KYBERNETIKA — VOLUME *46* (2010), NUMBER 2, PAGES 226–259

4D EMBRYOGENESIS IMAGE ANALYSIS USING PDE METHODS OF IMAGE PROCESSING

Paul Bourgine, Róbert Čunderlík, Olga Drblíková-Stašová, Karol Mikula, Nadine Peyriéras, Mariana Remešíková, Barbara Rizzi, and Alessandro Sarti

In this paper, we introduce a set of methods for processing and analyzing long time series of 3D images representing embryo evolution. The images are obtained by *in vivo* scanning using a confocal microscope where one of the channels represents the cell nuclei and the other one the cell membranes. Our image processing chain consists of three steps: image filtering, object counting (center detection) and segmentation. The corresponding methods are based on numerical solution of nonlinear PDEs, namely the geodesic mean curvature flow model, flux-based level set center detection and generalized subjective surface equation. All three models have a similar character and therefore can be solved using a common approach. We explain in details our semi-implicit time discretization and finite volume space discretization. This part is concluded by a short description of parallelization of the algorithms. In the part devoted to experiments, we provide the experimental order of convergence of the numerical scheme, the validation of the methods and numerous experiments with the data representing an early developmental stage of a zebrafish embryo.

Keywords: image processing, embryogenesis, image analysis, finite volume method, image filtering, object counting, segmentation, partial differential equation

Classification: 35A99, 74S10, 68U10

1. INTRODUCTION

The modern microscopy technique allows *in vivo* imaging of organisms at cell level and at very early stages of development, even as soon as one cell or few cells stage, without corrupting the cell integrity and normal development of the embryo. The speed of the scanning is also improving and therefore it is possible to obtain images of long periods of the embryonic development with a relatively short time step. Having such data, we can perform miscellaneous analyses of the embryogenesis and compare the development of different individuals. Some of the interesting challenges of nowadays are for example reconstructing the cell lineage tree – tracking the cells of an embryo from one cell stage to the fully developed organism, quantitative analysis of the embryo – determining the number of cells, cell density, density of cell divisions etc., finding some general characteristics of the embryogenesis process or measuring the differences between individuals developing in different conditions, e. g. after

application of some drugs.

In order to reach such goals, we need, besides sufficiently good quality data, precise and efficient image processing algorithms. For all the embryo analyses mentioned above, it is necessary to process a large number of 3D data, while each of the images can contain thousands of cells. The algorithms must be able to deal with the imperfections of the images that are intrinsically linked to the scanning technique – noise, presence of artifacts, incomplete or unclear structures. As the results of some algorithms are not used only for the analyses themselves but also as inputs for other image processing methods, it is important that each algorithm is as reliable as possible. All this puts high demands on all involved methods.

In this paper, we present an efficient strategy for analysis of time series of 3D embryogenesis images. Our technique consists of a set of image processing algorithms that are based on PDE models. We present a unified approach to all parts of the image processing chain – the mathematical models for all individual processes have similar features and the corresponding numerical solutions are based on the same principle. All the methods can be naturally parallelized using a common approach as well.

The first step of the technique is the image filtering. This is an essential step because some noise is always present in any microscope image and its level increases with the speed of image acquisition that is a key issue in the cell lineage tree reconstruction. We perform this step by numerical solution of the so-called geodesic mean curvature flow model (GMCF) [3, 4, 9] which was chosen from several available methods by careful testing. The second step is the flux-based level set center detection method (FBLSCD) used to extract the approximate position and number of cell nuclei centers. FBLSCD is an evolutionary process based on an advection-diffusion formulation of morphological operators. The process consists of numerical solution of the model using the flux-based finite volume discretization [6, 7] and consecutive detection of local maxima of the evolving function. The final step is the image segmentation by the generalized subjective surface method (GSUBSURF) [5, 11, 14]. The results obtained by segmentation can be directly used for various analyses of the embryo and cell tracking. In all mentioned steps, we apply the semi-implicit time discretization and finite volume space discretization.

The paper is organized as follows. First, we describe the data we are dealing with. In the next section, we present the mathematical models of all three processes of the chain. After, we explain our numerical methods used for solving all these models and we briefly discuss the parallel implementation of the algorithms. In the last section of the paper, we will mention several possible applications of our strategy and we will show results of some experiments and embryo analyses. The validation of the individual steps of the procedure is also provided.

2. EMBRYOGENESIS IMAGE DATA

Within the scope of our research, we work with two types of images – images of cell nuclei and cell membranes. The images are obtained by a confocal laser microscope with a color output. However, there is always only one color channel that is important (typically red for nuclei images and green for membrane images), which means

that the greylevel scale (a scalar intensity function) is sufficient for the representation of the data.

Let us denote by $u_N^0 : \Omega \to R$, $\Omega \subset R^3$ the intensity function representing the nuclei image and by $u_M^0 : \Omega \to R$, $\Omega \subset R^3$ the function corresponding to membrane image, Ω being a 3D rectangular image domain. Without loss of generality we can assume that $0 \leq u_N^0 \leq 1$ and $0 \leq u_M^0 \leq 1$.

Our research was mainly focused on the embryonic development of zebrafish (*danio rerio*, see Figures 1, 2). This animal is a popular model organism in developmental biology because its embryos develop rapidly and they are large, resistant and transparent. The part of embryogenesis we are interested in covers hundreds to thousands of cells in one time moment which is a reasonable and yet not trivial number to process.

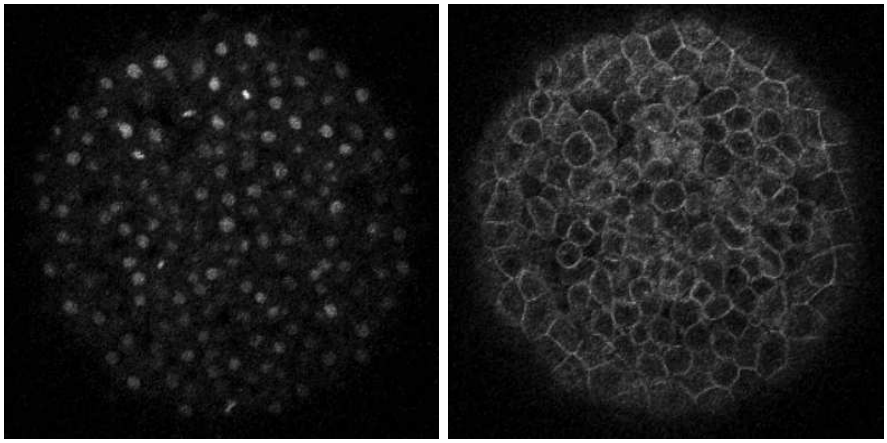

Fig. 1. Zebrafish embryo in a 2D horizontal slice view. On the left, the image of cell nuclei, on the right, the cell membranes.

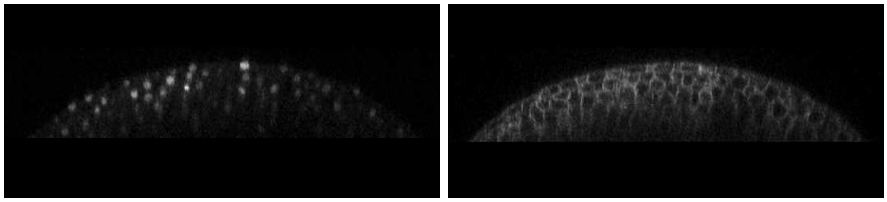

Fig. 2. Zebrafish embryo in a 2D vertical slice view. On the left, the image of cell nuclei, on the right, the cell membranes.

3. MATHEMATICAL MODELS FOR THE IMAGE PROCESSING ALGORITHMS

In this section, we will describe the PDE models involved in our image processing chain.

3.1. Image filtering by the geodesic mean curvature flow

The image filtering is the basic step of the overall procedure and it is necessary to perform it before any further method is applied. The noise is intrinsically linked to the microscopy technique and its presence increases with decreasing time step of the scanning. Optimizing the signal to noise ratio and the speed of imaging always means to accept a certain level of noise. Many image analysis algorithms are generally not able to work directly with noisy images because the presence of spurious noisy structures can lead to incorrect results. The developed methods are usually also computationally much faster when applied to properly filtered image data.

There is a number of methods that could be used to filter the image data sets described in Sec. 2 and therefore some tests had to be performed in order to choose the most appropriate one. In Sec. 6.2.1, we provide a quantitative study of the image filtering properties of several nonlinear diffusion models. The results show that the geodesic mean curvature flow model is the most suitable one for the data we are dealing with. The partial differential equation for the GMCF filtering reads as follows

$$u_t - |\nabla u| \nabla . \left(g\left(|\nabla G_\sigma * u|\right) \frac{\nabla u}{|\nabla u|} \right) = 0\,, \tag{1}$$

where $u(t,x)$, $t > 0$, represents the filtered image intensity function. We start from the initial condition $u(0,x) = u_N^0(x)$ for nuclei images resp. $u(0,x) = u_M^0(x)$ for membrane images and we consider the zero Neumann boundary conditions on the boundary $\partial\Omega$ of the image domain. In this model, the mean curvature motion of the level sets of function u is determined by the edge indicator function

$$g(s) = \frac{1}{1 + K s^2}, \qquad K \geq 0 \tag{2}$$

that is applied to the image gradient presmoothed by the Gaussian kernel with a small variance σ. The essential property of this function is that its negative gradient points towards the edges in the image. The diffusion term in (1) causes accumulation of the level sets of u along the boundaries of objects in the image and therefore this filtering is edge preserving.

The optimally filtered image is obtained as the solution of (1) after a few time steps, at time $t = T_F$. The optimality of the filtering result in a specific time step as well as of the method itself can be verified by taking in account the criterion of the mean Hausdorff distance from a gold standard, as we show in Sec. 6.2.1. Let $u_{N_f}^0$, $u_{M_f}^0$ represent the results of filtering for the nuclei and membrane images, respectively.

3.2. Object counting by the flux-based level set center detection

The second step of our approach is the detection of cell nuclei centers. The method that we suggest was originally designed to solve this particular task but it can be applied to a general problem of counting important objects in an image.

The principle of this process lies in the fact that all visible objects in the image can be seen as humps of relatively higher image intensity (in case of dark objects on bright background we consider the inverted image). Any such hump is represented by certain image intensity level sets. The diameters of these level sets allow us to distinguish between significant objects, e. g. cell nuclei, and spurious structures like noise or inner peaks of intensity, which still remain there after GMCF filtering. For cell nuclei, the diameter d is relatively large, $0 \ll c_1 \leq d \leq c_2$, while the diameter of the level sets corresponding to noise or inner structures is much smaller, $0 < d \ll c_1$. In general, the level sets are closed surfaces and if all their points are moving (advected) at a constant speed in the direction of the inner normal to the level set, the encompassed volume is decreasing and finally the hump disappears. Our object counting method is based on the fact that the level sets with small diameter corresponding to spurious structures disappear quickly, while level sets representing important objects are observable in a much longer time scale.

It is well-known that if the evolution of level sets depends on the local mean curvature then the speed of their shrinking tends to infinity as the diameter of level sets tends to zero. We use this fact for an additional speed up of the above mentioned advective level set motion. So, in our model, the normal velocity V of any level set is given as $V = \delta + \mu k$ where δ and μ are constants (model parameters) and k is the mean curvature. Such process of the level set evolution can then be represented by the following equation

$$u_t + \delta \frac{\nabla u}{|\nabla u|} \cdot \nabla u - \mu |\nabla u| \nabla . \left(\frac{\nabla u}{|\nabla u|} \right) = 0 \tag{3}$$

applied to the initial condition $u^0_{N_f}$, resp. $u^0_{M_f}$ (the result of the filtering algorithm). Again, we consider the zero Neumann boundary condition. Due to the shrinking and smoothing of all (real and spurious) structures in such an evolution process, we observe decreasing of the number of local maxima of the solution u satisfying the equation (3). This decrease is fast in the beginning and later it is stabilized. We stop this process when the slope of the decreasing is below a certain threshold. The approximate cell nuclei centers are represented by the points in which the local maxima of the function u are achieved at the stopping time T_C. The number of the local maxima n_C and their positions s_l, $l = 1, \ldots, n_C$ at the time T_C are the main outputs of FBLSCD algorithm. However, some algorithms for embryo analysis, e. g. cell tracking methods, can take advantage also of the centers detected in earlier time steps. The principle of the center detection process is illustrated in Figure 3.

3.3. Segmentation by the generalized subjective surface method

The result of the center detection can be directly used to provide information about the embryo, e. g. it gives the number of cells at a certain time point or the global

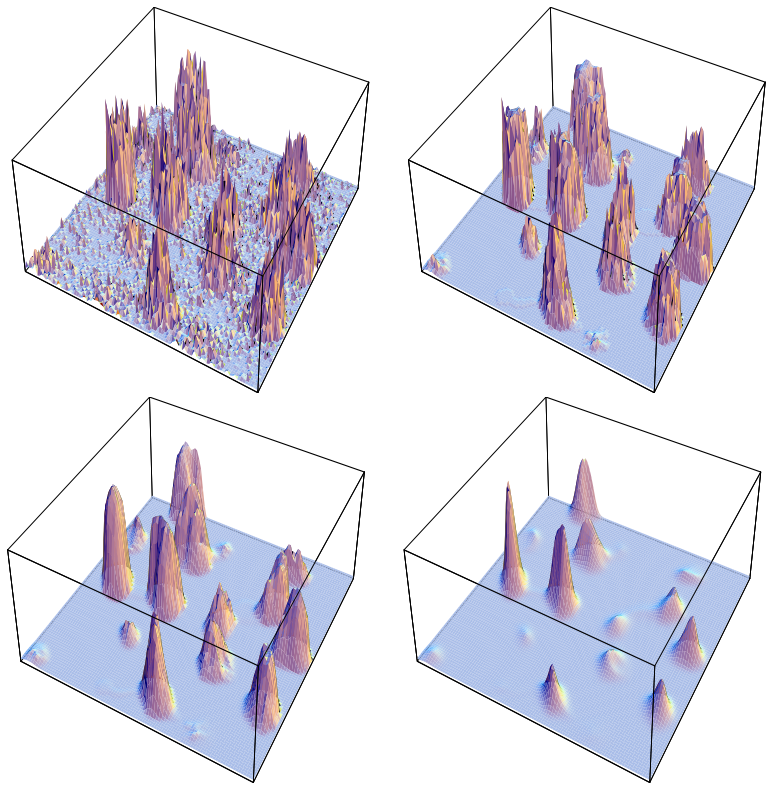

Fig. 3. The center detection process. Top left, the intensity function of the original image. Top right, the intensity function of the filtered image. Bottom left, the intensity after 5 steps of center detection process. Bottom right, the intensity after 26 steps when we stopped the process.

and local densities of cells (supposing that we know the total volume of the embryo). Another important application, and the final step of our procedure, is the cell segmentation and embryo segmentation. By cell segmentation we mean extraction of the shape of either the cell nucleus or the whole cell, depending on which of the two types of images we process. The embryo segmentation is extraction of the shape of the whole organism. The approximate cell centers represent the seeds for any of these segmentation processes.

Let us assume that s_l, $l = 1, \ldots, n_C$ are the points in R^3 where the approximate cell centers were detected by FBLSCD at the stopping time T_C. First, we construct an initial segmentation function for any of these points. The principle of the subjective surface method is that the position of the seed (cell center) is the main factor determining the form of the function. Having constructed the initial function, we

let it evolve by solving the following GSUBSURF equation

$$u_t - w_{\text{con}} \nabla g \cdot \nabla u - w_{\text{dif}} g |\nabla u| \nabla . \left(\frac{\nabla u}{|\nabla u|} \right) = 0 \,, \tag{4}$$

where u is the evolving function, $u(0,x) = u_0(x)$ (the initial segmentation function) and we consider the zero Dirichlet boundary condition on $\partial\Omega$. The function $g = g(|\nabla G_\sigma * u_f^0|)$, where $u_f^0 = u_{N_f}^0$ in case of nuclei segmentation and $u_f^0 = u_{M_f}^0$ in case of membrane segmentation, is of the form (2) or we can use a more general form

$$g(s) = f\left(\frac{1}{1 + Ks^2} \right) , \tag{5}$$

keeping in mind that the function f has to preserve the edge detecting property of the original function stated in (2). If f is suitably chosen, this generalization can speed up the process of segmentation and improve the quality of the results.

The reason why we call the model (4) generalized lies in introducing the parameters w_{con} and w_{dif} in the equation. If we set $w_{\text{con}} = 1.0$ and $w_{\text{dif}} = 1.0$, we can rewrite (4) in the form

$$u_t - |\nabla u| \nabla . \left(g \frac{\nabla u}{|\nabla u|} \right) = 0$$

which is a classical subjective surface formulation stated in [14]. Using the new parameters w_{con} and w_{dif} that can be considered as the weights for the advection and diffusion processes makes the model more flexible. Having the possibility to control separately the two processes, we have the potential to improve the efficiency of the method. Figure 4 illustrates the evolution of the solution for various values of parameters w_{con} and w_{dif}. We can see that the time T_S necessary to segment the object can significantly differ as well as the way the function is evolving. The optimal parameters are found after performing several tests of this type (see row 3 of Figure 4 where the appropriate choice of parameters led to a very fast segmentation).

4. THE DISCRETIZATION OF THE MODELS

In this section, we will describe the numerical methods that are used for solving the equations corresponding to filtering, center detection and segmentation explained in Sec. 3. Though we will show the discretization of all three problems separately, we will emphasize some general ideas that can be applied to any of the models.

4.1. The time discretization

For discretizing any of the equations (1), (3), (4) in time, we use the semi-implicit approach that guarantees unconditional stability of the diffusion terms. Let us suppose that we solve the filtering, center detection and segmentation problems in time intervals $\langle 0, T_F \rangle$, $\langle 0, T_C \rangle$ and $\langle 0, T_S \rangle$, respectively. Let N_F, N_C and N_S be the corresponding number of uniform time steps and $\tau_F = T_F/N_F$, $\tau_C = T_C/N_C$, $\tau_S = T_S/N_S$.

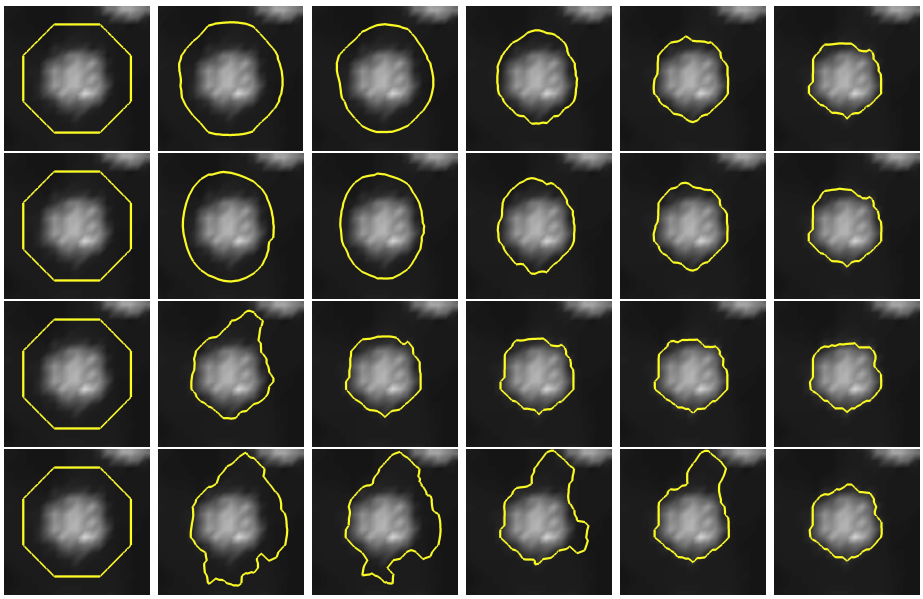

Fig. 4. The influence of convection and diffusion weights on the evolution of the solution of (4). The solution is displayed in times $t = 0$, $t = 1$, $t = 2$, $t = 5$, $t = 10$, $t = 40$. Row 1, $w_{\text{con}} = 1.0$, $w_{\text{dif}} = 1.0$. Row 2, $w_{\text{con}} = 1.0$, $w_{\text{dif}} = 10.0$. Row 3, $w_{\text{con}} = 10.0$, $w_{\text{dif}} = 2.0$. Row 4, $w_{\text{con}} = 10.0$, $w_{\text{dif}} = 0.1$.

The GMCF equation (1) can be approximated in the following way. For any $n = 1 \dots N_F$ we get an equation

$$\frac{u^n - u^{n-1}}{\tau_F} - |\nabla u^{n-1}| \nabla . \left(g \left(|\nabla G_\sigma * u^{n-1}| \right) \frac{\nabla u^n}{|\nabla u^{n-1}|} \right) = 0 \,, \tag{6}$$

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

The discretization of the FBLSCD model (3) reads as follows, for any $n = 1 \dots N_C$

$$\frac{u^n - u^{n-1}}{\tau_C} + \delta \frac{\nabla u^{n-1}}{|\nabla u^{n-1}|} \cdot \nabla u^{n-1} - \mu |\nabla u^{n-1}| \nabla . \left(\frac{\nabla u^n}{|\nabla u^{n-1}|} \right) = 0 \,. \tag{7}$$

For the GSUBSURF equation (4) we get for all $n = 1 \dots N_S$

$$\frac{u^n - u^{n-1}}{\tau_S} - w_{\text{con}} \nabla g \cdot \nabla u^{n-1} - w_{\text{dif}} g |\nabla u^{n-1}| \nabla \cdot \frac{\nabla u^n}{|\nabla u^{n-1}|} = 0 \,. \tag{8}$$

4.2. Finite volume space discretization

In order to discretize (6), (7), (8) in space, we apply the finite volume method. The principle that we explain here is applicable to all models mentioned above.

First, we identify the finite volumes of the mesh $\mathcal{T}_h$ with the voxels of the 3D image which is the most natural choice. We denote each finite volume by V_{ijk}, $i = 1 \dots N_1$, $j = 1 \dots N_2$, $k = 1 \dots N_3$. Such a finite volume grid is regular rectangular and let h_1, h_2, h_3 represent the size of the volumes in x_1, x_2, x_3 direction, respectively. Let $m(V_{ijk})$ denote the volume of V_{ijk} and c_{ijk} its barycenter. By u^n_{ijk} we will denote the approximate value of u^n in c_{ijk}.

For all volumes V_{ijk}, we define two index sets. First, let N_{ijk} denote the set of all (p, q, r) such that $p, q, r \in \{-1, 0, 1\}$, $|p| + |q| + |r| = 1$. Then, let P_{ijk} represent the set of all (p, q, r) satisfying $p, q, r \in \{-1, 0, 1\}$, $|p| + |q| + |r| = 2$. Let us first consider $(p, q, r) \in N_{ijk}$. The line connecting the center of V_{ijk} and the center of its neighbor $V_{i+p,j+q,k+r}$ is denoted by σ^{pqr}_{ijk} and its length $m(\sigma^{pqr}_{ijk})$. The faces of finite volume V_{ijk} are denoted by e^{pqr}_{ijk} with area $m(e^{pqr}_{ijk})$ and normal ν^{pqr}_{ijk}. Let x^{pqr}_{ijk} be the point where the line σ^{pqr}_{ijk} crosses the face e^{pqr}_{ijk}. Finally, for any $(p, q, r) \in P_{ijk}$, let y^{pqr}_{ijk} denote the midpoints of the voxel edges. The approximate value of u^{n-1} at x^{pqr}_{ijk} and y^{pqr}_{ijk}, where (p, q, r) belongs to the corresponding index set, is denoted by u^{pqr}_{ijk}, omitting the time index, as only the values from the time level $n - 1$ will be needed in these points.

The implementation of the boundary conditions is done in the following way. In case of Dirichlet boundary condition, we set $u^n_{ijk} = 0$ for the volumes touching the boundary of the domain and the above definitions are valid only for the inner volumes V_{ijk}. In case of Neumann boundary condition, we use the reflection principle.

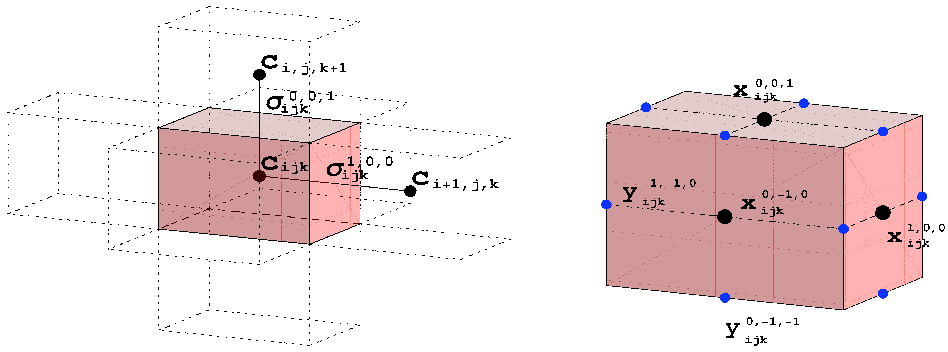

Fig. 5. The finite volume mesh.

Now let us integrate the GMCF equation discretized in time (6) over every finite volume V_{ijk}. We get

$$\int_{V_{ijk}} \frac{u^n - u^{n-1}}{\tau_F} \, \mathrm{d}x - \int_{V_{ijk}} |\nabla u^{n-1}| \nabla . \left(g \left(|\nabla G_\sigma * u^{n-1}| \right) \frac{\nabla u^n}{|\nabla u^{n-1}|} \right) \mathrm{d}x = 0 \,. \qquad (9)$$

The first term on the LHS can be simply approximated as

$$\int\limits_{V_{ijk}} \frac{u^n - u^{n-1}}{\tau_F}\,\mathrm{d}x \approx m(V_{ijk})\frac{u^n_{ijk} - u^{n-1}_{ijk}}{\tau_F}\,.$$

Assuming that the approximation of $|\nabla u^{n-1}|$ in V_{ijk} is a constant, the second term can be rewritten using the divergence theorem

$$\int\limits_{V_{ijk}} |\nabla u^{n-1}|\nabla.\left(g\left(|\nabla G_\sigma * u^{n-1}|\right)\frac{\nabla u^n}{|\nabla u^{n-1}|}\right)\mathrm{d}x$$
$$= \bar{Q}^{n-1}_{ijk}\sum_{N_{ijk}}\int\limits_{e^{pqr}_{ijk}} g\left(|\nabla G_\sigma * u^{n-1}|\right)\frac{\nabla u^n}{|\nabla u^{n-1}|}\nu^{pqr}_{ijk}\,\mathrm{d}\gamma\,, \tag{10}$$

where $\bar{Q}^{n-1}_{ijk}$ is an average modulus of $|\nabla u^{n-1}|$ in V_{ijk}.

A similar procedure can be applied to the FBLSCD problem. In order to obtain a space discretization of (7), we first perform the integration over V_{ijk}

$$\int\limits_{V_{ijk}} \frac{u^n - u^{n-1}}{\tau_C}\,\mathrm{d}x + \int\limits_{V_{ijk}} \delta\frac{\nabla u^{n-1}}{|\nabla u^{n-1}|}\cdot\nabla u^{n-1}\,\mathrm{d}x$$
$$- \int\limits_{V_{ijk}} \mu|\nabla u^{n-1}|\nabla.\left(\frac{\nabla u^n}{|\nabla u^{n-1}|}\right)\mathrm{d}x = 0\,. \tag{11}$$

Again, we can take the approximation

$$\int\limits_{V_{ijk}} \frac{u^n - u^{n-1}}{\tau_C}\,\mathrm{d}x \approx m(V_{ijk})\frac{u^n_{ijk} - u^{n-1}_{ijk}}{\tau_C}\,.$$

Now we will rewrite the second term on the LHS, performing the following procedure. Let us denote

$$v = \delta\frac{\nabla u^{n-1}}{|\nabla u^{n-1}|}\,.$$

As suggested in [6], the convective term can be written in an equivalent form

$$v\cdot\nabla u^{n-1} = \nabla.(vu^{n-1}) - u^{n-1}\nabla\cdot v\,. \tag{12}$$

Therefore, considering u^{n-1} constant in V_{ijk} we get

$$\int\limits_{V_{ijk}} v\cdot\nabla u^{n-1} = \sum_{N_{ijk}}\int\limits_{e^{pqr}_{ijk}} u^{n-1}v\cdot\nu^{pqr}_{ijk}\mathrm{d}\gamma - u^{n-1}_{ijk}\sum_{N_{ijk}}\int\limits_{e^{pqr}_{ijk}} v\cdot\nu^{pqr}_{ijk}\mathrm{d}\gamma\,. \tag{13}$$

The last term on the LHS of (11) is the diffusion term that can be rewritten similarly to (10)

$$\int\limits_{V_{ijk}} \mu|\nabla u^{n-1}|\nabla.\left(\frac{\nabla u^n}{|\nabla u^{n-1}|}\right)\mathrm{d}x = \mu\bar{Q}^{n-1}_{ijk}\sum_{N_{ijk}}\int\limits_{e^{pqr}_{ijk}}\frac{\nabla u^n}{|\nabla u^{n-1}|}\nu^{pqr}_{ijk}\,\mathrm{d}\gamma\,. \tag{14}$$

Finally we come to the space discretization of (8) that can be done following the same principle as in the case of FBLSCD. First, the integration over V_{ijk} reads

$$\int\limits_{V_{ijk}} \frac{u^n - u^{n-1}}{\tau}\,\mathrm{d}x - \int\limits_{V_{ijk}} w_{\mathrm{con}}\nabla g\cdot\nabla u^{n-1}\,\mathrm{d}x - \int\limits_{V_{ijk}} w_{\mathrm{dif}} g|\nabla u^{n-1}|\nabla\cdot\frac{\nabla u^n}{|\nabla u^{n-1}|}\,\mathrm{d}x = 0\,. \tag{15}$$

The approximation of the time derivative term is standard

$$\int\limits_{V_{ijk}} \frac{u^n - u^{n-1}}{\tau_S}\,\mathrm{d}x \approx m(V_{ijk})\frac{u^n_{ijk} - u^{n-1}_{ijk}}{\tau_S}\,.$$

As we are dealing with a convection-diffusion model as in the case of FBLSCD, we use an analogous expression of the two remaining terms on the LHS. This time we define v as

$$v = -w_{\mathrm{con}}\nabla g$$

and afterwards the second term on the LHS is expressed by (13). The diffusion term is rewritten as follows

$$\int\limits_{V_{ijk}} w_{\mathrm{dif}} g|\nabla u^{n-1}|\nabla.\frac{\nabla u^n}{|\nabla u^{n-1}|}\,\mathrm{d}x = w_{\mathrm{dif}} g_{ijk}\bar{Q}^{n-1}_{ijk}\sum_{N_{ijk}}\int\limits_{e^{pqr}_{ijk}}\frac{\nabla u^n}{|\nabla u^{n-1}|}.\nu^{pqr}_{ijk}\,\mathrm{d}\gamma\,, \tag{16}$$

where g_{ijk} is the average modulus of g in V_{ijk}.

Let us note that in all mentioned models, the absolute value of the gradient $|\nabla u|$, resp. $|\nabla u^{n-1}|$ for the equations discretized in time, appears in denominator. In practical implementations, this term is substituted by the regularized term $\sqrt{\varepsilon^2 + |\nabla u|^2}$, resp. $\sqrt{\varepsilon^2 + |\nabla u^{n-1}|^2}$, where ε is the regularization parameter, usually $\varepsilon \ll 1$. Then, instead of $\bar{Q}^{n-1}_{ijk}$ we use $\bar{Q}^{n-1}_{\varepsilon,ijk}$, the average modulus of $\sqrt{\varepsilon^2 + |\nabla u^{n-1}|^2}$.

Summarizing the expressions stated in (10), (13), (14) and (16), we can see that to properly approximate any of them, we need to approximate the average modulus of $|\nabla u^{n-1}|$, resp. $\sqrt{\varepsilon^2 + |\nabla u^{n-1}|^2}$, $g(|\nabla I_\sigma|)$, where $I_\sigma = G_\sigma * u^0_F$, and $g(|\nabla u^{n-1}_\sigma|)$, $u^{n-1}_\sigma = G_\sigma * u^{n-1}$, in both V_{ijk} and on voxel faces e^{pqr}_{ijk}. There are various possibilities how to do that. In [12], three variants based on the same principle were described and tested. Here we mention only one of them, the so called reduced diamond cell approximation, that appeared to be the most suitable for our practical purposes due to its simplicity and overall good performace.

Our scheme is using the values of u^{n-1} in the midpoints y^{pqr}_{ijk} of the voxel edges which are approximated for any $(p,q,r) \in P_{ijk}$ by

$$\begin{aligned}
u^{pq0}_{ijk} &= \frac{1}{4}(u^{n-1}_{ijk} + u^{n-1}_{i+p,j,k} + u^{n-1}_{i,j+q,k} + u^{n-1}_{i+p,j+q,k})\,,\\
u^{p0r}_{ijk} &= \frac{1}{4}(u^{n-1}_{ijk} + u^{n-1}_{i+p,j,k} + u^{n-1}_{i,j,k+r} + u^{n-1}_{i+p,j,k+r})\,,\\
u^{0qr}_{ijk} &= \frac{1}{4}(u^{n-1}_{ijk} + u^{n-1}_{i,j+q,k} + u^{n-1}_{i,j,k+r} + u^{n-1}_{i,j+q,k+r}).
\end{aligned}$$

Let us denote by $\nabla^{pqr} u_{ijk}^{n-1}$ the approximation of the gradient in the barycenter x_{ijk}^{pqr} of the face e_{ijk}^{pqr}, $(p,q,r) \in N_{ijk}$, of the voxel V_{ijk}. Using this notation, we can define

$$\begin{aligned}
\nabla^{p00} u_{ijk}^{n-1} &= (p(u_{i+p,j,k}^{n-1} - u_{ijk}^{n-1})/h_1, (u_{ijk}^{p10} - u_{ijk}^{p,-1,0})/h_2, (u_{ijk}^{p01} - u_{ijk}^{p,0,-1})/h_3)\,, \\
\nabla^{0q0} u_{ijk}^{n-1} &= ((u_{ijk}^{1q0} - u_{ijk}^{-1,q,0})/h_1, q(u_{i,j+q,k}^{n-1} - u_{ijk}^{n-1})/h_2, (u_{ijk}^{0q1} - u_{ijk}^{0,q,-1})/h_3)\,, \\
\nabla^{00r} u_{ijk}^{n-1} &= ((u_{ijk}^{10r} - u_{ijk}^{-1,0,r})/h_1, (u_{ijk}^{01r} - u_{ijk}^{0,-1,r})/h_2, r(u_{i,j,k+r}^{n-1} - u_{ijk}^{n-1})/h_3)\,.
\end{aligned}$$

The required approximations can be now defined as

$$\begin{aligned}
&Q_{ijk}^{pqr;n-1} = |\nabla^{pqr} u_{ijk}^{n-1}|, \quad Q_{\varepsilon,ijk}^{pqr;n-1} = \sqrt{\varepsilon^2 + |\nabla^{pqr} u_{ijk}^{n-1}|^2}\,, \\
&\bar{Q}_{ijk}^{n-1} = \frac{1}{6} \sum_{N_{ijk}} |\nabla^{pqr} u_{ijk}^{n-1}|, \quad \bar{Q}_{\varepsilon,ijk}^{n-1} = \sqrt{\varepsilon^2 + \frac{1}{6} \sum_{N_{ijk}} |\nabla^{pqr} u_{ijk}^{n-1}|^2}\,, \\
&g_{ijk}^{pqr;n-1} = g\left(|\nabla^{pqr} u_{\sigma;ijk}^{n-1}|\right), \quad g_{ijk} = g\left(\frac{1}{6} \sum_{N_{ijk}} |\nabla^{pqr} I_{\sigma;ijk}|\right)\,.
\end{aligned} \tag{17}$$

4.3. The fully discrete formulation of the problems

Having computed all the necessary approximations, we are ready to write the fully discrete formulations of the problems (1), (3), (4).

The discretization of the filtering problem (1) is straightforward. (17) gives us all required approximations and recalling (10), we can write the discrete equation

$$m(V_{ijk}) \frac{u_{ijk}^n - u_{ijk}^{n-1}}{\tau_F} = \bar{Q}_{ijk}^{n-1} \sum_{N_{ijk}} m(e_{ijk}^{pqr}) \frac{g_{ijk}^{pqr;n-1}}{Q_{ijk}^{pqr;n-1}} \frac{u_{i+p,j+q,k+r}^n - u_{ijk}^n}{m(\sigma_{ijk}^{pqr})}. \tag{18}$$

In order to write the discrete form of FBLSCD, we define for $(p,q,r) \in N_{ijk}$

$$v_{ijk}^{pqr} = m(e_{ijk}^{pqr}) \left(\delta \frac{u_{i+p,j+q,k+r}^{n-1} - u_{ijk}^{n-1}}{Q_{ijk}^{pqr;n-1} m(\sigma_{ijk}^{pqr})}\right). \tag{19}$$

We will distinguish between the outflow and inflow boundaries by defining two sets of indices $N_{ijk}^{out} = \{(p,q,r) \in N_{ijk}\,,\, v_{ijk}^{pqr} > 0\}\,, \quad N_{ijk}^{in} := \{(p,q,r) \in N_{ijk}\,,\, v_{ijk}^{pqr} \le 0\}\,.$ If we use the upwind principle for approximating the first integral on the RHS in (13), we obtain

$$\begin{aligned}
\int_{V_{ijk}} v \cdot \nabla u^{n-1} &\approx \sum_{N_{ijk}^{out}} u_{ijk}^{n-1} v_{ijk}^{pqr} + \sum_{N_{ijk}^{in}} u_{i+p,j+q,k+r}^{n-1} v_{ijk}^{pqr} - u_{ijk}^{n-1} \sum_{N_{ijk}} v_{ijk}^{pqr} \\
&= \sum_{N_{ijk}^{in}} (u_{i+p,j+q,k+r}^{n-1} - u_{ijk}^{n-1}) v_{ijk}^{pqr}\,.
\end{aligned} \tag{20}$$

The diffusion term is approximated as follows

$$\int_{V_{ijk}} \mu |\nabla u^{n-1}| \nabla \cdot \left(\frac{\nabla u^n}{|\nabla u^{n-1}|}\right) \mathrm{d}x \approx \mu \bar{Q}_{ijk}^{n-1} \sum_{N_{ijk}} m(e_{ijk}^{pqr}) \frac{u_{i+p,j+q,k+r}^n - u_{ijk}^n}{Q_{ijk}^{pqr;n-1} m(\sigma_{ijk}^{pqr})}. \tag{21}$$

The final fully discrete form of FBLSCD is given by

$$m(V_{ijk})\frac{u^n_{ijk}-u^{n-1}_{ijk}}{\tau_C} = \sum_{N^{in}_{ijk}}(u^{n-1}_{ijk}-u^{n-1}_{i+p,j+q,k+r})v^{pqr}_{ijk} + \mu\bar{Q}^{n-1}_{ijk}\sum_{N_{ijk}} m(e^{pqr}_{ijk})\frac{u^n_{i+p,j+q,k+r}-u^n_{ijk}}{Q^{pqr;n-1}_{ijk}m(\sigma^{pqr}_{ijk})}, \quad (22)$$

where v^{pqr}_{ijk} is substituted by (19).

The last step is the full discretization of the GSUBSURF equation which is analogous to FBLSCD. Again, we define

$$v^{pqr}_{ijk} = m(e^{pqr}_{ijk})\left(-w_{\text{con}}\frac{g_{i+p,j+q,k+r}-g_{ijk}}{m(\sigma^{pqr}_{ijk})}\right), \quad (23)$$

where we consider the approximation of $\nabla g.\nu^{pqr}_{ijk}$. Then the discrete convection term is exactly of the form (20). For the diffusion term we have

$$\int_{V_{ijk}} w_{\text{dif}} g|\nabla u^{n-1}|\nabla\cdot\frac{\nabla u^n}{|\nabla u^{n-1}|}\,\mathrm{d}\mathbf{x} \approx w_{\text{dif}}\, g_{ijk}\,\bar{Q}^{n-1}_{ijk}\sum_{N_{ijk}} m(e^{pqr}_{ijk})\frac{u^n_{i+p,j+q,k+r}-u^n_{ijk}}{Q^{pqr;n-1}_{ijk}m(\sigma^{pqr}_{ijk})}. \quad (24)$$

The fully discrete form of the GSUBSURF equation reads as follows

$$m(V_{ijk})\frac{u^n_{ijk}-u^{n-1}_{ijk}}{\tau_S} = \sum_{N^{in}_{ijk}}(u^{n-1}_{ijk}-u^{n-1}_{i+p,j+q,k+r})v^{pqr}_{ijk} + w_{\text{dif}}\, g_{ijk}\,\bar{Q}^{n-1}_{ijk}\sum_{N_{ijk}} m(e^{pqr}_{ijk})\frac{u^n_{i+p,j+q,k+r}-u^n_{ijk}}{Q^{pqr;n-1}_{ijk}m(\sigma^{pqr}_{ijk})}, \quad (25)$$

where v^{pqr}_{ijk} is given by (23).

All the discrete equations (18), (22), (25) together with boundary conditions represent linear systems with unknowns u^n_{ijk}. These systems can be solved by SOR method. Since we start the iterative process using the result from the previous time step, we have a good initial approximation and SOR method is sufficiently fast.

5. PARALLELIZATION OF THE ALGORITHMS

The microscopes produce images with a resolution that results in a huge amount of data. Our algorithms were used to process 3D images of size up to 50MB (using one-byte representation of the image intensity) and the newest devices produce data that can be much larger. Since several double precision floating point arrays with dimensions corresponding to the image size are used in the implementations of our algorithms, the memory requirements of the programs highly exceed the limits of a standard single processor machine and the algorithms have to be parallelized.

The parallelization of all methods that we mentioned is based on the same principle. Let us note that nuclei and cell shape segmentation do not usually require

a parallel approach because the corresponding equation can be solved only on a small subvolume surrounding the segmented object. Filtering, center detection and embryo segmentation are mostly not suitable for serial computing.

We use MPI interface and the so called SPMD (single program multiple data) model, where each parallel process executes the same code on a different set of data. First, we split the data into several parts (proportional to the number of processors). Then we rewrite the serial program in such a way that each process handles the correct part of data and transmits necessary information to other processes. Figure 6 shows our data distribution among the processes.

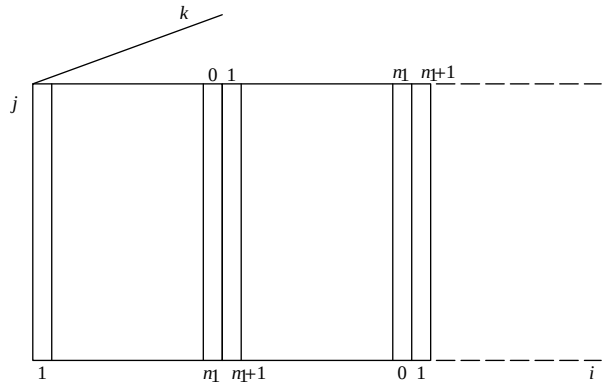

Fig. 6. Data distribution and its overlap over parallel processes (Figure from [13]).

3D images are represented by 3D arrays indexed by i, j, k. They are given in the index ranges $i = 1, \ldots, N_1$, $j = 1, \ldots, N_2$, $k = 1, \ldots, N_3$. The computational domain given by the finite volume space discretization is equal to the image domain. In order to distribute the data (i. e. the image as well as the solution of the corresponding equation) to n_{procs} processors, we define $n_1 = N_1/n_{\text{procs}} + 1$, $n_1^{\text{last}} = N_1 - (n_{\text{procs}} - 1)n_1$ and we set $n_2 = N_2$, $n_3 = N_3$. We store the first part of 3D data, i. e. $i = 1, \ldots, n_1 + 1$, $j = 1, \ldots, n_2 + 1$, $k = 1, \ldots, n_3 + 1$, cf. Figure 5, on the root process with rank 0. The next process with rank 1 handles the next part of the image, i. e. all 2D slices $j = 1, \ldots, n_2 + 1$, $k = 1, \ldots, n_3 + 1$ locally indexed by i in the range $i = 0, \ldots, n_1 + 1$, where the 2D slice with $i = 0$ corresponds to the slice with index $i = n_1$ in the root process with rank 0. The data distribution is similar on further processes. For the last process, we have $i = 1, \ldots, n_1^{\text{last}}$. The complete solution is obtained by merging the solutions from the individual processes. The data overlap is necessary for information exchange between neighboring processes in order to solve iteratively the linear system and compute its coefficients.

All our algorithms use the SOR method in order to solve the corresponding linear system. However, there is a dependence of the currently updated value u_{ijk} on its six neighbors $u_{i+p,j+q,k+r}$, $(p, q, r) \in N_{ijk}$. Such dependence is not well suited for the parallelization but there exists an elegant way how to change the standard SOR method so that it can be efficiently parallelized, cf. e. g. [1]. All voxels are split

in two groups. RED elements are those with the sum of their indexes equal to an even number. Conversely, BLACK elements fulfill the condition that the sum of their indexes is an odd number. Then the six neighbors of a RED element are BLACK elements and the value of a RED element depends only on the BLACK elements, and vice-versa. Due to this fact, we can split one SOR iteration in two steps. First we update RED elements simultaneously on all processors. Then the updates in overlapping slices are interchanged between neighboring processes using MPI functions. After, the BLACK elements are updated on all processors and again, the updates from the overlapping regions are transferred between the neighboring processes. This completes one step of the RED-BLACK SOR algorithm which we use in all parallel versions of our methods.

6. APPLICATIONS AND EXPERIMENTS

In this section we will discuss the practical aspects of the methods introduced in the above text. First, we will examine the behaviour of the numerical scheme by determining its experimental order of convergence. After, we will illustrate how our image processing chain works on typical data. We will start by showing some results of filtering, after the center detection and finally some segmented objects. Each part will start with some sort of validation of the corresponding method. Finally we will explain how the results of segmentation can be used to analyze the process of embryogenesis.

6.1. Experimental order of convergence of the numerical schemes

In order to verify the convergence properties of the proposed schemes, we made several experiments to examine the experimental order of convergence for various problems. Such tests confirm the reliability of our discretization methods.

A representative example (with known exact solution) for all our models is a simple level set equation in the following form

$$\partial_t u = |\nabla u| \nabla \cdot \left(\frac{\nabla u}{|\nabla u|} \right) , \tag{26}$$

resp. its regularized variant

$$\partial_t u = \sqrt{\varepsilon + |\nabla u|^2}\, \nabla \cdot \left(\frac{\nabla u}{\sqrt{\varepsilon + |\nabla u|^2}} \right) . \tag{27}$$

The exact solution of (26) is

$$u(x, y, z, t) = \frac{x^2 + y^2 + z^2 - 1}{4} + t \tag{28}$$

and let us consider Dirichlet boundary conditions and the initial condition given by this solution. The regularized problem (27) was solved in the time domain $I = [0, 0.16]$ and in the spatial domain $\Omega = [-1.25, 1.25]^3$ that consisted of n^3

voxels and the space step was $h_1 = h_2 = h_3 = h = 2.5/n$. The time step τ was taken proportional ho h^2. The error of the numerical solution was measured in $L_\infty(I, L_2(\Omega))$ norm which is natural for testing the schemes for solving parabolic problems. The results of our tests are shown in Table 1. We display results for gradually refined grid and also for various values of ε. We can see that for this coupling of the space and time steps, our schemes are second order accurate for this type of smooth solution.

Table 1. Experimental order of convergence for the reduced diamond cell scheme, equation (27), analytical solution (28). The $L_\infty(I, L_2(\Omega))$ errors and corresponding EOC are displayed for various values of ε.

n	τ	$\varepsilon = h$	EOC	$\varepsilon = h^2$	EOC	$\varepsilon = 10^{-6}$	EOC
10	0.04	4.9584e-2		9.6094e-3		4.3292e-3	
20	0.01	1.6991e-2	1.545	1.3388e-3	2.844	1.0697e-3	2.017
40	0.0025	4.4410e-3	1.936	8.8258e-4	2.244	2.6622e-4	2.007
80	0.000625	1.1104e-3	2.000	6.7506e-5	2.066	6.6493e-5	2.001
160	0.00015625	2.7709e-4	2.003	1.6681e-5	2.017	1.6618e-5	2.000

Now let us try a more difficult example. We consider a highly singular solution of (26) given by

$$u(x, y, z, t) = \min\left(\frac{x^2 + y^2 + z^2 - 1}{4} + t, 0\right). \tag{29}$$

We use the same space and time step coupling $\tau\ h^2$ and because the exact solution is not smooth, we measure the error in $L_1(I, L_1(\Omega))$ norm. The best convergence properties were observed for $\varepsilon = h^2$ and therefore we show just results for this choice. Table 2 displays the results for gradually refined grid similarly as in the previous case. We can see that our method converges also for the singular solution, although the EOC is now less than 2.

Table 2. Experimental order of convergence for the reduced diamond cell scheme, equation (27), analytical solution (29).

n	τ	$L_1(I, L_1(\Omega))$	EOC
10	0.04	2.089161e-2	
20	0.01	9.704067e-3	1.10626
40	0.0025	4.094488e-3	1.24491
80	0.000625	1.674962e-3	1.28955
160	0.00015625	6.949633e-4	1.26912

Another test was realized for the GSUBSURF problem. Since the analytical

solution of the equation (4) is not known, we consider a modified problem

$$\partial_t u - w_{\text{con}} \nabla g \nabla u - w_{\text{dif}} g \sqrt{\varepsilon^2 + |\nabla u|^2}\, \nabla \cdot \frac{\nabla u}{\sqrt{\varepsilon^2 + |\nabla u|^2}} = f(u)\,. \tag{30}$$

We consider the function $\tilde{u}(x, y, z, t) = t \cos(\pi x) \cos(\pi y) \cos(\pi z)$ to be the analytical solution of the problem. Then we get for $f(u)$:

$$f(u) = \partial_t \tilde{u} - w_{\text{con}} \nabla g \nabla \tilde{u} - w_{\text{dif}} g \sqrt{\varepsilon^2 + |\nabla \tilde{u}|^2}\, \nabla \cdot \frac{\nabla \tilde{u}}{\sqrt{\varepsilon^2 + |\nabla \tilde{u}|^2}}\,.$$

The function g was taken in the form $g(x, y, z) = \cos^2(\pi x) \cos^2(\pi y) \cos^2(\pi z)$. The values of model parameters were chosen as follows: $w_{\text{con}} = 1.0$, $w_{\text{dif}} = 1.0$, $\varepsilon = h^2$, $\Omega = [-0.5, 0.5]^3$, $T = 0.16$, $\tau = 4h^2$. The results are displayed in Table 3 where we again see the experimental order of convergence of the method.

Table 3. Experimental order of convergence for the reduced diamond cell scheme, equation (30).

n	τ	$L_\infty(I, L_2(\Omega))$	EOC
10	0.04	2.294529e-2	
20	0.01	9.546493e-3	1.26516
40	0.0025	3.636421e-3	1.39245
80	0.000625	1.478174e-3	1.29870
160	0.00015625	6.501010e-4	1.18508

6.2. The image processing chain

In all experiments below, we use image data representing the development of a zebrafish embryo. The imaging started approximately 4 hours after fertilization and continued for several hours. The time step Δt between the subsequent 3D images varies from 1 to 5 minutes. Each 3D image is constructed by scanning several dozens to hundreds of 2D slices, the size of each slice is typically 512×512 pixels. Each image in the time sequence usually contains several thousands of cells.

For most of the experiments, we used datasets with a low time step Δt ($\Delta t \approx 1\,\text{min}$). Such datasets are suitable for good cell tracking but due to the high speed of scanning, they suffer from some inevitable loss of quality. Anyway, as we will see, most of the algorithms are able to operate well also on such data. Some of the experiments, especially for the membrane segmentation, were performed on data with a higher Δt and better signal, as the membrane images are in general more susceptible to the presence of artifacts, incomplete borders and spurious inner structures. Many applications of the membrane segmentation, e. g. determining the shape of cells or the area of the cell contact surface, are not connected to the cell tracking and therefore they do not depend so strongly on the speed of the scanning.

Before starting, let us make a short remark about the presentation of the results. Where possible, we use 3D visualization. In some cases, especially for the segmentation results, we show 2D slices that provide a better insight to the data and illustration of the ideas. However, such a visualization is more likely to emphasize local errors and distract the three-dimensional relations. This fact should be taken in account while viewing such figures.

6.2.1. Image filtering

The image filtering is a widely studied topic and there are a number of techniques able to efficiently remove the noise from the image data. For our purposes, we tested several nonlinear diffusion models, namely the basic mean curvature flow (MCF), slowed mean curvature flow (SMCF), geodesic mean curvature flow (GMCF) and the regularized Perona–Malik model (PM). Detailed description of the methods can be found in [10].

The filtering results were quantified using the mean Hausdorff distance between the manually segmented nuclei surfaces – the 'gold standard' and the corresponding nuclei isosurfaces in the original and filtered data. Further tested criterion for filtering is related to the splitting ability of artificially connected neighboring nuclei due to the image acquisition error.

First, let us define the mean Hausdorff distance. Let $A = \{a_1, \ldots, a_p\}$ and $B = \{b_1, \ldots, b_q\}$ denote two finite point sets. Then the *mean Hausdorff distance* is defined as, cf. [18],

$$MHD(A, B) = \max\left(mhd(A, B), mhd(B, A)\right),$$

where

$$mhd(A, B) = 1/p \sum_{i=1}^{p} \min_{b \in B} \|a_i - b\|$$

is called *mean directed Hausdorff distance* and $\|\cdot\|$ is an underlying norm (usually Euclidean). The mean Hausdorff distance is widely used to measure the similarity between two point sets, cf. e. g. [8, 18]. In our case the sets A and B are given by discrete points that form the surface of the gold standard and an isosurface either in an original or in a filtered volume.

The results of the comparison are presented in Figure 7 and Table 4. The optimal filtering parameters for each method were chosen according to the mean Hausdorff distance observed for a large range of isosurfaces. We can see that the mean curvature flow achieves the minimum value of the mean Hausdorff distance for the isosurface value 25. It has the highest smoothing capability in a small range of intensity levels in a neighborhood of this value. This leads to a strongly convex graph which indicates significant changes in the distribution of the image intensity and consequent modifications of shape of the objects in the image. Other filtering techniques significantly reduce the mean Hausdorff distance for a large range of isosurfaces. We can observe that the graph of GMCF has the lowest convexity for lower intensities which represents the robustness of this method, in other words, any of the corresponding isosurfaces can be used for correct representation of the

nucleus because they all have more or less the same Hausdorff distance from the gold standard.

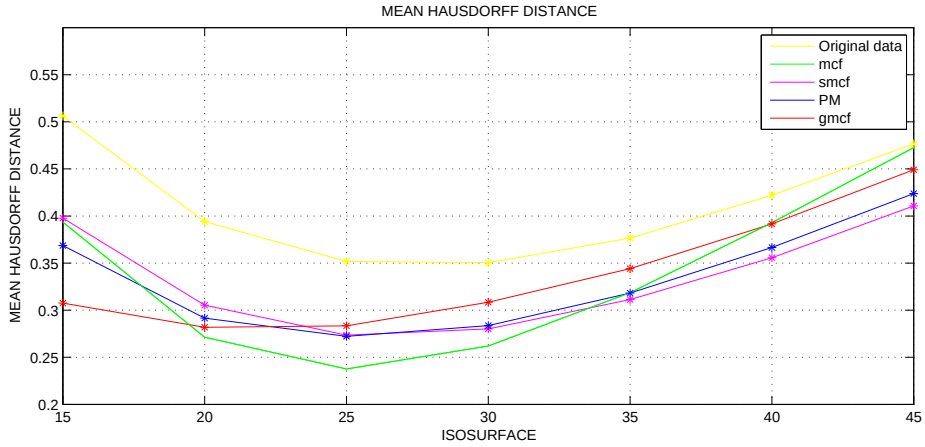

Fig. 7. Graphs of the mean Hausdorff distances for the original data and data filtered using the optimal values of parameters. (The figure from [10].)

Table 4. The mean Hausdorff distances in μm for the original and filtered data using the optimal values of parameters. (The table from [10].)

Isosurface	Original	MCF	SMCF	PM	GMCF
15	0.506	0.394	0.411	0.369	0.308
20	0.394	0.271	0.302	0.292	0.282
25	0.352	0.238	0.265	0.272	0.283
30	0.350	0.262	0.274	0.284	0.308
35	0.377	0.319	0.313	0.318	0.344
40	0.422	0.393	0.364	0.367	0.392
45	0.476	0.473	0.429	0.424	0.449

Another experiment inspected the capability of the methods to smooth the small intensity variations and preserve or even improve the edge information. Figure 8 shows the graphs of the image intensity along a line crossing a nucleus. We can see that the line representing GMCF is totally flat outside the nucleus. Such flattening of low intensity regions causes enhancement of the nuclei boundaries.

The last tested property was the ability to split spuriously connected objects that occur in the images due to the physical constraints of microscopy. A subvolume of the original data with 22 connected regions was selected and the number of connected regions in filtered subvolumes was evaluated, cf. Table 5. The result shows that the GMCF method was able to divide 50 % of the spuriously connected regions.

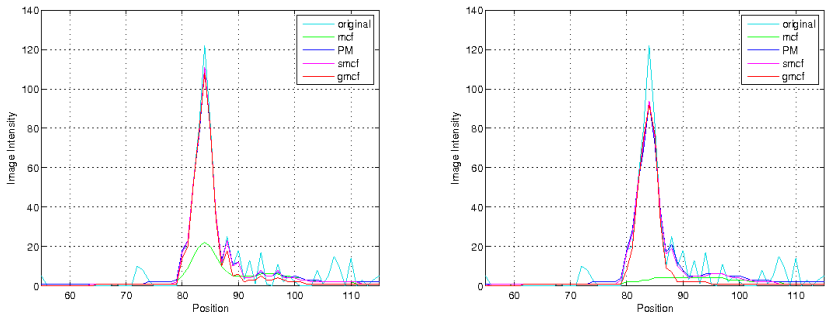

Fig. 8. Graphs of the original and filtered image intensities in a neighborhood of a nucleus after 5 and 15 time steps. (The figure from [10].)

Table 5. The number of spuriously connected regions found in original and filtered data using the optimal choice of parameters. (The table from [10].)

Original	MCF	SMCF	PM	GMCF
22	20	21	18	11

All the above mentioned merits of the GMCF led to the choice of this filtering technique for our image algorithm chain.

Figures 9 and 10 show a few examples of the GMCF filtering. We used a uniform rectangular grid with $h_1 = h_2 = h_3 = 0.01$ and the time step τ_F is indicated below the figures. The values of the other model parameters were $K = 5.0$ and $\sigma = 0.0001$. The norm of gradient $|\nabla u^{n-1}|$ was regularized with $\varepsilon = 0.01$. Figure 9 shows 2D slices of filtered nuclei and membrane images compared to original data. Figure 10 illustrates how the filtration process gradually removes the noise from the original image.

6.2.2. Center detection

In order to validate our object counting algorithm, we performed a manual center detection on a subvolume of a nuclei image and we tested the correctness of the result obtained by automatic detection. Considering the further applications of the center detection, there are three basic types of errors that can occur. *False positive detection* is the case when the algorithm finds a center that does not correspond to any important object in the image. This happens especially in large epithelial nuclei that can contain inner structures that cause detection of more than one center in a single nucleus. Another false positive case is detection of a center that corresponds to some artifact or noise structure. *False negative detection* is the case when an object was not detected. This is usual for small nuclei of very low intensity. Finally, if the detected center is too far from the center of mass of the object, it can cause

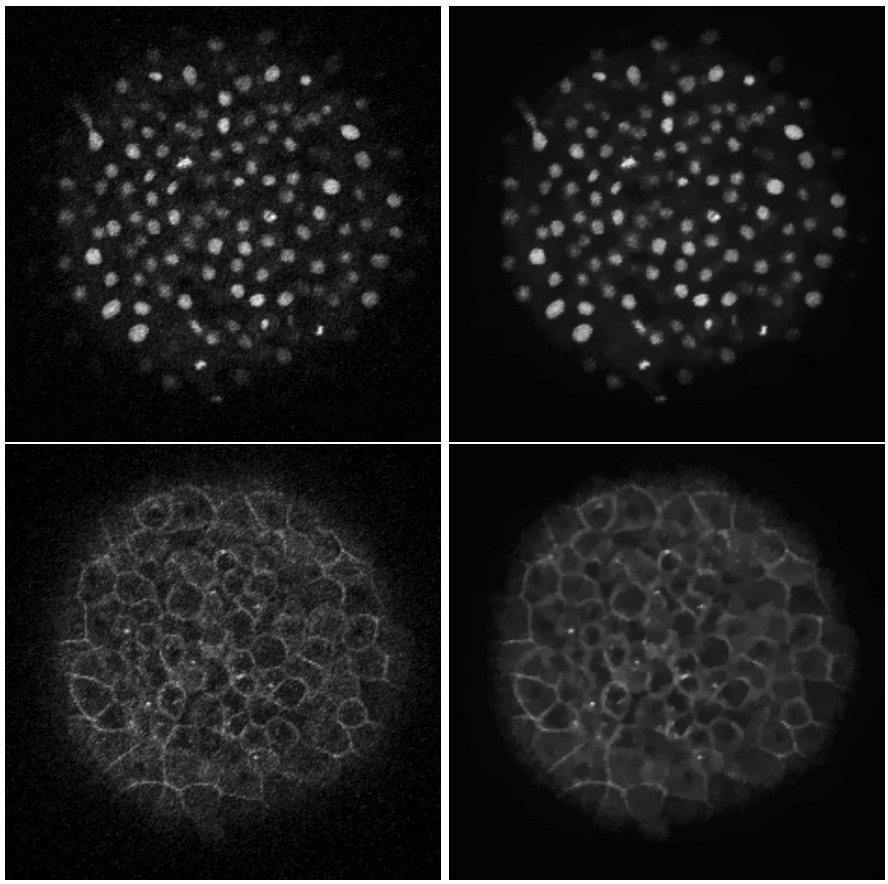

Fig. 9. Comparison of original noised and filtered images. In the left column, we can see the original data, in the right column the corresponding results of GMCF filtering. We used $\tau_F = 0.0002$ and 5 time steps for nuclei images and 10 steps for membrane images.

some problems in the segmentation process as the segmentation result significantly depends on the location of the compact support of the initial segmentation function which is given by the position of the segmentation seed.

In order to measure the quality of our automatic object detection, we evaluated the number of false positive (FP) and false negative (FN) cases and the average distance $\bar{D}$ of an automatically detected center to the corresponding center found by hand. The results are presented in Table 6 and Figure 11. As we can see, for this particular subvolume the algorithm found 102 out of 106 nuclei, with no false positives. The false negative detections appear only on the border of the subvolume. The average error in the position of the centers is small compared to the diameter of the nuclei which is typically from 10μm to 20μm. 45 % of the centers were detected at

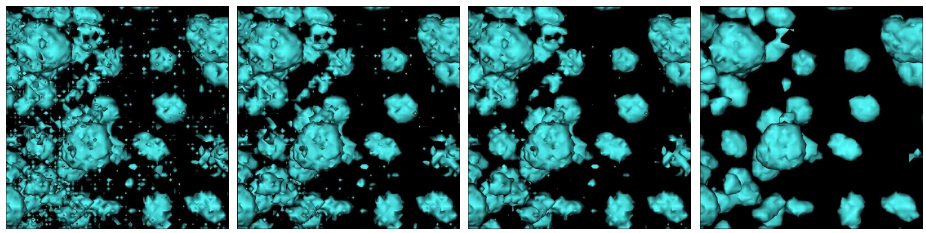

Fig. 10. The GMCF filtering process illustrated on the isosurface $I = 30$ of the data. The pictures show the result after 0, 5, 10 and 40 time steps. τ_F was 2.10^{-5}.

the exact position, the others are slightly shifted which does not cause any problem for the subsequent segmentation.

The chosen subvolume is a representative part of the inner cell layers of the embryo. For the outer (epithelial) layer, we would usually observe false positive cases. In Sec. 6.3, we explain how we can correct such errors. Higher rate of false negative cases can occur in the parts with weaker staining. Discovering or correcting false negative detections would be much more difficult than correction of the false positives. Therefore, for data with varying level of staining, the algorithm must be set up in a way that avoids production of false negative detections, even if it often means to accept a certain number of (usually correctable) false positives. In practice it means that the stopping criterion must be adjusted, making the FBLSCD algorithm stop at an earlier stage of the evolution.

The parameters for this test were set as follows. The space grid is uniform with $h_1 = h_2 = h_3 = 0.01$ and $\tau_C = 0.00125$. Further we set $\delta = 1.0$, $\mu = 0.00125$ and we regularize $|\nabla u^{n-1}|$ by $\varepsilon = 10^{-6}$ in the convection term and $\varepsilon = 1.0$ in the diffusion term. Even though the images are filtered, they can be still deteriorated by a certain level of low intensity noise. In such case we apply an additional FBLSCD parameter – threshold. Then the detection algorithm ignores the local maxima which are below the chosen threshold value. This value depends on the intensity of the objects and the noise in the image. In this experiment we used $thres = 0.08$, considering the image intensity from the interval $[0; 1]$.

Table 6. Evaluation of the quality of the automatic center detection on a subvolume.

Centers manual	Centers FBLSCD	FP	FN	D	Exact positions
106	102	0	4	0.8197μm	48

Using the same parameters, we make an experiment on an image from another data set. The evolution is illustrated in Figure 12 and 13 which give us an image of how the FBLSCD method works. In Figure 12, we visualize the centers detected in the original image, in the image filtered by GMCF and then in time steps $2, 3$

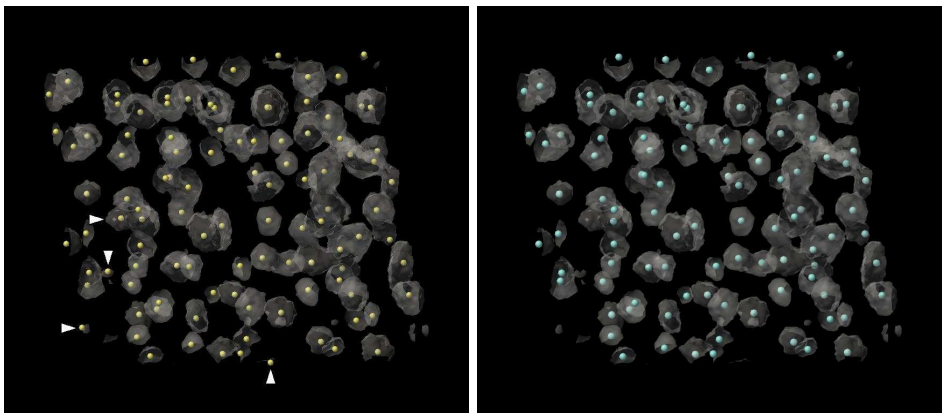

Fig. 11. Comparison of the nuclei centers detected manually (left)and by FBLSCD (right). On the left, the centers missing in the automatic detection are indicated.

and 9 of FBLSCD. In the original image, we can see a large amount of redundant centers. After GMCF filtering, the number of centers is reduced, though it is still too high, and due to formation of regions with constant intensity, some local maxima cannot be localized. In time step 2 of FBLSCD, we can still observe about 25% of extra centers but what is important, each object contains at least one detected local maximum. Then, a single time step is sufficient to smooth the subtle intensity peaks inside the nuclei and only 3 redundant centers corresponding to tiny noise structures remain. In time step 9, all redundant centers are removed while we still have no false negative detection. Figure 13 presents the intensity profile of a 2D slice of the original and filtered data, and afterwards a 2D slice of the function evolving by the FBLSCD model (3). The graph of the evolution of the number of centers is also provided and it shows that the process is really most stabilized at time step 9, when the number of detected centers is 30.

6.2.3. Cell nuclei segmentation

In order to start the segmentation process, we first need to construct an appropriate initial segmentation function. If we deal with ideal images, the final result is almost independent of the initial function, but in our case the objects in images can be nonhomogeneous and the images can contain some artifacts or inner structures in the objects, or the objects can be artificially connected. Moreover, because there are millions of cells to process in one 3D+time data set, the segmentation function should evolve into its final shape very quickly. Therefore we try to set the starting form of the segmentation function as close to the actual segmented object as possible. The nuclei are in general approximately spherical or ellipsoidal objects, so we start from a function whose all isosurfaces are equal ellipsoids. The center of such an ellipsoid is given by the detected position of the nucleus center and the radii are

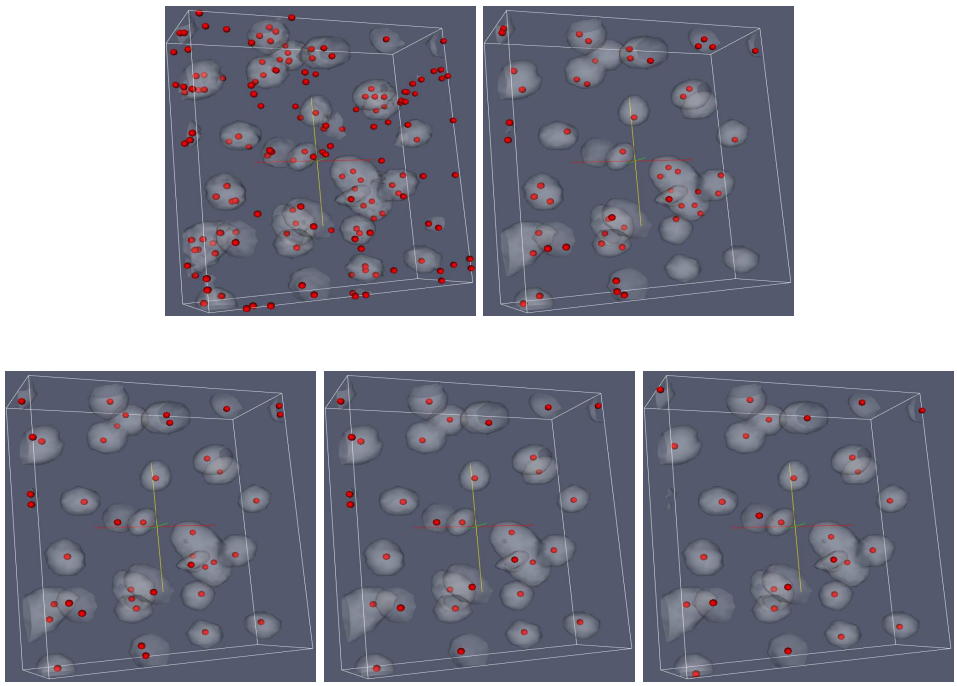

Fig. 12. The center detection process. Top left, local maxima in the original data (159). Top right, data filtered by GMCF (54 centers). Bottom, time steps 2 (42 centers), 3 (33 centers) and 9 (30 centers) of FBLSCD.

chosen by estimating the size of the cell, based on measuring the distance from its nearest neighbor.

The validation, similarly as in the case of image filtering, was done by measuring the mean Hausdorff distance between the manually segmented surfaces (the manual segmentation was done in ITK-SNAP [16]) and an isosurface ($I = 127$) of the segmentation function obtained by GSUBSURF method. The model parameters were chosen as follows: $g(s) = G_\rho * \frac{1}{(1+Ks^2)^6}$, $\rho = 0.0001$, $K = 1000$, $\sigma = 0.0$, $w_{\text{con}} = 10.0$, $w_{\text{dif}} = 2.0$ (note that the convolution was in this case applied to the edge detector instead of the image data). We used time step $\tau_S = 0.1$ and voxel dimensions $h_1 = h_2 = h_3 = 1.0$, $T_S = 5$. These values were found by thorough testing on various zebrafish data, with the goal to minimize the CPU time and maximize the accuracy. They also minimize the mean Hausdorff distance from the gold standard in this test example.

The results of the test are presented in Table 7 and Figure 14. The gold standard consists of 28 cells. The Hausdorff distance between all these cells and their corresponding GSUBSURF counterparts was computed and Table 7 displays the minimum, maximum and arithmetic mean of these distances. The gold standard

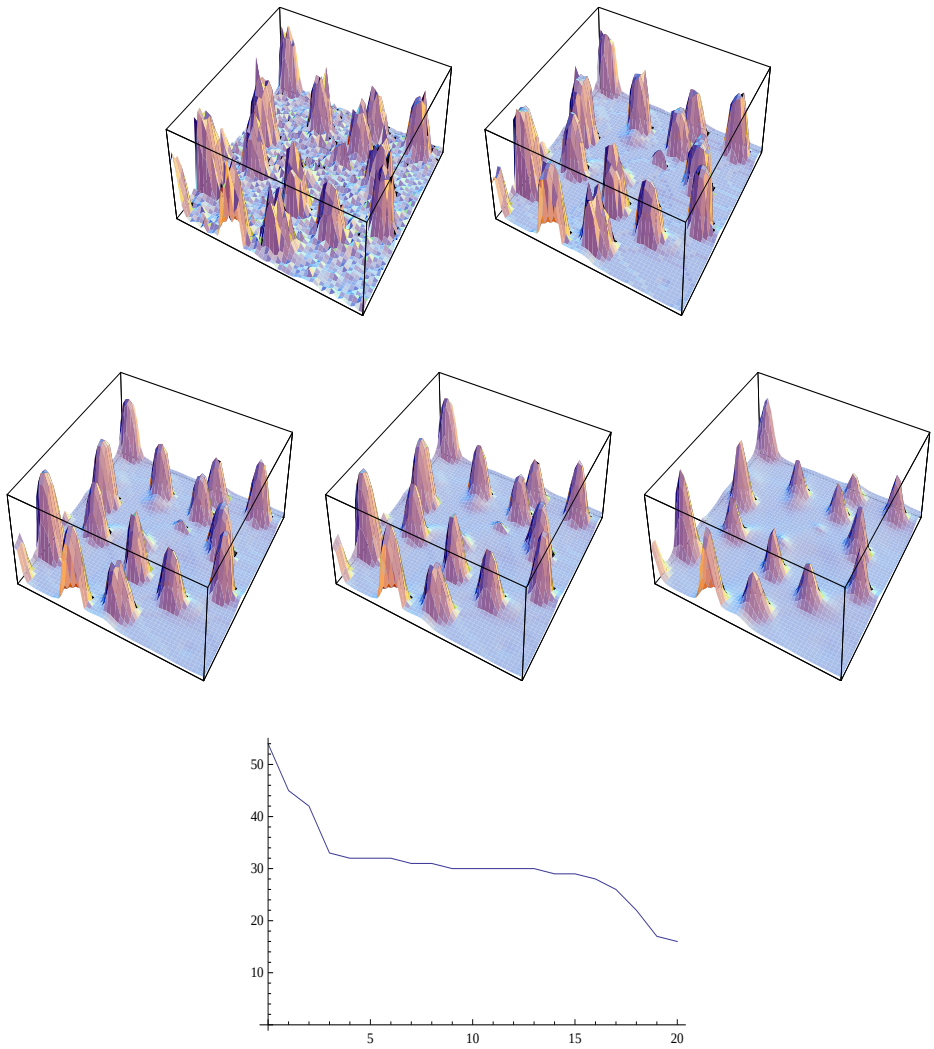

Fig. 13. The center detection process. Top, profile of a 2D slice of the original and filtered data. Middle, profile of a 2D slice of the function evolving by FBLSCD, time steps $2, 3, 9$. Bottom, time evolution of the number of detected centers. A flat region starting at time step 9 indicates that the process is stabilized and can be stopped.

cells are a part of the subvolume used for FBLSCD validation, so we recall that the typical diameter of the nucleus is $10\mu\text{m} - 20\mu\text{m}$. Comparing with this dimension, we can say that the GSUBSURF method provides reliable results.

Using the same parameters, we segmented two complete nuclei images representing two different stages of the embryonic development. The result is shown in

Table 7. The mean Hausdorff distance of the automatically segmented nuclei from the gold standard.

Cells	Min. MHD	Max. MHD	Mean MHD
28	0.270μm	1.590μm	0.654μm

Figure 15.

Let us also remark that extensive testing of the generalized subjective surface method really proved its efficiency and applicability in practice. For suitably chosen parameters w_{con}, w_{dif}, segmentation of one nucleus on a standard CPU does not take more than a few seconds and it appears to be several times faster than the classical subjective surface approach ($w_{\mathrm{con}} = 1.0, w_{\mathrm{dif}} = 1.0$) or the segmentation using explicit finite difference schemes [17].

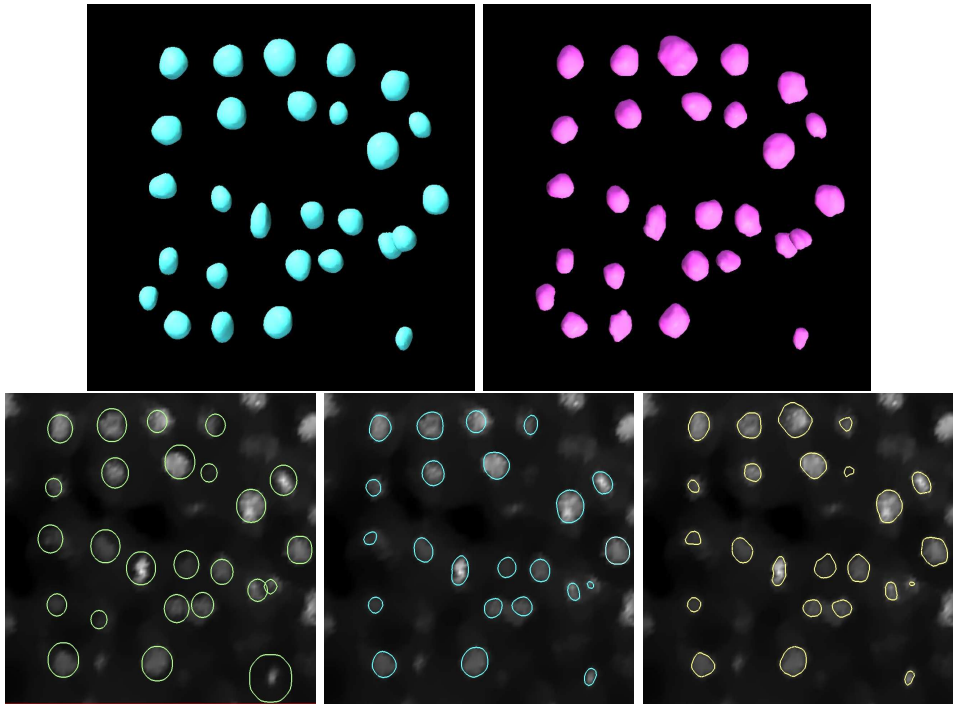

Fig. 14. Comparison of the GSUBSURF segmentation of cell nuclei with manual segmentation. Top left, nuclei segmented by GSUBSURF. Top right, manually segmented nuclei. Bottom left, the initial condition for the GSUBSURF segmentation, 2D slice. Bottom middle, the result of GSUBSURF segmentation. Bottom right, the manual segmentation.

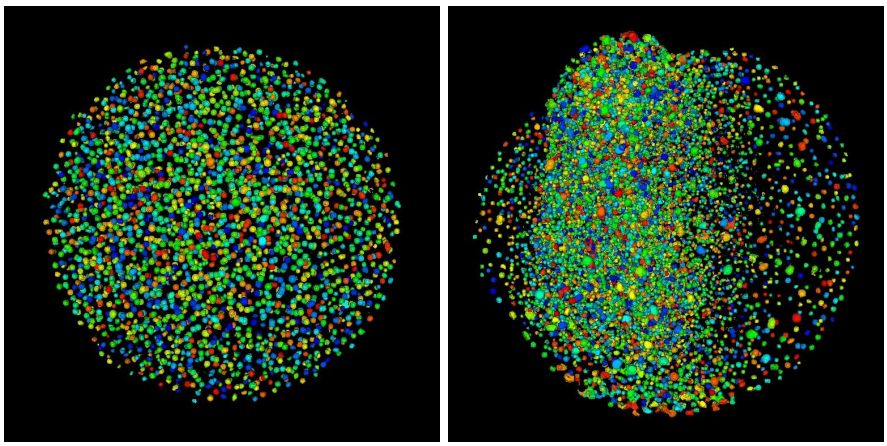

Fig. 15. Segmented nuclei of the embryo, isosurface $I = 127$ of the segmentation function. On the left, time step 1 of the data series, on the right, time step 500.

6.2.4. Cell membrane segmentation

The validation of the membrane segmentation was done in the same way as in the case of cell nuclei. 10 manually segmented cells represent the gold standard that was compared with the GSUBSURF segmentation. The results of the test are shown in Table 8 and Figure 16. The size of the cells in the sample was approximately $10\mu\text{m} - 15\mu\text{m}$.

For the segmentation of the whole cells, we use an initial condition which is naturally more similar to the cell structure than simple spheres or ellipsoids. We start from an ellipsoid situated in the nucleus center as in the case of nuclei segmentation. Then, when some point inside this ellipsoid is closer to the center of another cell, the value of the segmentation function in this point is set to zero. In this way, we obtain an initial segmentation function of Voronoi type. The parameters for segmentation were set as follows. The edge detector $g(s) = \frac{1}{1+Ks^2}$, $K = 1000$, $w_{\text{con}} = 10.0$, $w_{\text{dif}} = 0.2$, $\tau_S = 0.1$, $h_1 = h_2 = h_3 = 1.0$. Again, the method appears to be faster than the classical subjective surface method, though we need to perform approximately 10 times more time steps of the segmentation as in the case of nuclei.

Table 8. The mean Hausdorff distance of the automatically segmented cells from the gold standard.

Cells	Min. MHD	Max. MHD	Mean MHD
10	0.348μm	0.977μm	0.648μm

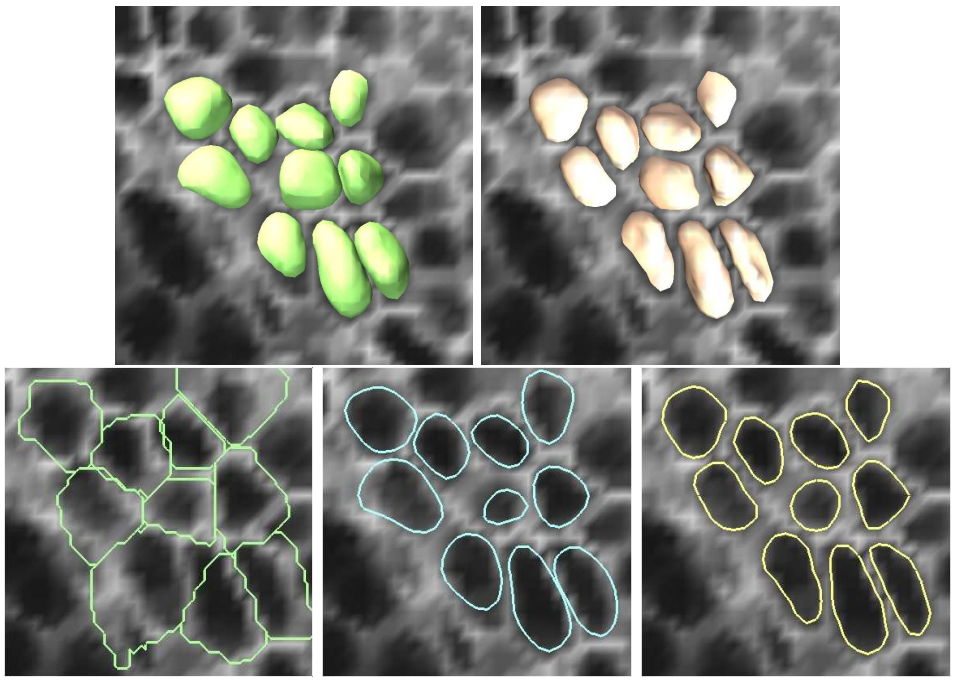

Fig. 16. Comparison of the GSUBSURF segmentation of the whole cells with manual segmentation. Top left, cells segmented by GSUBSURF. Top right, manually segmented cells. Bottom left, the initial condition for the GSUBSURF segmentation, 2D slice. Bottom middle, the result of GSUBSURF segmentation. Bottom right, the manual segmentation.

6.2.5. Embryo segmentation

The membrane images can be used not only for extraction of the shape of the individual cells but also for segmentation of the shape (or surface) of the whole embryo. In order to construct a suitable initial segmentation function, we place an ellipsoidal segmentation function in each cell center, similarly as in the case of nuclei segmentation, so that the union of all these functions covers the whole embryo. Then we start the segmentation process. Figure 17 shows the result of such a procedure. The parameters that we used were $g(s) = \frac{1}{1+Ks^2}$, $K = 1000$, $w_{\text{con}} = 10.0$, $w_{\text{dif}} = 2.0$, $\tau_S = 0.1$, $h_1 = h_2 = h_3 = 1.0$. The stopping time was $T_S = 100$.

6.3. Further applications in embryogenesis

Finally let us mention a few other interesting practical applications of the methods we introduced. The results of the image processing chain can provide a lot of useful information about the embryogenesis.

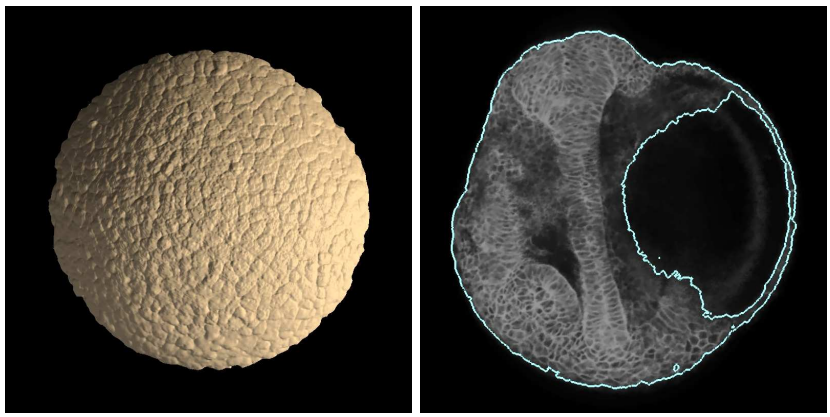

Fig. 17. Segmentation of the whole embryo. On the left, segmentation of time step 1 of the data set, 3D view of the isosurface $I = 40$ of the segmentation function. On the right, a 2D slice of data and the isosurface $I = 40$ of the segmentation function, time step 500.

6.3.1. Correction of the center detection

An essential information concerning the process of embryogenesis is the number of cells in each time step of the image sequence. The center detection method that we described provides a good estimate of this number, however, this can still be made more precise. What we have in mind while performing the FBLSCD is to avoid the false negative detection, that means losing some cell center. In each step of this multiscale process, there is a certain number of detected centers and we try to stop the procedure at a time moment when all real cell centers are still present. In practice, this means admitting a certain number of false positive detections, i. e. some extra centers that do not correspond to reality, usually double or multiple detection inside one nucleus. The amount of such centers is usually not higher than 5% of all detected centers. After the detection, we start the nuclei segmentation which represents a useful tool for correcting the detected centers. Having segmented all nuclei, we can simply compare their segmentation functions. If there is more than one center in one nucleus, the results of segmentation started from all these centers should be very similar to each other. By finding situations like this, we can easily eliminate the redundant centers and get a more accurate estimate of the number of cells in the embryo. The idea is illustrated in Figure 18.

6.3.2. Mitosis detection

Another interesting and important application useful especially in the lineage tree reconstruction, is the detection of cell divisions (mitoses). This is based on both nuclei and membrane segmentation. We take advantage of the fact that the process of cell division starts by division of the nucleus and the division of the whole cell takes place afterwards. This gives us the possibility to detect the anaphase stage

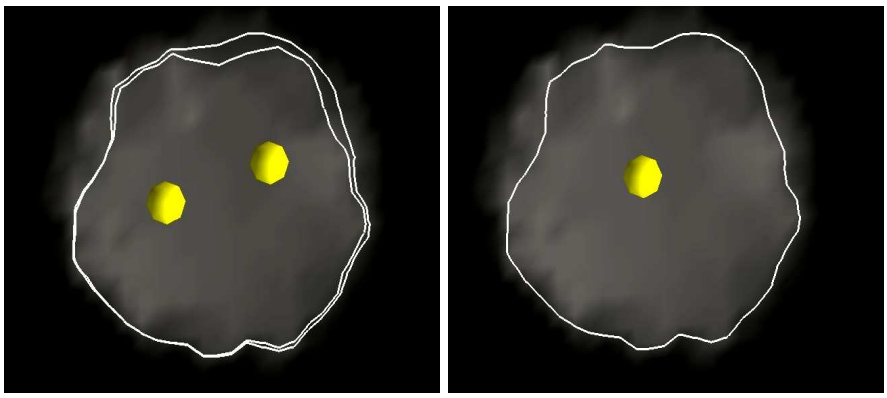

Fig. 18. Correction of nuclei centers. On the left, two centers found in one nucleus and their corresponding segmentation functions. On the right, the corrected center obtained by averaging the two detected centers and the segmentation function constructed by merging the original two functions.

of the mitosis when there are two nuclei in one cell. Having segmented the nuclei and membranes and obtained the correct positions of the cell centers, we have all information sufficient for a reliable mitosis detection. The idea is as follows. After segmentation of all cells using the membrane images, we compare the resulting segmentation functions corresponding to the centers that are situated close to each other. If we find out that segmentations starting from two different centers give comparable results, we can suppose that the two centers and corresponding nuclei are inside the same cell. This is the situation that we are interested in – a candidate for mitosis, see Figure 19. Using this approach, we find several possible cell divisions, however some of them are false positives. This can be caused by the noise in the image, lower local quality of the data, error in center detection or in the cell segmentation. Therefore, some more criteria have to be employed in order to obtain a reliable result. The basic criterion lies in a forward and backward tracking of the two centers in a short time scale (corresponding to a couple of time steps). If the mitosis really takes place, there should be always two centers in the neighborhood of the traced centers in the following time steps. On the other hand, going backwards in time, we should come to a time moment when there is just one center – this is the start of the mitosis. Other criteria that are applied deal with the volume of the nuclei corresponding to the centers and the distance between the centers. Essentially, the two nuclei arising in mitosis should have similar volume and the distance between them should increase going forward in time and decrease going backwards.

6.3.3. Quantitative characteristics of the embryo

The segmentation can also provide some important quantitative characteristics of the embryo. For example, using the membrane segmentation, we can obtain an estimate

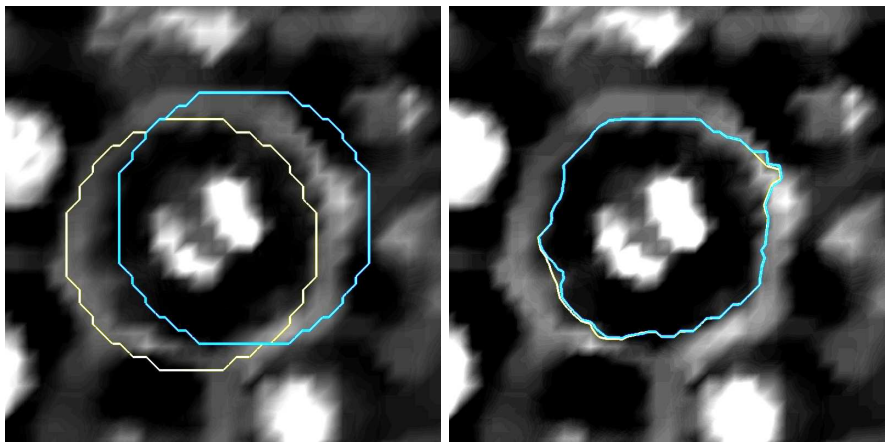

Fig. 19. Mitosis detection. On the left, we can see a 2D slice of data (nuclei and membranes superimposed) and the initial segmentation functions corresponding to the centers of the two nuclei. On the right, the result of segmentation which is almost the same for both centers.

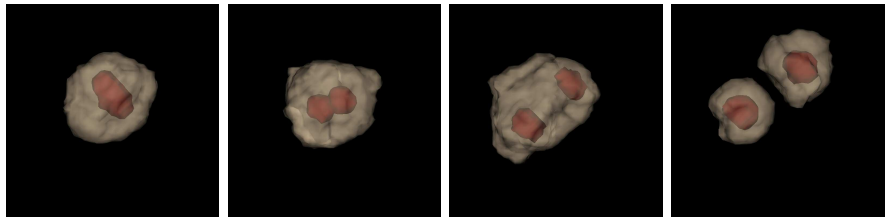

Fig. 20. The process of cell division, isosurfaces of nuclei and membrane segmentation functions are displayed.

of the contact surface between the cells or geometrical descriptors such as sphericity, flatness or elongation of the cells [15]. The segmentation of the whole embryo allows us to compute the volume or surface of the organism. In addition, if we know the number of cells in the embryo, we can calculate its global or local density of cells. Another interesting characteristic is the density of cell divisions in selected parts of the embryo. We can also explore just the epithelium (outer layer of cells) of the animal – its thickness, cell size, density of cells etc. All this information can be very useful in measuring the differences between various developing individuals and thus determining how much certain conditions (e. g. drug treatment) affect the standard process of embryogenesis.

Figure 21 shows one of the basic measurements – the local density of cells. Figure 22 displays a detail of membrane segmentation and the corresponding cell contact

surface (intercellular skeleton). The method for finding this surface was explained in [2].

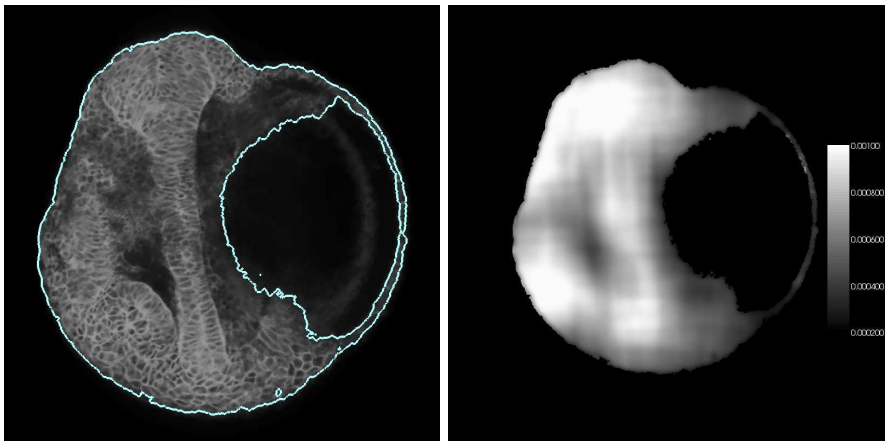

Fig. 21. Embryo segmentation and local density of cells in a 2D slice of the image data.

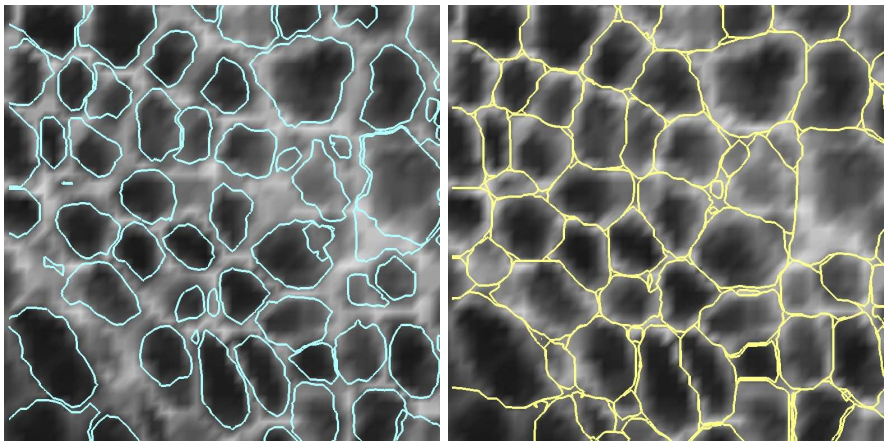

Fig. 22. Detection of the cell contact surface. Left, the membrane segmentation representing the starting point for the detection. Right, the corresponding intercellular skeleton.

ACKNOWLEDGEMENT

This work was supported by the European projects Embryomics and BioEmergences, the grants APVV-RPEU-0004-06, APVV-0351-07, APVV-LPP-0020-07 and the grant of

VEGA 1/0269/09. We would like to express our great thanks to all our collaborators participating in the projects and to supercomputing centre ccIN2P3 in Lyon where the large-scale parallel computations were performed.

(Received March 3, 2010)

REFERENCES

[1] Y. Aoyama, J. Nakano: RS/6000 SP: Practical MPI Programming. IBM 1999.

[2] P. Bourgine, P. Frolkovič, K. Mikula, N. Peyriéras, and M. Remešíková: Extraction of the intercellular skeleton from 2D microscope images of early embryogenesis. (Lecture Notes in Comp. Sci., *5567*.) (Proc. 2nd Internat. Conference on Scale Space and Variational Methods in Computer Vision, Voss 2009), Springer, Berlin pp. 38–49.

[3] V. Caselles, R. Kimmel, and G. Sapiro: Geodesic active contours. Internat. J. Comput. Vision *22* (1997), 61–79.

[4] Y. Chen, B. C. Vemuri, and L. Wang: Image denoising and segmentation via nonlinear diffusion. Comp. Math. Appl. *39* (2000), 131–149.

[5] S. Corsaro, K. Mikula, A. Sarti, and F. Sgallari: Semi-implicit co-volume method in 3D image segmentation. SIAM J. Sci. Comput. *28* (2006), 6, 2248–2265.

[6] P. Frolkovič and K. Mikula: Flux-based level set method: A finite volume method for evolving interfaces. Appl. Numer. Math. *57* (2007), 4, 436–454.

[7] P. Frolkovič, K. Mikula, N. Peyrieras, and A. Sarti: A counting number of cells and cell segmentation using advection-diffusion equations. Kybernetika *43* (2007), 6, 817–829.

[8] D. P. Huttenlocher, G. A. Klanderman, W. J. Rucklidge: Comparing images using the Hausdorff distance. IEEE Trans. Pattern Analysis and Machine Intelligence *15* (1993), 9, xxx–xxx.

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

[10] Z. Krivá, K. Mikula, N. Peyriéras, B. Rizzi, and A. Sarti: Zebrafish early embryogenesis 3D image filtering by nonlinear partial differential equations. Medical Image Analysis. Submitted for publication.

[11] K. Mikula, N. Peyriéras, M. Remešíková, and A. Sarti: 3D embryogenesis image segmentation by the generalized subjective surface method using the finite volume technique. In: Proc. FVCA5 – 5th International Symposium on Finite Volumes for Complex Applications, Hermes Publ. Paris 2008.

[12] K. Mikula and M. Remešíková: Finite volume schemes for the generalized subjective surface equation in image segmentation. Kybernetika *45* (2009), 4, xxx–xxx.

[13] K. Mikula and A. Sarti:, Parallel co-volume subjective surface method for 3D medical image segmentation. Parametric and Geometric Deformable Models: An application in Biomaterials and Medical Imagery, Vol. II, Springer Publishers, 2007, pp. 123–160.

[14] A. Sarti, R. Malladi, and J. A. Sethian: Subjective surfaces: A method for completing missing boundaries. In: Proc. National Academy of Sciences of the United States of America *12* (2000), 97, 6258–6263.

[15] O. Tassy, F. Daian, C. Hudson, V. Bertrandt, and P. Lemaire: A quantitative approach to the study of the cell shapes and interactions during early chordate embryogenesis. Currrent Biology *16* (2006), 345–358.

[16] P. A. Yushkevich, J. Piven, H. Cody Hazlett, R. Gimpel Smith, S. Ho, J. C. Gee, and G. Gerig: User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability. Neuroimage *31* (2006), 3, 1116–28.

[17] C. Zanella, M. Campana, B. Rizzi, C. Melani, G. Sanguinetti, P. Bourgine, K. Mikula, N. Peyriéras, and A. Sarti: Cells segmentation from 3-D confocal images Of early zebrafish embryogenesis. IEEE Trans. Image Process. *19* (2010), 3, 770–781.

[18] J. W. Zhang, G. Q. Han, and Y. Wo: Image registration based on generalized and mean Hausdorff distances. In: Proc. Fourth International Conference on Machine Learning and Cybernetics, Guangzhou 2005.

Paul Bourgine, CREA, Ecole Polytechnique-CNRS, 1 Rue Descartes, 75005, Paris. France
e-mail: paul.bourgine@polytechnique.edu

Róbert Čunderlík, Department of Mathematics, Slovak University of Technology, Radlinského 11, 813 68 Bratislava. Slovakia.
e-mail: cunderlik@math.sk

Olga Drblíková-Stašová, Department of Mathematics, Slovak University of Technology, Radlinského 11, 813 68 Bratislava. Slovakia.
e-mail: stasova@math.sk

Karol Mikula, Department of Mathematics, Slovak University of Technology, Radlinského 11, 813 68 Bratislava. Slovakia.
e-mail: mikula@math.sk

Mariana Remešíková, Department of Mathematics, Slovak University of Technology, Radlinského 11, 813 68 Bratislava. Slovakia.
e-mail: remesikova@math.sk

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

Barbara Rizzi, DEIS, University of Bologna, Via Risorgimento 2, 40136 Bologna. Italy.
e-mail: barbara.rizzi@unibo.it

Alessandro Sarti, DEIS, University of Bologna, Via Risorgimento 2, 40136 Bologna. Italy.
e-mail: asarti@deis.unibo.it