PDE Based Algorithms for Smooth Watersheds

Erlend Hodneland*, Xue-Cheng Tai, and Henrik Kalisch

***Abstract*—Watershed segmentation is useful for a number of image segmentation problems with a wide range of practical applications. Traditionally, the tracking of the immersion front is done by applying a fast sorting algorithm. In this work, we explore a continuous approach based on a geometric description of the immersion front which gives rise to a partial differential equation. The main advantage of using a partial differential equation to track the immersion front is that the method becomes versatile and may easily be stabilized by introducing regularization terms. Coupling the geometric approach with a proper "merging strategy" creates a robust algorithm which minimizes over- and under-segmentation even without predefined markers. Since reliable markers defined prior to segmentation can be difficult to construct automatically for various reasons, being able to treat marker-free situations is a major advantage of the proposed method over earlier watershed formulations. The motivation for the methods developed in this paper is taken from high-throughput screening of cells. A fully automated segmentation of single cells enables the extraction of cell properties from large data sets, which can provide substantial insight into a biological model system. Applying smoothing to the boundaries can improve the accuracy in many image analysis tasks requiring a precise delineation of the plasma membrane of the cell. The proposed segmentation method is applied to real images containing fluorescently labeled cells, and the experimental results show that our implementation is robust and reliable for a variety of challenging segmentation tasks.**

***Index Terms*—Biomedical image processing, partial differential equations, level set, object segmentation, object recognition, cells (biology).**

I. Introduction

WATERSHED segmentation is a widely used and well described tool for image segmentation [1]–[7]. The name of the method refers to a landscape flooded by water, where the watersheds are the dividing contours between the basins. These are the positions where water from different basins meets upon filling of depressions corresponding to local minima. The application of watershed by immersion [1], [7], [8] involves a recursive increase of gray levels similar to a water flooding a landscape. One may reformulate watershed by immersion in terms of a time-dependent transport equation [9]–[11], and this will be the basis for the new method proposed in this paper.

Watershed is well suited for the segmentation of objects outlined by high intensity ridges. On the other hand, many real-world objects are distinguishable from background as bright or dark objects surrounded by pronounced edges. For these cases, the watershed method can often be applied after the application of edge-enhancing pre-processing methods and the gradient operator, thus transforming the intensity based image into a ridge-based image. Still, for such images the first method of choice is often intensity based segmentation methods like clustering or level set segmentation. Thus, in this work, we focus on segmentation of objects that are naturally surrounded by ridges, where intensity based segmentation methods have insufficient performance.

The watershed method will normally result in severe over-segmentation due to a large number of local minima in a typical data set. Several methods exist to overcome this problem, for example marker-based algorithms such as [2], [12]–[14] and hierarchical watershed algorithms [6], [15]–[17].

The marker-based watershed segmentation initializes the flooding from markers placed inside natural minima of a certain depth or extent, and then makes the remaining image convex within each catchment basin. Practically speaking, all remaining natural minima are filled with a constant, the maximum value of the local minimum. By this method, the over-segmentation is implicitly restricted by the number of markers. With the aim to perform a fully automated segmentation, the markers should be generated automatically [2], [4], thus reducing the degree of human intervention for huge data sets. However, automatic generation of a one-to-one marker set is a non-trivial task in the presence of large image inhomogeneities and artefacts, thereby restricting the practical performance of marker-controlled watershed segmentation. Specifically relevant for the task of cell segmentation, one solution to the marker problem is to use data-driven marker generation, i.e. a segmentation of stained cell nuclei as markers [18]. However, stained nuclei are not always available due to a limited number of available channels for fluorescence imaging, as well as the Hoechst staining interfering with the DNA replication, which in some situations can alter the experimental conditions.

Among the hierarchical watershed algorithms, the viscous watershed transform [16] applies viscous closing prior to segmentation, thus obtaining a merging of local minima at low water heights. The hierarchical, graph-based watershed transforms described in [15], [17], also referred to as waterfalls, rely on intensity similarity of adjacent watershed regions as well as the definition of significant markers. Our merging protocol is related to these ideas as we also define significant objects, but rather use volume as a merging criterion for adjacent regions.

Manuscript received October 12, 2015; revised November 18, 2015; accepted November 19, 2015. Date of publication November 25, 2015; date of current version March 31, 2016. *Asterisk indicates corresponding author.*

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors.

*E. Hodneland is with the MedViz, 5021 Bergen, Norway, and also with Christian Michelsen Research, 5072 Bergen, Norway (e-mail: erlend.hodneland@gmail.com).

X.-C. Tai and H. Kalisch are with the Department of Mathematics, University of Bergen, NO-5020 Bergen, Norway.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMI.2015.2503328

Our method can easily be extended to include intensity values as well, depending on the data and whether homogeneous intensity values are to be expected for a continuous object. As pointed out in [17], a potentially useful trade off between volume and intensities are integrated regional intensities, accounting both for volume and intensity. In Jalba *et al.* [19], the authors applied the waterfall ideas in combination with the minimum spanning tree (MST) to generate robust and parameter insensitive segmentations within diffusion tensor imaging (DTI) data sets.

Common for most algorithms dealing with over-segmentation is that they are not simulatenously able to deal with the problem of oscillatory contours. The traditional watershed algorithm has no natural regularizing effect [7]. To avoid highly oscillating boundaries and to improve the overall segmentation quality a pre-smoothing of the input data is usually required for better results. Alternatively, with the aim of developing an integrated system for automated segmentation, we embed an anisotropic smoother into the segmentation algorithm in order to produce a combined model for simultaneous segmentation and regularization. By these means, the complexity of the algorithm can be reduced to a single processing step. Previous work on regularized watersheds has successfully been conducted [4], [6], [16]. However, the usefulness of [6] was only shown for 2D and their method will not, in contrary to our proposed method, affect the number of regions obtained. The viscous watershed transform [16] is conceptually special in the sense of simultaneously attacking the problem of over-segmentation as well as regularization. It applies a viscous closing with a circular structural element to the initial image. However, the viscous closing does not entirely solve the problem of over-segmentation, and some of the detected objects can take shapes from the structural elements being used. The approach in [4] was based on the four-colour theorem which is not easily extended into higher dimensions for arbitrary graphs. In this work we develop a versatile algorithm formally valid in an arbitrary number of dimensions.

Our proposed method to reduce over-segmentation is not dependent on pre-defined markers, but is rather based on regularization and merging of objects within the watershed algorithm. Formulating the watershed algorithm in terms of a transport equation, as described in [9], [10], allows a direct regularization of the algorithm by the introduction of a correction term in the equation. We use an anisotropic diffusion operator with parameter settings chosen such that the normal speed of the watershed front is not affected, and smoothing takes place in the direction perpendicular to predominant structures. The introduction of this anisotropic regularization term produces more regular watershed contours, which are better aligned with smooth biological structures. The resulting new algorithm has the following main features: it requires no pre-defined markers, it is automatically regularizing, and efficient in terms of accuracy.

As the model system we have chosen fluorescence microscopy, which is a particularly useful tool for high-throughput screening of cells. We demonstrate the robustness and versatility of our algorithm by applying it for segmentation of various cell types with different stainings. A whole cell segmentation where every cell is outlined represents a basis for further analysis of single cell properties in huge data sets. By using automated approaches for image acquisition and post-processing, statistically valid information can easily be acquired for single cells.

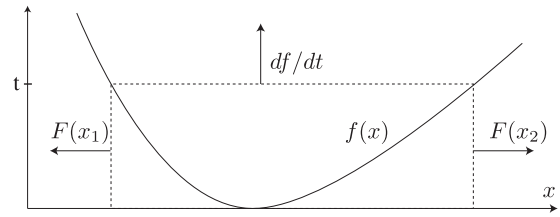

Fig. 1. The idea behind a watershed segmentation is filling of water into a basin, here represented by a function $f(x)$. There is a two-fold motion pattern. First, there is a vertical motion of the water surface (horizontal dashed line) as the basin is filled with water. Second, the moving water front (vertical dashed line), defined as the points of intersection between the water surface and the basin function $f(x)$ will move horizontally with a speed denoted by F, depending on the shape of f as well as the raising speed df/dt of the water surface.

Our main contributions in this work are the following: (i) Mathematical description and implementation of continuous watersheds in 3D, allowing for positive and negative speed functions. (ii) Integration of a regularization term. (iii) An algorithm allowing for, but not depending on pre-defined markers to deal with the problem of over-segmentation. (iv) A comprehensive evaluation of the segmentation performance as compared to manually defined ground truth images. (v) Demonstration of usefulness for the task of automated cell segmentation.

II. Methodology

A. Watershed by Immersion

Watershed by immersion is motivated by the natural phenomenon of water filling into a number of adjacent basins. In Fig. 1, we illustrate the geometry of a basin corresponding to an image function f in one dimension. Related to the stage of the "flooding" process at a time t for a given d-dimensional image function f defined on $\Omega \in \mathbb{R}^d$, we shall denote the Eulerian coordinates $x \in \mathbb{R}^d$ and the Lagrangian coordinates $X \in \mathbb{R}^d$. The Lagrangian coordinates X are fixed in space for a given particle, representing the initial position of that particle at $t = 0$. Using this formulation, a particle with initial position $x(X, t = 0) = X$ will be dislocated to position $x(X, t)$ at time point t in the "flooding" process driven by the water front.

Within watershed by immersion, an imagined water filling of the basin f takes place, where the water is "raining" into the basin. The filling of the basin with water will generate a motion pattern with two major components, as shown in Fig. 1. First, there will be a vertical motion of the water surface. Second, there will be a horizontal motion as the water front is becoming dislocated horizontally. It is the horizontal motion of the water front we use for segmentation, but the vertical positioning of the water surface is used to track the horizontal motion since the horizontal and the vertical motion are intrinsically connected.

To control the rate at which the waterlevel is rising, we define a filling function $g(t)$ independent of space, representing the height of the water surface (represented by the horizontal dashed line in Fig. 1). For our purpose we choose a filling function $g(t) = at$ for a constant $a \in \mathbb{R}^+$. This choice of a will ensure that the waterlevel at every time point t will be at level

at, implying $f(x(X,t)) = at$. We shall now derive the horizontal speed $F : \mathbb{R}^n \to \mathbb{R}^+$ of the water front. Applying the chain rule through the Eulerian coordinates $x(X,t)$, we arrive at the condition

$$\frac{df\,(x(X,t))}{dt} = \nabla f \cdot \frac{dx}{dt} = a. \tag{1}$$

The term dx/dt is the horizontal motion of the developing water front. The water front coincides with the level set of f, which has unit normal vectors given by $N := \nabla f/|\nabla f|$. Note that only the velocity component along the normal direction ∇f will contribute to motion of the water front. Thus, we get from (1) that the normal velocity F of the water front is

$$F = \frac{dx}{dt} \cdot N = \frac{dx}{dt} \cdot \frac{\nabla f}{|\nabla f|} = \frac{a}{|\nabla f|}. \tag{2}$$

The above F gives the normal speed of the front at every point x. Applying different values of the scaling factor a will affect the global rate of water filling but it will not influence the shape of the water front. Therefore, without loss of generality we choose $a = 1$, corresponding to the positively valued speed function F

$$F = \frac{1}{|\nabla f|}. \tag{3}$$

The watersheds are the dividing boundaries, points in 1D, curves in 2D and surfaces in 3D where the waterfront from different basins meet and merge with each other.

B. Watershed With Topographical Distance Functions

Watershed by *topographical distance* is based on a distance measure where the watersheds are the points of equal and minimum distance from given markers. A much used distance measure for the watershed transform is the topographical distance function $\mathcal{T} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ [20]

$$\mathcal{T}(p,q) = \inf_{\gamma} \int_{\gamma} |\nabla f| ds \tag{4}$$

defining the infimum path length between points p and q along all possible paths γ. The topographical distance between a point p and a set $A \subset \Omega$ is defined as $\mathcal{T}(p,A) = \min_{a \in A} \mathcal{T}(p,a)$. In this context, the understanding of A is a region or a point marker. Let $\{m_i\}$, $i \in I$ be a set of markers for an index set I. We assume that each m_i is a point. The catchment basin CB_i, also denoted as the segmentation around marker m_i, is defined as the points that are topographically closer to m_i than to any other marker m_j:

$$\begin{aligned} CB(m_i) = \{x \in \Omega |\ \forall j \in I \setminus \{i\} :& \\ f(m_i) + \mathcal{T}(x,m_i) < f(m_j) + \mathcal{T}(x,m_j)\}&. \end{aligned} \tag{5}$$

The watersheds are the points not belonging to any catchment basin, which are the points of equidistance from the two closest markers. Note that if f is convex in $CB(m_i)$ then we have the relation

$$f(x) = f(m_i) + \mathcal{T}(x,m_i), \quad \forall x \in CB(m_i). \tag{6}$$

This observation connects watershed by topographical distance to watershed by immersion. Equation (5) shows that the watershed defined by $CB(m_i)$ are exactly the points where water front from two nearest markers meets if the function f is convex over each $CB(m_i)$. Thus, the watershed given by topographical distance is the same as the watershed given by water immersion. The above formulations can be extended to marker regions.

In [9], it was noted that the function $U_j(x) = \mathcal{T}(x,m_j)$ is the viscosity solution of the Eikonal equation

$$|\nabla U_j| = |\nabla f| \quad \forall x \in \Omega, \quad \text{and} \quad U_j(x) = f(m_i) \quad \forall x \in m_j. \tag{7}$$

Fast implementations of this boundary value problem have been developed by the fast marching method (FMM) [9], [21]–[23]. Extended and formal definitions on continuous watersheds can be found in [24].

C. Watershed by Immersion Using a Transport Equation

We shall now present other approaches to compute the watersheds as initial value problems rather than boundary value problems. Let us consider a function $\phi : \mathbb{R}^n \times \mathbb{R} \to \mathbb{R}$ on the domain $\Omega \times [T_0, T]$. The function ϕ consists of densely packed particles where each particle can be tracked, and the function can either be expressed using Lagrangian coordinates $\phi(X,t)$, or equivalently as a function of Eulerian coordinates $\phi(x(t),t)$, implying $\phi(X,t) = \phi(x(t),t)$. Given ϕ and for a suitable constant $c > 0$, we define the time dependent domain

$$\Omega_t = \{x : \phi(x(t),t) > c\}, \quad \text{and} \quad \Gamma_t = \partial\Omega_t$$

for any time point $t \geq T_0$. From the definition, assuming that the function ϕ is smooth, the front Γ_t is moving under the condition

$$\phi\,(x(t),t) = c. \tag{8}$$

We shall have the moving front Γ_t to be the water front for water immersion, noting that (8) states that ϕ is constant on the front. The material derivative with respect to t applied to (8) results in the evolution equation [25]

$$\phi_t + \nabla\phi \cdot x_t = 0. \tag{9}$$

The term x_t represents the velocity of the moving particles within the front. However, only the normal velocity will contribute to the displacement of the front and we therefore denote the normal speed function $F \geq 0$ as the absolute component of x_t pointing to the normal direction which is parallel to $\nabla\phi$. A positive speed function results in monotonic expansion of the curve Γ_t. We also assume a smooth F. With these criteria, the initial value problem in (9) can be phrased as the transport equation

$$\phi_t + |\nabla\phi| F = 0. \tag{10}$$

where the speed F is defined in (3). Implementing this model for curve evolution is referred to as the *watershed level set* for segmentation. Condition (8) is also valid for ϕ having a discontinuity around x, since

$$\phi\left(x^+(t),t\right) = c_1 \text{ and } \phi\left(x^-(t),t\right) = c_2,\ c_1 \neq c_2$$

will lead to the same evolution equation (10) around a discontinuity in $x(t)$ as long as the speed function is smooth across the discontinuity. We also need to supply a proper initial condition $\phi(x(T_0), T_0) = \phi_0(X)$ for

$$\phi_0(X) = \begin{cases} 2c & \text{if } X \in \Omega_0, \\ 0 & \text{if } X \in \Omega \setminus \Omega_0. \end{cases} \tag{11}$$

for a given domain Ω_0. With initial values given by (11), the solution of (10) can only take values 0 or $2c$ according to (8) and the discontinuity of ϕ is propagated with the normal speed F. In the following, we shall take $\Omega_0 = \{x | f(x) < T_0\}$ for a given T_0. Then, it is easy to see that the discontinuity of ϕ is exactly the water front by water immersion starting with the region Ω_0 filled with water.

The usage of a transport equation similar to (10) was applied by Tek and Kimia (see Eq. (4) of [26]), where this type of shock inspired segmentation is referred to as reaction-diffusion bubbles. The authors also point out the labor intensive manual initialization of multiple developing fronts. We solve this problem by automatically treating every minimum in the image as a separate initialization region, which are later merged according to merging rules. Furthermore, compared to their approach, we avoid the usage of the curvature term since this will affect the speed of the evolving front differently in the convex and concave case, which can lead to segmentation errors.

Solving the Eikonal equation (7) is equivalent to the level set formulation (10) for $F \geq 0$. Both the boundary value formulation (7) and the initial value formulation (10) are easily extended to higher dimensions. However, there are at least two advantages present for the level set formulation, which are the reason for choosing (10) as our model for curve evolution instead of (7). First, a diffusion term can easily be introduced into the numerical implementation of the level set framework. Second, negative speed functions can be a useful extension with the integration of speed functions based on curve-shape. Locally, negative speed functions can thus be found to be effective to shape the watersheds according to given smoothness constraints. Related ideas were successfully explored in [11] were the concept of volume flooding was introduced as a speed function depending on the volume of the segmented region. As an alternative to measuring the absolute volume one could consider a relative volume, which could introduce negative speed functions. Negative speed functions in the sense of $F < 0$ have not been explored in this work. However, negativity by indirect means from curve contraction due to the regularization effect are accounted for and represent an important feature of our regularization.

D. Smoothing the Level Curves of Evolution

In order to smooth the level curves, we seek a regularization term which does not systematically change the normal speed of the curves since this will greatly alter the position of the watersheds. One possibility would be to add a regularization proportional to the mean curvature motion. However, such a model would slow down convex curves and accelerate concave curves, which would affect the final watershed positions for small and large objects in a different way, and thereby lead to poor segmentation.

Alternatively, we propose to apply anisotropic diffusion within the framework of coherence enhancing diffusion [27]. For this model the diffusion is directed along the tangential directions of the water front, smoothing out small oscillations. For suitable parameters this type of smoothing will not significantly change the locations of the front since it is volume preserving. Adding the anisotropic diffusion operator to the initial value problem in (10) leads to a regularized model for curve evolution, described by

$$\phi_t + |\nabla\phi| F - \alpha \nabla \cdot (D \nabla \phi) = 0,\ \phi\left(x(T_0), T_0\right) = \phi_0(X) \tag{12}$$

where D is a diffusion tensor of second order, and $\alpha \in \mathbb{R}^+$ is a scaling factor for the diffusion strength. Define the symmetric structure tensor

$$S = G_\sigma * (\nabla\phi \nabla\phi^T) \tag{13}$$

where $G_\sigma * (\cdot)$ means convolved with a Gaussian G_σ with standard deviation σ. Let us take the 3D problem as an example for our explanations. Since S is Hermitian, a diffusion tensor with the same eigenvectors as S can be defined as $D = RCR^T$ where R is the rotation matrix whose columns are the eigenvectors of the structure tensor S, and $C_{i,i} = \{c_i\}$ is a diagonal matrix. The difference to the eigenvalue decomposition of S lies in the change of eigenvalues to tunable conductivity coefficients c_i. These parameters can be used to control the diffusion along the direction of eigenvectors of the structure tensor. Considering an object in 3D, defined by a level surface of ϕ, our aim is to smooth the function ϕ along the tangential directions of the surface. Assuming sorted eigenvalues $|\lambda_1| < |\lambda_2| < |\lambda_3|$, the largest intensity variation takes place along the third eigenvector, corresponding to the direction across the most high intensity boundary. A scale invariant set of locally defined parameters reflecting local anisotropic properties can be defined by

$$b_{i,j} = \beta + (1 - \beta) e^{-\kappa / (\lambda_i - \lambda_j)^2}\ \forall i, j = \{1, 2, 3\}, i \neq j \tag{14}$$

and the conductivity values can from this be chosen as $c_1 = \max(b_{1,2}, b_{1,3}, b_{2,3})$, $c_2 = c_1$, $c_3 = \beta$ for a small and positive $\beta \ll 1$ [27]. This will ensure little diffusion across the boundary and promote a strong diffusion in the tangent plane of the boundary. More homogeneously valued conductivity coefficients will lead to a higher degree of isotropic diffusion. For the remaining, implementing the continuous model for regularized curve evolution described in (12)–(14) is referred to as *regularized watershed level set* for segmentation.

E. Strategies to Avoid Over- and Under-Segmentation Within Watershed Segmentation

"Non-marker" controlled watershed algorithms can suffer from severe over segmentation due to the large number of naturally occurring minima in real images. Each minima point will produce one region in the final watershed. This problem can be partly overcome in at least two ways: Using hierarchical watersheds [15]–[19], [28] or using predefined markers as in [2], [12]–[14]. These approaches for dealing with over and under segmentation are equivalent in terms of the mathematical understanding of rising the waterlevel or computing the topographical distance function, but they may vary practically

due to the merging process taking place in the first approach. In this paper we essentially follow the first approach and define new markers each time when a local minimum is under the waterlevel. When the water level raises, there are new regions (or points) that are immersed by the water. Whenever a new region is under the water level, we will add a new marker to make this region and start to trace the water front from these new regions as well. We will trace the water front as the water level raises and detect the places when one water front meets another one and these points are defined as the watersheds.

We trace the water front from each marker by solving the (regularized) watershed level set through (12)–(14). This implies solving different initial value problems each time a new marker is added. By this approach we obtain fronts approaching the object boundaries from two sides. However, instead of merging the objects after computing the entire watershed transform we merge the objects on the "fly". This approach simplifies the merging since we can assign an object to a parent object with a higher certainty. The merging approach is further explained in Supplementary material, Appendix B.

III. Numerical Implementation

A. Methods for Watershed Segmentation

In tests given later, we will use four methods for watershed segmentation and compare their performances. The methods we use are:

(i) *Watershed by immersion*.
(ii) *Watershed level set*.
(iii) *Regularized watershed level set*.
(iv) *The viscous watershed transform* [16].

While (i) is a traditional watershed algorithm used as a reference, (ii) and (iii) are new methods based on the formulation of watershed immersion in terms of the transport equation (ii) and the regularized transport equations (iii). Method (iv) is not new [16], and is used as reference to a state-of-the-art method. This algorithm was chosen as a state-of-the-art algorithm dealing with both over-segmentation as well as regularization. All methods (i)–(iv) use the same merging algorithm described in Supplementary material, Algorithm 2 For the normal speed F in (3) we add a small regularization parameter $\epsilon = 0.0025$ to avoid singularities by dividing with zero. Therefore we redefine the speed function as $F = 1/\sqrt{|\nabla f|^2 + \epsilon^2}$.

B. Watershed by Immersion, Method (i)

This algorithm serves as a reference method, and is closely related to the standard watershed method described in [7]. In this algorithm all voxels in the image are sorted from low to high image values. The immersion is simulated by removing voxels from the queue as the waterlevel is rising above each voxel's intensity value, and they are assigned to various parent objects. We could not use the unmodified version in [7] because that method is marker dependent in order to function properly. The reference method would thereby depend on input variables that our proposed methods are not depending on. Therefore, the reference method (i) differs from the original algorithm in [7] as we apply the merging algorithm described in Algorithm 2 instead of pre-defined input markers. Each voxel is thereby assigned to a parent object based on the merging rules described in Supplementary material, Appendix A.

C. Watershed Level Set (Method (ii)) and Regularized Watershed Level Set (Method (iii))

Watershed level set and regularized watershed level set are the implementation of (12). They only differ within the regularization parameter α, where $\alpha(x) = 0$ for watershed level set and $\alpha(x) > 0$ for regularized watershed level set. Thus, we omit the coherence enhancing diffusion operator for watershed level set. Details on initialization of markers and usage of predefined markers can be found in Supplementary material, Appendix B and C, respectively. Further algorithmic details are described in Supplementary material, Algorithm 1.

The partial differential equation is solved by operator splitting, where the transport part is discretized using upwinding and coherence enhancing diffusion by standard finite differences. Further details on discretization and numerical implementation are found in Appendixes D, E and F, in Supplementary material. Software can be downloaded from https://github.com/ehodneland/watertransport.git.

D. Method for Evaluation of Segmentation Quality

For the quantitative analysis in Section IV-B we computed three evaluation parameters, C, C_{prec}, C_{sens} and the average cell surface area $\overline{A}$. Within a set of segmented regions $\mathcal{A} = \{\mathcal{A}_i\}$ and $\mathcal{B} = \{\mathcal{B}_i\}$, a matrix of Dice coefficients between any two regions $\mathcal{A}_i$ and $\mathcal{B}_j$ was computed as

$$\tilde{C}(i,j) = \frac{2|\mathcal{A}_i \cap \mathcal{B}_j|}{|\mathcal{A}_i| + |\mathcal{B}_j|}. \tag{15}$$

The segmented regions $\mathcal{A}_i$ are disconnected components within $l_{\Omega_n}(x)$ (see Supplementary material Appendix B) at the end of water raising, and $\mathcal{B}_i$ are corresponding regions from a manual segmentation. We also computed matrices of precision and sensitivity,

$$\tilde{C}_{prec}(i,j) = \frac{|\mathcal{A}_i \cap \mathcal{B}_j|}{|\mathcal{A}_i|},\ \tilde{C}_{sens}(i,j) = \frac{|\mathcal{A}_i \cap \mathcal{B}_j|}{|\mathcal{B}_j|}. \tag{16}$$

The matrix $\tilde{C}_{prec}$ will be a measure of over-segmentation, and correspondingly the matrix $\tilde{C}_{sens}$ will be a measure of under-segmentation. Then, we used the Hungarian algorithm for an optimal matching between the evaluation scores in $\mathcal{A}$ and $\mathcal{B}$, as described in [13]. The final evaluation scores of segmentation quality are obtained by normalizing the non-zero row values of $\tilde{C}$ with (cell volume)/(sum of all cell volumes) within the image, and thereafter summing the values into weighted coefficient arrays C, C_{prec} or C_{sens}, corresponding to $\tilde{C}, \tilde{C}_{prec}, \tilde{C}_{sens}$, respectively. C we refer to as the weighted Dice coefficient, and C_{prec} and C_{sens} are precision and sensitity coefficients reflecting under- and over segmentation. The evaluation parameters have per definition a lower limit of 0 and an upper theoretical limit of 1. Values of $\{C, C_{prec}, C_{sens}\} \to 0, 1$ reflect a poor and a high-quality segmentation, respectively.

We also computed the average cell surface area for each of the algorithms. The surface area of the binarized, single cell

within $\mathcal{A}_i$ (similarly for $\mathcal{B}_i$) can be approximated using the total variation

$$A_i = \int_{\Omega} |\nabla \mathcal{A}_i(x)|\, dx, \tag{17}$$

leading to the average surface area $\overline{A}$ upon averaging over A_i.

For the benchmarking experiments of the Broad institute data sets in Section IV-C we used the recommended evaluation criteria described as "O = Outlines of object" at the Broad Bioimage Benchmark Collection in order to enable a comparison of performance with previous publications using the same data sets: 'To compare an algorithms results to the manual outlines, define the relevant boundary pixels as the pixels that are on the boundary found by the algorithm and that are not adjacent to any background pixels. For each relevant pixel, compute the Euclidean distance to the corresponding pixel on the manually found outline. Report the percentage of relevant pixels that are within two pixels of the corresponding pixel on the manually found outline.' (cited from http://www.broadinstitute.org/bbbc/index.html)

IV. Experimental Results

We performed three types of experimental tests. In Section IV-A we present a qualitative comparison of methods (i)–(iii). In Section IV-B we perform a 3D segmentation of manually annotated datasets, and compare the results to a state-of-the-art software for cell segmentation, CellProfiler [29]. In Section IV-C we apply methods (i)–(iii) and a state-of-the-art method, the viscous watershed transform (iv) [16], to benchmarking datasets from the Broad institute. For all experiments we applied $\beta = 10^{-3}$ (14) for method (iii), and the software package `Cellsegm` [18] was used for object classification of the watershed regions into cells or background.

A. Qualitative Image Analysis

For all experiments described in this section we applied methods (i)–(iii) defined in Section III-A. The example images employed for demonstrating the segmentation algorithm contain cells imaged using fluorescence microscopy. Two different cell lines were used for demonstration: (i) PC12 (pheochromocytoma 12) cells and (ii) HeLa Kyoto cells. Both cell lines were stained with a plasma membrane staining, wheat germ agglutinin Alexa Fluor 488 conjugate (WGA-AF-488). WGA-AF-488 stains the plasma membrane and is later biologically internalized. It diffuses into the cells, resulting in a bright staining of intracellular membranes, mainly vesicular. Those membranes can be recognized by the segmentation algorithm and can lead to false definitions of cell borders, and is a major reason for the need of a regularization term.

Within the examples of Figs. 2–4 we used the following parameter settings: $T_0 = 0.20$, $V_{merge} = 1500\ \mu\text{m}^3$, and (only method (iii)) $\kappa = 10^{-6}$ and $\alpha = 500$. For Figs. 2 (E-F), we instead used $V_{merge} = 100\ \mu\text{m}^3$.

The example in Fig. 2 shows two PC12 cells. The detected ridges (the plasma membrane of the cells) are outlined by the black segmentation contours surrounding the dark cell interior.

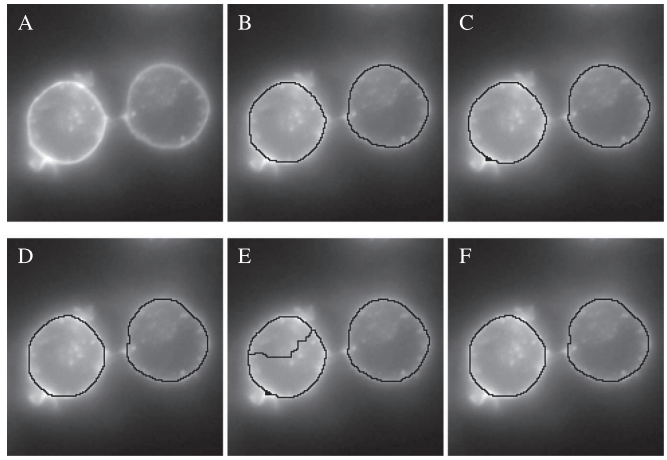

Fig. 2. Segmentation of a PC12 cell in 2D using the watershed algorithm. The dividing watershed lines are marked in black and superimposed onto the image to visualize their location. (A) Unprocessed image, (B) segmentation by watershed by immersion, (C) watershed level set, and (D) regularized watershed level set. For this cell the watershed lines are precisely located on the crest lines of the cell for all presented methods. In (E-F) we applied a lower value of V_{merge}, resulting in over-segmentation for watershed level set (E) but not for regularized watershed level set (F). This is due to the connecting properties of the regularization, affecting over-segmentation.

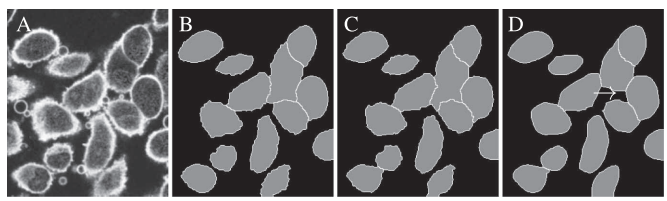

Fig. 3. Segmentation of HeLa Kyoto cells in 2D (gray regions = cells). Description as in Fig. 2 (A)–(D). These images contain a more oscillatory and inhomogeneous signal compared to Fig. 2. The arrow points to a region where the regularized watershed has performed better not only in terms of smoothness, but also in terms of improved object detection.

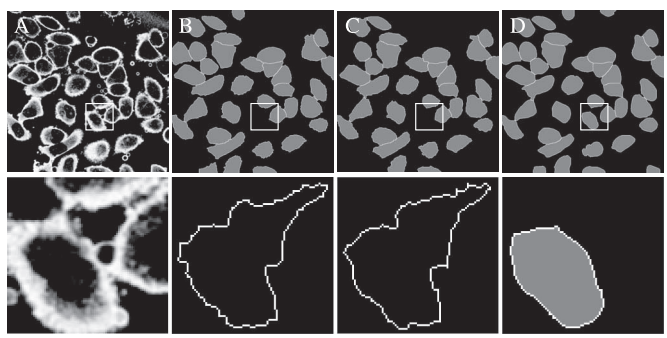

Fig. 4. Segmentation of HeLa Kyoto cells in 2D. Description as in Fig. 2, (A)–(D). The segmentation is smoother in the right panel using regularized watershed level set. Lower row: Magnification of an area (indicated by the square in the upper row) where all methods found an object within, but only method (iii) found a smooth object. As a consequence of this, the object was discarded as a cell in (B) and (C) by the cell classification algorithm assuming a certain degree of convexity. However, in (D), using regularized watershed level set, the object was smooth enough to become a cell.

The segmentation of the two cells was similar and successful for all three methods applied to this high-quality data set. This shows that in the presence of little noise and artefacts the regularized watershed level set has a similar performance to watershed level set. However, there can be differences in terms

TABLE I
COMPARATIVE EVALUATION OF SEGMENTATION QUALITY BETWEEN TWO MANUAL OBSERVERS (O_1, O_2), AND BETWEEN EACH OF THE MANUAL OBSERVERS TO THE PROPOSED METHODS (I)–(III), AND CELLPROFILER. FOR REGULARIZED WATERSHED LEVEL SET WE USED TWO DIFFERENT SETTINGS OF THE REGULARIZATION PARAMETER $\alpha_1 = 500$, $\alpha_2 = 1000$. THE EVALUATION PARAMETERS C, C_{prec}, C_{sens} ARE DESCRIBED IN Section III-D. SINCE ANY OF THE TWO OBSERVERS HAVE EQUALLY GOOD EXPERTIZE IN RECOGNIZING CELLS, WE ALSO COMPUTED ADJUSTED SCORES $C^a, C^a_{prec}, C^a_{sens}$, WHICH ARE THE RAW SCORES AFTER SUBTRACTING THE INTER-OBSERVER DISAGREEMENT ($O_1 - O_2$). THE BEST EVALUATION RESULTS COMPARED TO ANY OF THE MANUAL OBSERVERS ARE MARKED IN BOLD, AND THE BEST, ADJUSTED DICE COEFFICIENT C^a WAS ACHIEVED BY REGULARIZED WATERSHED LEVEL SET

	Raw			Adjusted		
Comparison	C	C_{prec}	C_{sens}	C^a	C^a_{prec}	C^a_{sens}
$O_1 - O_2$	0.896	0.847	0.962	1.000	1.000	1.000
O_1-method (i)	0.762	0.753	0.789	0.866	0.906	0.827
O_2-method (i)	0.749	0.786	0.731	0.853	**0.939**	0.768
O_1-method (ii)	0.751	0.753	0.772	0.856	0.907	0.810
O_2-method (ii)	0.736	0.784	0.711	0.840	0.937	0.749
O_1-method (iii), α_1	0.752	0.721	0.827	0.856	0.874	0.865
O_2-method (iii), α_1	0.758	0.772	0.778	0.862	0.925	0.816
O_1-method (iii), α_2	0.754	0.714	0.841	0.858	0.867	**0.879**
O_2-method (iii), α_2	0.765	0.769	0.796	**0.870**	0.922	0.834
O_1-Cellprofiler	0.175	0.307	0.154	0.279	0.460	0.192
O_2-Cellprofiler	0.160	0.295	0.137	0.264	0.448	0.175

of merging, highlighted in Fig. 2 (E-F). Here, we used a lower value of V_{merge}, generating an over-segmentation for (E, watershed level set) but not for (F, regularized watershed level set). This example illustrates how the regularization is able to reduce over-segmentation by connecting nearby, small objects.

The example in Fig. 3 demonstrates segmentation of HeLa Kyoto cells. The boundaries of this cell type are typically more oscillatory than those seen for the PC12 cells. Clearly, regularized watershed level set has smoother boundaries than the two other methods. The arrow points to an interesting feature of the regularization. The diffusion protocol creates a more homogeneous front of ϕ, thus affecting the order of merging. In this example, due to this effect, regularized watershed level set resulted in a smooth and geometrically improved object than was obtained by watershed level set.

Fig. 4 demonstrates another image for segmentation of HeLa Kyoto cells. Similar to Fig. 3, the boundaries are relatively oscillatory for watershed by immersion and watershed level set applied to the unprocessed image. However, this is not the case for the regularized watershed level set. Related to the oscillatory cell indicated by an arrow in Fig. 3, another example is seen in the lower row of Fig. 4. However, in this case, the highly irregular shape of the object found by watershed level set led to a incorrect rejection of this object as a cell due to lack of convexity. The settings used for cell classification by `cellsegm.classifycells` were intincell = 1.0, convexperim = 0.40, convexarea = 0.80, intborder = 1.0, minvolfull = 4.0, maxvolfull = 45.0.

B. Quantitative Image Analysis

A quantitative comparative analysis of the segmentation quality of the proposed methods was also conducted with manually annotated cells used as a gold standard. The manual segmentation was accomplished by two experts in cell biology, and are in the following denoted as O_1 and O_2. The performance of Cellprofiler [29] was also included. We used four data sets in 3D for this analysis, taken from two different experimental conditions. The parameters reflecting segmentation quality are reported in Table I for the four data sets, showing

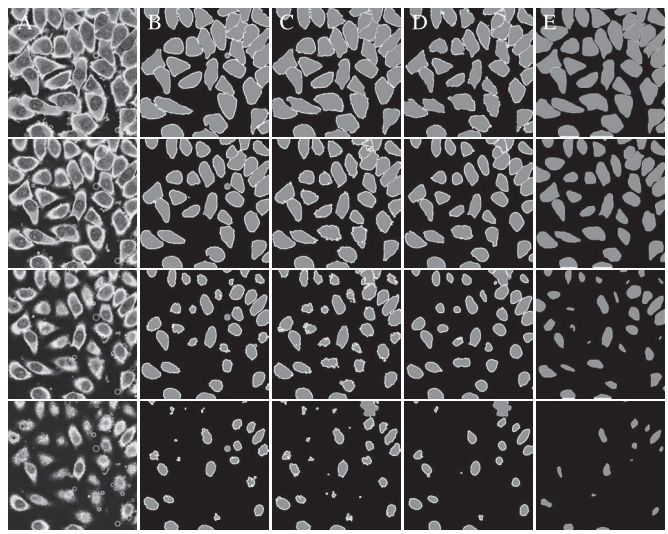

Fig. 5. Segmentation of HeLa Kyoto cells in 3D for one image stack. (A) Unprocessed image, (B) segmentation by watershed by immersion, method (i), (C) watershed level set, method (ii), (D) regularized watershed level set, method (iii), and (E) manual segmentation by O_1. Clearly, the regularized watershed level set provides a smoother segmentation than methods (i) and (ii). Top to bottom: Every second focal section from the same 3D image stack, starting with section number one. Note that the manual outline in (E) labels the interior of the cells, whereas the white lines in (B–D) highlights the plasma membrane. This might be interpreted as missing cells in the lower row.

improved volume segmentation for regularized watershed level set (method (iii)) compared to methods (i) and (ii). Segmentation results for three watershed methods are compared to manual segmentation in Fig. 5 for the first image stack among the four data sets used. The four rows in cross-sections taken at four different focal sections with respect to the z-axis. From this figure as well as Fig. 6 can see that the object boundaries within the regularized watershed level set are smoother than within the other methods. This is also quantitatively shown in Table II, reporting the average cell surface area of the segmentation, where the regularized watershed level set has a surface area being far closes to manual annotation than the other two methods. The settings used for cell classification by `cellsegm.classifycells` were intincell = 0.60,

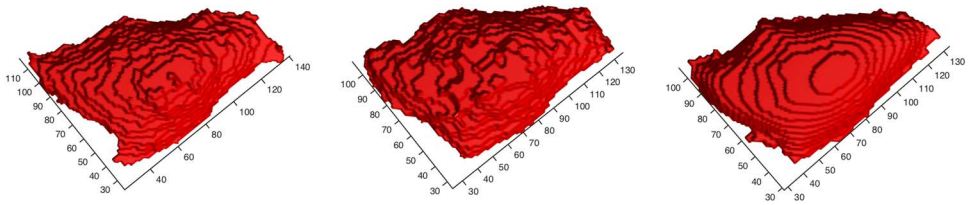

Fig. 6. Surface plots for watershed by immersion (left), watershed level set (middle) and regularized watershed level set (right) applied to a 3D data set. The surface is smoother for the regularized watershed level set.

TABLE II
AVERAGE SURFACE AREA PER CELL, $\overline{A}$. THE SURFACE AREA USING REGULARIZED WATERSHED LEVEL SET IS CLOSER TO THE MANUAL SEGMENTATION THAN THE OTHER METHODS. THIS RESULT IS DUE TO LESS OSCILLATIONS OF THE CELL BOUNDARIES USING REGULARIZED WATERSHED LEVEL SET

Observer or method	Average surface area (μm^2/cell)
O_1	5359
O_2	5018
Ws by immersion (i)	5776
Ws level set (ii)	6137
Regularized ws level set (iii) , α_1	5169
Cellprofiler	34402

$\mathrm{convexperim} = 0.40$, $\mathrm{convexarea} = 0.80$, $\mathrm{intborder} = 1.0$, $\mathrm{minvolfull} = 1.0$, $\mathrm{maxvolfull} = 28.0$.

C. Benchmarking Image Segmentation

In order to compare our methods to a state-of-the-art approach for regularized watershed segmentation we implemented the viscous watershed transform (iv), oil immersion, as described in [16]. The viscous watershed transform created strong over-segmentation and we therefore had to integrate the same merging algorithm that was applied for methods (i)–(iii). Applying ways of controlling over-segmentation is in line with [16].

We used benchmarking image set BBBC007v1 version 1 (Drosophila Kc167 cells stained for DNA and actin, using DNA stain for marker detection, actin stain for segmentation) [30], as well as BBBC018v1 (HT29 human colon cancer cell line, using Hoechst stain for marker detection, actin stain for segmentation), available from the Broad Bioimage Benchmark Collection [31]. For the data set BBBC007v1 version 1, Jones *et al.* [30] obtained a performance of 64% according to the "O = Outlines of object" evaluation criteria of the Broad Bioimage Benchmark Collection, although it is not clear from the publication whether they only used boundary pixels between adjacent cells (see criteria for evaluation in Section III-D). Parameter settings used are shown in Table III. For classification of watershed regions into cells or background we applied `cellsegm.classifycells` with three features: minimum and maximum cell volume (given as number of pixels in the Table), as well as the presence of a nucleus marker. The minimum cell volume, measured in number of pixels, was set to V_{merge}, the same parameter also used for the segmentation. Binary nucleus markers were generated using CellSegm [18] by the subroutine `cellsegm.segmct` with the option adaptive thresholding (method = 'adth') and threshold value $adth$. The "O = Outlines of object" evaluation criterion as well as parameter settings are reported in Table IV. These data sets were challenging to accurately segment for whole cells. Regularized watershed level set has the highest performance among the tested methods, and also higher than reported in [30] for image set BBBC007v1.

TABLE III
PARAMETER SETTINGS FOR THE BENCHMARKING DATA SETS FROM THE BROAD INSTITUTE USED IN Section IV-C

Image set	V_{merge}	Max. cell volume	$adth$	α (12)	κ (14)	T_0
I BBC007v1	100	$3 \cdot 10^4$	0.1	500	10^{-5}	0.01
II BBC018v1	200	$5 \cdot 10^4$	0.05	100	10^{-6}	0.02

TABLE IV
RESULTS APPLYING THE PROPOSED METHODS TO BENCHMARKING DATA SETS FROM THE BROAD INSTITUTE USED IN Section IV-C. THE NUMBERS ARE PERCENTAGE SUCCESS "O = Outlines of objects" (see Section III-D). BOLD INDICATES BEST PERFORMANCE. IMAGE SETS I AND II REFER TO Table III, lit. = literature, transf. = transform, and reg. = regularized

Image set	No. of images	Ws by immersion	Ws level set	Reg. ws level set	Viscous ws transf. [16]	Lit.
I	16	61.48	67.13	**68.50**	60.80	64 [30]
II	55	65.55	70.50	**73.57**	60.28	NA

V. DISCUSSION

The motivation for the current study was to develop improved methods for automated segmentation of ridge defined structures, as opposed to intensity based structures. As a highly useful application we have chosen the task of automated segmentation of cells, which is a powerful tool for automated single cell detection within imaging data. Such analysis can lead to significant biological insight with respect to cell functionality and behavior in response to drug treatment. Operational real-time segmentation is a challenging task mainly due to inevitable intensity variations both on the cell boundaries and inside the cells. Still, a labelling of cells, preferably as a surface staining, provides images with high-intensity signals located at the plasma membrane, thus being suitable for a segmentation of separated as well as clustered cells. We refer to this kind of objects as *ridge defined objects*. Data with these characteristics have the advantage that one can distinguish also between adjacent objects due to the high signal on boundaries. This type of single object segmentation is not easy to accomplish for objects characterized by high or low interior intensities, as the signal between adjacent objects can be more or less constant.

Watershed segmentation is adequate for ridge segmentation as it simulates water flooding by detecting segmentation contours along the ridges of the objects. However, traditional watershed segmentation has major challenges related to over-seg-

mentation and unregularized contours. The problems connected to over-segmentation can possibly be handled using a marker-controlled watershed. However, while this approach improves the segmentation, it introduces the difficulty of finding high quality markers, which can be a substantial challenge for real world images. Several attempts have been made to solve the over-segmentation either by marker-based [2], [12]–[14] or hierarchial watershed algorithms [6], [15]–[17], but the success of these algorithms is strongly data dependent, and the challenges posed by over- and under-segmentation will continuously be further debated and improved in future. As a contribution to reduce the general dependence on pre-defined markers in the cases where high-quality markers are not easily available, we have in this work developed an algorithm independent of initial marker regions. We shall not refer to our method as "marker-free" since markers are introduced into the level set function during iterations. However, our algorithm requires no prior knowledge of marker placement or number of markers. Every minimum in the image is considered a potential marker, and regions appearing from raising of the waterlevel are merged on the fly. This approach is potentially more stable than merging a high number of small regions after convergence since the overall object shape can be hard to assemble correctly. In our proposed merging approach, only a single volume parameter is used to control the merging, such that two large regions will not merge if they have a large probability of being individual cells. This merging protocol can easily be extended into more complex schemes based on a variety of classificators and descriptors.

We have also attacked the challenge of oscillatory contours within the traditional watershed segmentation. The regularization is a challenging task, partly because one must allow sharp corners at the vertex where three or more objects meet, take into account elongated protrusions often present in cells, and at the same create smooth contours. We have integrated a regularization term based on coherence enhancing diffusion, resulting in method (iii). The regularization allows smoother segmentation contours for the segmentation, such as demonstrated in Figs. 3–6. Coherence enhancing diffusion was chosen as diffusion operator since it is volume preserving, and does not change the average speed of the front. The diffusion term has the effect of smoothing the contours while preserving the general segmentation accuracy, and it can also partly combine smoothness and at the same time allow for segmentation of elongated protrusions. We also demonstrated that the regularization further than pure smoothing has an effect on over-segmentation as it is capable of joining nearby, small objects during iterations (Fig. 2 (E-F)). Moreover, the regularization will affect the smoothness of the propagating front, which will lead to a different merging order (Figs. 3 and 4). Both these effects can have a positive outcome on the segmentation.

It appears from our data that method (iii) handles noisy data better than the other methods without the need for preprocessing filters. Our method thus combines segmentation and regularization, instead of traditional, sequential processing, with detection of markers, filtering and segmentation. In this respect, the proposed method represents an integrated system combining all these processing steps into one. This property can have several advantages with respect to both computational time and also computational complexity and quality assessment, since a multi-stage system can be challenging to debug. The improved smoothness of the plasma membrane as seen in regularized watershed level set can represent a major improvement for the task of cell feature extraction on the plasma membrane [32]. A slightly misplaced contour will lead to significant errors when sampling a signal on the thin watershed surface. As a further consequence of smooth contours, the proposed method will have improved performance for the task of extracting average surface area of the cells.

The connection between a watershed segmentation phrased as the stationary Eikonal equation or a time-dependent transport equation was previously described mathematically [9]–[11], [21] and also experimentally demonstrated for the Eikonal equation [10], [11]. We have in this publication mathematically further elaborated on watershed immersion by a time-dependent transport equation. Furthermore, we present experimental results from 3D watershed immersion as a transport equation. In particular, the transport equation allows for positive and negative speed functions, which in terms of flexibility advocates this approach compared to the stationary Eikonal equation for immersion.

The proposed methods were compared to the performance of CellProfiler, the viscous watershed transform, as well as literature reports on benchmarking data sets. All comparison was based on manually annotated data sets. Regarding the Dice coefficients, the regularized watershed level set had slightly improved performance compared to watershed by immersion and watershed level set, and also compared to the performance of CellProfiler. Dice coefficients reflect the accuracy of the overall segmentation volume (cfr. Table I), and the obtained results are therefore not surprising since the regularization mainly modifies a oscillatory surface into a smooth surface, which will contribute little to overall volume estimates of the segmentation. More essential is the improved surface smoothness obtained by the regularized watershed level set compared to the other methods (cfr. Table II). Thus, we claim that the regularized watershed level set is the method of choice among the investigated methods. The results also revealed no major differences in performance between two different settings of the regularization parameter α.

Further, comparison of methods (i)–(iii) with the viscous watershed transform applied to benchmarking data sets of the Broad institute revealed that the best performance was achieved by the regularized watershed level set. Also compared to [30] we achieved improved segmentation results.

The performance of our proposed watershed segmentation by immersion was demonstrated both with and without the inclusion of a regularization term. The included regularization term can be considered as one particularly useful idea on how the proposed PDE framework allows for easy integration of additional constraints affecting the segmentation in a desired direction. Further extensions can be shape priors, and differentially and locally varying regularization terms, possibly inferring negative speed functions. However, implementation of these extensions is outside the scope of the current work.

Despite promising results of our method we suggest regularization terms in general must be used with care since not all objects are expected to be smooth. Many real-world images de-

pict non-smooth objects, and other types of regularization techniques can be better suited. This is the reason why we have chosen to focus on a strictly defined application, segmentation of biological structures (cells), to suggest an important field of application where our method could make an impact. Still, the mathematical formulation and algorithmic implementation are versatile and can have general implications, even outside the field of image segmentation.

In conclusion we have proposed a PDE based watershed level-set method with simultaneous regularization of the watershed contours. Our method requires no pre-defined markers, but can also be used with pre-defined markers for applications where high-quality markers exist. Our algorithm provides accurate segmentations, regularization and controls over-segmentation within the same numerical framework. The method was evaluated for the task of cell segmentation of fluorescently labeled cells, and was found to perform better in terms of Dice coeffients, surface smoothness and boundary overlap ("O") than a traditional watershed immersion method without regularization, and also compared to state-of-the art segmentation methods.

ACKNOWLEDGMENT

The authors want to thank Prof. Hans-Hermann Gerdes (*deceased*), University of Bergen, for providing image data, as well as Tanja Kögel and Dominik Frei for manually outlining cells.

REFERENCES

[1] S. Chien, Y. Huang, and L. Chen, "Predictive watershed: A fast watershed algorithm for video segmentation," *CirSysVideo*, vol. 13, no. 5, pp. 453–461, May 2003.

[2] V. Grau, A. J. U. Mewes, M. A. Raya, R. Kikinis, and S. K. Warfield, "Improved watershed transform for medical image segmentation using prior information.," *IEEE Trans. Med. Imag.* vol. 23, no. 4, pp. 447–458, Apr. 2004.

[3] N. Malpica *et al.*, "Applying watershed algorithms to the segmentation of clustered nuclei," *Cytometry*, vol. 28, pp. 289–297, 1997.

[4] E. Hodneland, X.-C. Tai, and H.-H. Gerdes, "Four-color theorem and level set methods for watershed segmentation," *Int. J. Comput. Vis.*, vol. 82, no. 3, pp. 264–283, 2009.

[5] F. Meyer, "Topographic distance and watershed lines," *Signal Process.*, vol. 38, no. 1, pp. 113–125, 1994.

[6] H. Nguyen, M. Worring, and R. van den Boomgaard, "Watersnakes: Energy-driven watershed segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.* vol. 25, no. 3, pp. 330–342, Mar. 2003.

[7] L. Vincent and P. Soille, "Watersheds in digital spaces: An efficient algorithm based on immersion simulations," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 6, pp. 583–598, Jun. 1991.

[8] C. Rambabu and I. Chakrabarti, "An efficient immersion-based watershed transform method and its prototype architecture," *J. Syst. Archit.*, vol. 53, no. 4, pp. 210–226, 2007.

[9] L. D. Cohen and T. Deschamps, "Segmentation of 3D tubular objects with adaptive front propagation and minimal tree extraction for 3D medical imaging.," *Comput. Methods Biomech. Biomed. Eng.* vol. 10, no. 4, pp. 289–305, Aug. 2007.

[10] F. Meyer and P. Maragos, "Multiscale morphological segmentations based on watershed, flooding, eikonal PDE," in *Scale-Space*, ser. Lecture Notes in Computer Science, M. Nielsen, P. Johansen, O. F. Olsen, and J. Weickert, Eds. New York: Springer, 1999, vol. 1682, pp. 351–362.

[11] A. Sofou and P. Maragos, "Generalized flooding and multicue PDE-based image segmentation," *IEEE Trans. Image Process.* vol. 17, no. 3, p. 364376, Mar. 2008.

[12] P. Felkel, M. Bruckschwaiger, and R. Wegenkittl, "Implementation and complexity of the watershed-from-markers algorithm computed as a minimal cost forest," *Comput. Graph. Forum* vol. 20, p. 2001, 2002.

[13] E. Hodneland *et al.*, "A unified framework for automated 3-D segmentation of surface-stained living cells and a comprehensive segmentation evaluation.," *IEEE Trans. Med. Imag.* vol. 28, no. 5, pp. 720–738, May 2009.

[14] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis, and Machine Vision*. New York: Thomson-Engineering, 2007.

[15] S. Beucher and F. Meyer, "The morphological approach to segmentation: The watershed transformation," in *Mathematical Morphology in Image Processing*. New York: Marcel Dekker, 1993, pp. 433–481.

[16] C. Vachier and F. Meyer, "The viscous watershed transform," *J. Math. Imag. Vis.*, vol. 22, no. 2–3, pp. 251–267, 2005.

[17] B. Marcotegui and S. Beucher, "Fast implementation of waterfall based on graphs," in *Mathematical Morphology: 40 Years On*. New York: Springer, 2005, pp. 177–186.

[18] E. Hodneland, T. Kgel, D. M. Frei, H.-H. Gerdes, and A. Lundervold, "CellSegm—A MATLAB toolbox for high-throughput 3D cell segmentation," *Source Code Biol. Med.*, vol. 8, no. 16, pp. 1–24, 2013.

[19] A. C. Jalba and M. Westenberg *et al.*, "Interactive segmentation and visualization of DTI data using a hierarchical watershed representation," *IEEE Trans. Image Process.*, vol. 24, no. 3, pp. 1025–1035, Mar. 2015.

[20] J. Roerdink and A. Meijster, The watershed transform: Definitions, algorithms and parallelization techniques Inst. Math. Comput. Sci., Univ. Groningen, Groningen, The Netherlands, 2001, vol. 41, pp. 187–228.

[21] J. A. Sethian, *Level set methods and fast marching methods*, ser. Cambridge Monographs on Applied and Computational Mathematics, 2nd ed. Cambridge, U.K.: Cambridge Univ. Press, 1999, vol. 3.

[22] R. Kimmel and J. A. Sethian, "Optimal algorithm for shape from shading and path planning," *J. Math. Imag. Vis.* vol. 14, no. 3, pp. 237–244, May 2001.

[23] J. N. Tsitsiklis, "Efficient algorithms for globally optimal trajectories," *IEEE Trans. Automatic Control*, vol. 40, no. 9, pp. 1528–1538, Sep. 1995.

[24] L. Najman and M. Schmitt, "Watershed of a continuous function," *Signal Process.*, vol. 38, no. 1, pp. 99–112, 1994.

[25] R. Malladi, J. A. Sethian, and B. C. Vemuri, "Shape modeling with front propagation: A level set approach," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 2, pp. 158–175, Feb. 1995.

[26] H. Tek and B. B. Kimia, "Image segmentation by reaction-diffusion bubbles," in *Proc. IEEE 5th Int. Conf. Comput. Vis.*, 1995, pp. 156–162.

[27] J. Weickert, "Coherence-enhancing diffusion filtering," *Int. J. Comput. Vis.*, vol. 31, no. 2–3, pp. 111–127, 1999.

[28] F. Meyer and S. Beucher, "Morphological segmentation," *J. Vis. Commun. Image Rep.*, vol. 11, pp. 21–45, 1990.

[29] A. E. Carpenter *et al.*, "Cellprofiler: Image analysis software for identifying and quantifying cell phenotypes," *Genome Biol.*, vol. 7, pp. 1–11, 2006.

[30] T. R. Jones, A. Carpenter, and P. Golland, "Voronoi-based segmentation of cells on image manifolds," in *Computer Vision for Biomedical Image Applications*. New York: Springer, 2005, pp. 535–543.

[31] V. Ljosa, K. L. Sokolnicki, and A. E. Carpenter, "Annotated high-throughput microscopy image sets for validation," *Nat. Methods*, vol. 9, no. 7, p. 637, 2012.

[32] N. V. Bukoreshtliev *et al.*, "Partitioning and exocytosis of secretory granules during division of PC12 cells," *Int. J. Cell Biol.*, vol. 2012, pp. 1–14, 2012.

[33] T. Lu, P. Neittaanmäki, and X.-C. Tai, "A parallel splitting up method and its application to Navier-Stokes equations," *Appl. Math. Lett.* vol. 4, no. 2, p. 2529, 1991.

[34] T. Lu, P. Neittaanmaki, and X.-C. Tai, "A parallel splitting-up method for partial differential equations and its applications to Navier-Stokes equations," *ESAIM: Math. Model. Numerical Analysis*, vol. 26, no. 673–708, 1992.

[35] S. Descombes and M. Thalhammer, "The Lie-Trotter splitting for nonlinear evolutionary problems with critical parameters: A compact local error representation and application to nonlinear Schrodinger equations in the semiclassical regime," *IMA J. Numerical Analysis* vol. 33, no. 2, p. 722745, Oct. 2012.

[36] G. Strang, "On the construction and comparison of difference schemes," *SIAM J. Numerical Analysis*, vol. 5, no. 3, pp. 506–517, 1968.

[37] O. K. Smith, "Eigenvalues of a symmetric 3×3 matrix," *Commun. ACM*, vol. 4, no. 4, 1961.

[38] D. Peng, B. Merriman, S. Osher, H. Zhao, and M. Kang, "A PDE-based fast local level set method," *J. Comput. Phys.*, vol. 155, no. 2, pp. 410–438, 1999.