u}{|\nabla u|} \right) = 0 \,, \tag{1}$$

solved in the domain $[0, T_S] \times \Omega$ where $\Omega \subset R^4$ is the spatiotemporal image domain, i.e. the whole time sequence of 3D images. We set $u(0,x) = u_0(x)$ and we consider the zero Dirichlet boundary condition on $\partial\Omega$. The edge detector function $g = g(|\nabla G_\sigma * I_0|)$, I_0 being the 4D image intensity function, and w_a and w_c are the advection and curvature parameters of the model. The desired cell evolution tree segmentation is represented by a selected isosurface of the function $u(x, T_S)$. We would like to point out the importance of performing this segmentation in 4D. Although the cell evolution tree object could be more easily composed of less time and memory consuming 3D cell segmentations, this could lead to spurious interruptions of the cell trajectories in the points where the cell center and consequently the corresponding cell segmentation is missing for some reason. Looking for a whole spatiotemporal structure rather than a composition of 3D objects makes the procedure more robust and resistant to the possible errors of the center detection technique.

Having segmented the tree object, we now want to find a way down to its root from any of its inner points. Since the root can be represented by the center of the root cell, a reasonable descend direction indicator could be the gradient of the distance function d_1 to this center computed inside the segmented 4D object. However, this might not be sufficient. In real data containing a large number of cells, we can observe that the trees corresponding to different root cells are not always perfectly isolated. In order to prevent dropping into a wrong tree, it is desirable to descend along the center line of the tree branches. For this purpose, we compute the distance function to the border of the 4D tree, denoted by d_2, whose negative gradient leads us towards the center line that we want to follow. The distance function to a set Ω_0 can be computed by solving the time relaxed eikonal equation

$$d_t + |\nabla d| = 1 \tag{2}$$

in the domain $[0,T_D] \times \Omega_D$. In our case, Ω_D is the inner part of the segmented tree object, i.e. the part where $u(x,T_S) > V$, V being the isosurface value chosen to represent the segmentation result. The equation (2) has to be coupled with a Dirichlet type condition

$$d(x,t) = 0, \quad x \in \Omega_0 \tag{3}$$

where Ω_0 can represent the root point of the tree or its boundary, i.e. the set of points where $u(x,T_S) = V$.

The descend to the root of the tree is performed as follows. Given an arbitrary point (doxel center) $[x_1,x_2,x_3,x_4]$ inside the tree, we move to the center of the nearest doxel in the direction given by ∇d_1. Supposing that x_4 represents the time dimension of the 4D data, we repeat this step until we drop to the level $x_4 - 1$. After, we move in the direction of $-\nabla d_2$ until we find the nearest ridge point of d_2. Thus we are situated on the center line of the current branch of the tree. From there we repeat the whole procedure until we descend to the level $x_4 = 0$, resp. to the root of the tree.

2 Discretization of the models

The time discretization of the generalized subjective surface model (1) is semi-implicit since in this way we can guarantee unconditional stability of the curvature term. Let τ_S be the time discretization step, $\tau_S = T_S/N_S$. Then for any $n = 1 \ldots N_S$ we get

$$\frac{u^n - u^{n-1}}{\tau_S} - w_a \nabla g \cdot \nabla u^{n-1} - w_c\, g |\nabla u^{n-1}| \nabla \cdot \frac{\nabla u^n}{|\nabla u^{n-1}|} = 0\,. \tag{4}$$

where u^n represents the numerical solution on the nth time level.

The space discretization is realized by applying the finite volume strategy where one doxel of the 4D image corresponds to one volume of the discretization. Let us suppose that the volumes are 4D cubes of side length h and let $V_{\mathbf{i}}$ denote the volume with index vector $\mathbf{i} = (i,j,k,l)$ and $u^n_{\mathbf{i}}$ the value of the numerical solution u^n in the center $c_{\mathbf{i}}$ of this volume. Further, let $\mathbf{e}_p$, $p = 1 \ldots 4$ represent the standard basis vectors in R^4, $F^{+p}_{\mathbf{i}}$ and $F^{-p}_{\mathbf{i}}$ the two faces of $V_{\mathbf{i}}$ orthogonal to $\mathbf{e}_p$, $\nu^{\pm p}_{\mathbf{i}}$ the normal of the face $F^{\pm p}_{\mathbf{i}}$ and $m(F^{\pm p}_{\mathbf{i}})$ its measure.

Now let us integrate (4) over $V_{\mathbf{i}}$. We get

$$\int\limits_{V_{\mathbf{i}}} \frac{u^n - u^{n-1}}{\tau_S} dx - \int\limits_{V_{\mathbf{i}}} w_a \nabla g \cdot \nabla u^{n-1} dx - \int\limits_{V_{\mathbf{i}}} w_c\, g |\nabla u^{n-1}| \nabla \cdot \frac{\nabla u^n}{|\nabla u^{n-1}|} dx = 0. \tag{5}$$

The time derivative term is approximated by

$$\int_{V_{\mathbf{i}}} \frac{u^n - u^{n-1}}{\tau_S}\,dx \approx m(V_{\mathbf{i}})\frac{u_{\mathbf{i}}^n - u_{\mathbf{i}}^{n-1}}{\tau_S}\,. \tag{6}$$

The advection term is approximated by the upwind approach, i.e.

$$\int_{V_{\mathbf{i}}} (-w_a \nabla g \cdot \nabla u)\,dx \approx \tag{7}$$

$$w_a m(V_{\mathbf{i}}) \sum_{p=1}^{4} \left(\max\left(-D_{\mathbf{i}}^p g, 0\right) \frac{u_{\mathbf{i}}^{n-1} - u_{\mathbf{i}-\mathbf{e}_p}^{n-1}}{h} + \min\left(-D_{\mathbf{i}}^p g, 0\right) \frac{u_{\mathbf{i}+\mathbf{e}_p}^{n-1} - u_{\mathbf{i}}^{n-1}}{h} \right)$$

where $D_{\mathbf{i}}^p g = (g_{\mathbf{i}+\mathbf{e}_p} - g_{\mathbf{i}-\mathbf{e}_p})/(2h)$ and $g_{\mathbf{i}}$ is the average value of g in $V_{\mathbf{i}}$. For the curvature term we get the approximation

$$\int_{V_{\mathbf{i}}} w_c g |\nabla u^{n-1}| \nabla . \frac{\nabla u^n}{|\nabla u^{n-1}|} dx = w_c g_{\mathbf{i}} \bar{Q}_{\mathbf{i}}^{n-1} \sum_{p=1}^{4} \sum_{q=-p,+p} \int_{F_{\mathbf{i}}^q} \frac{\nabla u^n}{|\nabla u^{n-1}|} . \nu_{\mathbf{i}}^q \, d\gamma, \tag{8}$$

where $\bar{Q}_{\mathbf{i}}^{n-1}$ is the average value of $|\nabla u^{n-1}|$ in $V_{\mathbf{i}}$. Further

$$\int_{F_{\mathbf{i}}^{\pm p}} \frac{\nabla u^n}{|\nabla u^{n-1}|} . \nu_{\mathbf{i}}^{\pm p} \, d\gamma \approx \frac{m(F_{\mathbf{i}}^{\pm p})}{Q_{\mathbf{i}}^{\pm p;n-1}} \frac{u_{\mathbf{i}\pm\mathbf{e}_p}^n - u_{\mathbf{i}}^n}{h} \tag{9}$$

where $Q_{\mathbf{i}}^{\pm p;n-1}$ is the average value of $|\nabla u^{n-1}|$ on the face $F_{\mathbf{i}}^{\pm p}$.

As we can see, in order to properly perform the approximations indicated in (7) and (8), we need to find an appropriate approximation of the average value of $|\nabla u^{n-1}|$ in both $V_{\mathbf{i}}$ and on the faces $F^{\pm p}$ and the average modulus of $g(|\nabla I_\sigma|)$, $I_\sigma = G_\sigma * I_0$, in $V_{\mathbf{i}}$. There are various possibilities how to do that [5].

Let us first consider the approximation of ∇u^{n-1} in the barycenter $c_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}_p}$ of the doxel face $F_{\mathbf{i}}^{\pm p}$. The component corresponding to the direction of $\mathbf{e}_p$ is simply approximated by

$$D^{\pm p} u_{\mathbf{i}}^{n-1} = \pm \frac{u_{\mathbf{i}\pm\mathbf{e}_p}^{n-1} - u_{\mathbf{i}}^{n-1}}{h}\,. \tag{10}$$

The other components corresponding to the directions of $\mathbf{e}_q$, $q = 1 \ldots 4$, $q \neq p$, can be approximated as follows. The doxel face $F_{\mathbf{i}}^{\pm p}$ is a 3D cube with faces denoted by $F_{\mathbf{i}}^{\pm p,\pm q}$. The barycenter of $F_{\mathbf{i}}^{\pm p,\pm q}$ can be expressed as $c_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}_p\pm\frac{1}{2}\mathbf{e}_q}$. Thus, the value of u^{n-1} at this point can be approximated as

$$u_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}_p\pm\frac{1}{2}\mathbf{e}_q}^{n-1} = \frac{1}{4}(u_{\mathbf{i}}^{n-1} + u_{\mathbf{i}\pm\mathbf{e}_p}^{n-1} + u_{\mathbf{i}\pm\mathbf{e}_q}^{n-1} + u_{\mathbf{i}\pm\mathbf{e}_p\pm\mathbf{e}_q}^{n-1})\,. \tag{11}$$

The partial derivatives of u^{n-1} at $c_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}_p}$ are then approximated as

$$D^{\pm p,q} u_{\mathbf{i}}^{n-1} = \frac{u^{n-1}_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}_p+\frac{1}{2}\mathbf{e}_q} - u^{n-1}_{\mathbf{i}\pm\frac{1}{2}\mathbf{e}_p-\frac{1}{2}\mathbf{e}_q}}{h} \tag{12}$$

Finally, we can define the required approximations

$$Q_{\mathbf{i}}^{\pm p;n-1} = \sqrt{(D^{\pm p}u_{\mathbf{i}}^{n-1})^2 + \sum_{q\neq p}\left(D^{\pm p,q}u_{\mathbf{i}}^{n-1}\right)^2}, \; \bar{Q}_{\mathbf{i}}^{n-1} = \frac{1}{8}\sum_{p=1}^{4}\left(Q_{\mathbf{i}}^{-p;n-1} + Q_{\mathbf{i}}^{+p;n-1}\right)$$

$$G_{\mathbf{i}}^{\pm p} = \sqrt{(D^{\pm p}I_{\sigma;\mathbf{i}})^2 + \sum_{q\neq p}(D^{\pm p,q}I_{\sigma;\mathbf{i}})^2}, \qquad g_{\mathbf{i}} = \frac{1}{8}\sum_{p=1}^{4}\left(G_{\mathbf{i}}^{-p} + G_{\mathbf{i}}^{+p}\right) \tag{13}$$

Combining (6)–(13), we get the finite volume scheme for solving the problem (1).

The eikonal equation (2) is discretized by the Rouy-Tourin scheme [6]. Let $\tau_D = T_D/N_D$ be the time discretization step and $d_{\mathbf{i}}^n$ the value of the numerical solution in the barycenter of the doxel $V_{\mathbf{i}}$ on the nth time level. Let us define for $p = 1\ldots 4$

$$D_{\mathbf{i}}^{\pm p} = \left(\min\left(d_{\mathbf{i}\pm\mathbf{e}_p}^{n-1} - d_{\mathbf{i}}^{n-1}\right)\right)^2, \quad M_{\mathbf{i}}^{p} = \max\left(D_{\mathbf{i}}^{-p}, D_{\mathbf{i}}^{+p}\right)$$

Then the numerical scheme is written as follows

$$d_{\mathbf{i}}^{n} = d_{\mathbf{i}}^{n-1} + \tau_D - \frac{\tau_D}{h}\sqrt{\sum_{p=1}^{4} M_{\mathbf{i}}^{p}} \tag{14}$$

This scheme is stable for $\tau_D \leq h/4$ and it produces monotonically increasing updates that gradually approach a steady state. This leads to an efficient implementation of the scheme that uses a fixing strategy [2].

3 Experiments

Before we proceed to the experiments concerning the actual segmentation and tracking, we test the experimental order of convergence of the finite volume scheme presented above on a simple regularized mean curvature flow equation

$$\partial_t u = |\nabla u|\,\nabla\cdot\left(\frac{\nabla u}{|\nabla u|}\right) \tag{15}$$

with the exact solution

$$u(x_1,x_2,x_3,x_4,t) = \frac{x_1^2+x_2^2+x_3^2+x_4^2-1}{6} + t.$$

We use the Dirichlet boundary condition and the initial condition given by this analytical solution. The problem was solved in the domain $[-1.25, 1.25]^4 \times [0, 0.08]$. The spatial domain consisted of n^4 doxels with $h = 2.5/n$. The time step τ was pro-

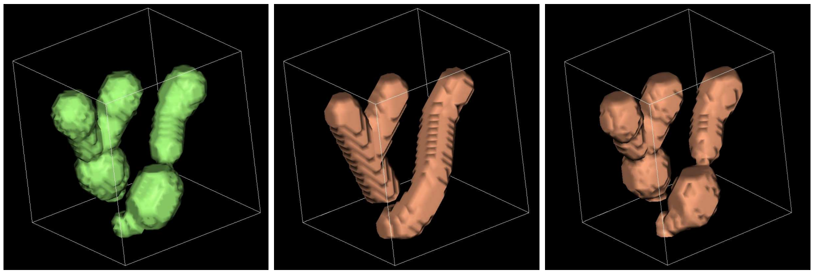

Fig. 1 Segmentation of artificial 4D data. Left, the isosurface $V = 128$ of the 3D representation of the data. Middle, the isosurface $V = 15$ of the 3D representation of the initial segmentation function. Right, the isosurface $V = 15$ of the 3D representation of the segmentation result. This isosurface was chosen as the best representation of the segmented object.

portional to h^2. The error of the numerical solution was measured in the $L_\infty(I, L_2(\Omega))$ norm. The result of this test is displayed in Tab. 1.

Table 1 The experimental order of convergence of the finite volume scheme described in Sec. 2

n	τ	error	EOC
10	0.04	5.531426e-3	
20	0.01	7.276024e-4	2.926
40	0.0025	1.407815e-4	2.370
80	0.000625	3.264185e-5	2.109

The second experiment illustrates the segmentation of artificial 4D data. The 4D image was constructed as an analogy of the cell nuclei movement and division. The cell nuclei are more or less spherical objects, so we started with two spheres. In each time slice of the 4D image, these two spheres are situated at different positions but not far from their positions in the previous time slice. We construct 25 time slices. At time $x_4 = 9$, one of the spheres divides and from then on, we have 3 spheres in the image. To make the situation more general, the radii of the spheres are not constant in time. The centers of these spheres are used to construct the initial segmentation function for the GSUBSURF segmentation. We place a 4D ellipsoid with radii a, b, c, d in each of these centers and we set $u_0(x) = 1$ inside these ellipsoids and $u_0(x) = 0$ outside. The model parameters were set as follows: $K = 1.0$, $h = 1.0$, $\tau_S = 0.1$, $w_a = 5.0$, $w_c = 0.1$, $T_D = 30$. Instead of $|\nabla u|$ we use its regularization $\sqrt{\varepsilon + |\nabla u|^2}$ with $\varepsilon = 10^{-6}$. The procedure is illustrated in Fig. 1. In order to visualize a 4D discrete function $u(x_1, x_2, x_3, x_4)$ with m slices in x_4-coordinate, we construct its 3D representation by setting the value in each 3D voxel (x_1, x_2, x_3) to $\max\limits_{x_4=1\ldots m} u(x_1, x_2, x_3, x_4)$. Then we visualize an isosurface of this representation.

Another experiment shows the segmentation of the zebrafish embryogenesis data. We segmented a sequence of 20 3D cell nuclei images preprocessed (denoised) by

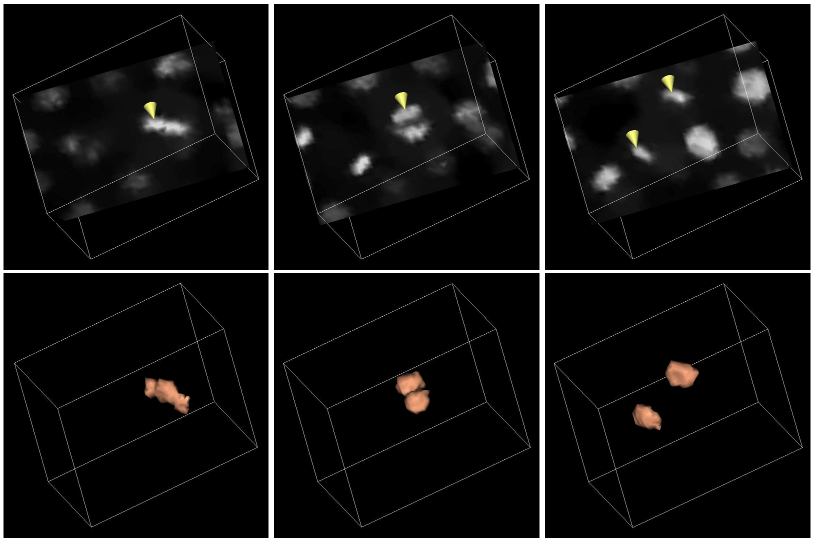

Fig. 2 Segmentation of the zebrafish embryogenesis data. On the top, we display 2D slices of the 4D image corresponding to different x_4 (time) values with indication of the position of the segmented object. On the bottom, we provide the corresponding segmentation result in the form of isosurface $V = 128$ of x_4-slices of the 4D segmentation function.

the geodesic mean curvature flow filter [4]. The initial segmentation function was constructed in the same way as in the case of the artificial data. Further, we set $K = 100.0$, $h = 1.0$, $\tau_S = 0.1$, $w_a = 10.0$, $w_c = 1.0$, $T_D = 50$, $\varepsilon = 10^{-6}$. Fig. 2 displays 2D slices of the 4D data (more precisely, 2D slices of x_4-slices of the 4D data). The object that we tried to segment was a simple cell evolution tree containing one cell division. Together with the image slices, we provide the segmentation result, now displayed as isosurfaces of x_4 (time) slices of the segmentation function.

Fig. 3 shows the result of the cell tracking performed on the artificial data described above. We backtrack the cells (spheres) from the positions of their centers at the end of the time sequence. Both distance functions d_1 and d_2 were computed by setting $h = 1.0$, $\tau_D = 0.25$. The result of the tracking is a set of 4D points characterizing the cell position on the individual time levels. At each time level, we get one point that represents the intersection of the time hyperplane with the ridge of the 4D distance function d_2 (note that these points in general do not correspond to the geometrical centers of the individual 3D spheres). The points are visualized by neglecting their x_4 coordinate.

Finally, we present a test illustrating the effect of using the distance function d_2. In Fig. 4, we can see four branches of 2D cell evolution trees. As we can observe, if using only the distance function d_1, the tracking lines tend to go along the borders of their respective branches or, if the branches are not completely isolated, they can drop into a wrong tree. Taking in account the function d_2, we are able to proceed along the center lines of the branches and thus avoid this unwanted effect.

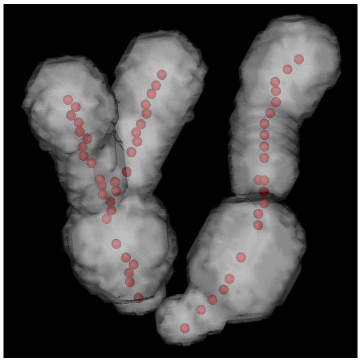

Fig. 3 The result of the cell tracking performed on artificial 4D data. We can see the points characterizing the positions of the cells at each time level visualized by neglecting their x_4 coordinate (as this is a projection from 4D to 3D, in some cases we have actually more than one 4D point projected to one 3D point). The starting points for the tracking are situated on the top of the point sequences.

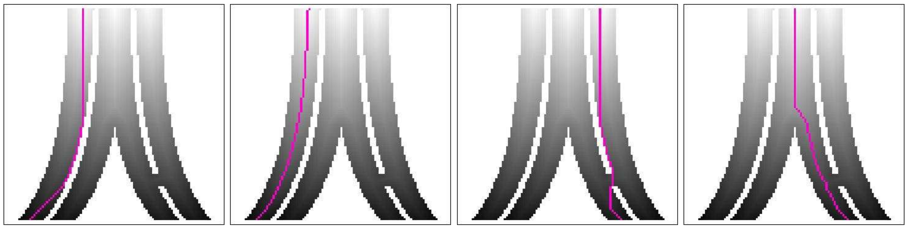

Fig. 4 The effect of using the distance function d_2. From the left: first, the tracking line in an isolated branch obtained by using only d_1, second, the tracking line in the same branch when using d_1 and d_2, third, the tracking line in a branch interconnected with a neighboring branch drops into a wrong branch if only d_1 and not d_2 is applied, fourth, by applying both d_1 and d_2, the line remains in the correct branch. The grey level shading of the branches represents the values of d_1.

References

1. Bourgine, P., Čunderlík, R., Drblíková-Stašová, O., Mikula, O. , Peyriéras, N., Remešíková, M., Rizzi, M., Sarti, A.: 4D embryogenesis image analysis using PDE methods of image processing. Kybernetika **46 (2)**, 226–259 (2010).
2. Bourgine, P., Frolkovič, P., Mikula, K., Peyriéras, N., Remešíková, M.: Extraction of the intercellular skeleton from 2D microscope images of early embryogenesis. In Lecture Notes in Computer Science **5567** (Proceeding of the 2nd International Conference on Scale Space and Variational Methods in Computer Vision, Voss, Norway, June 2009) (Springer, 2009), p. 38–49.
3. Frolkovič, P., Mikula, K., Peyriéras, N., Sarti, A.: A counting number of cells and cell segmentation using advection-diffusion equations. Kybernetika **43 (6)**, 817–829 (2007).
4. Krivá, Z., Mikula, K., Peyriéras, N., Rizzi, B., Sarti, A., Stašová, O.: Zebrafish early embryogenesis 3D image filtering by nonlinear partial differential equations. Medical Image Analysis, **14 (4)**, 510–526 (2010).
5. Mikula, K., Remešíková, M.: Finite volume schemes for the generalized subjective surface equation in image segmentation. Kybernetika **45 (4)**, 646–656 (2009).
6. Rouy, E., Tourin, A.: Viscosity solutions approach to shape-from-shading. SIAM Journal on Numerical Analysis **29 (3)**, 867–884 (1992).
7. Zanella, C., Campana, M., Rizzi, B., Melani, C., Sanguinetti, G., Bourgine, P., Mikula, K., Peyriéras, N., Sarti, A.: Cells Segmentation from 3-D Confocal Images Of Early Zebrafish Embryogenesis. IEEE Transactions on Image Processing **19 (2)**, (2010).