Segmentation of whole cells and cell nuclei from 3-D optical microscope images using dynamic programming

doi: 10.1109/TMI.2007.913135

Segmentation of Whole Cells and Cell Nuclei From 3-D Optical Microscope Images Using Dynamic Programming

Dean P. McCullough, Prabhakar R. Gudla, *Member, IEEE*, Bradley S. Harris, Jason A. Collins, Karen J. Meaburn, Masa-Aki Nakaya, Terry P. Yamaguchi, Tom Misteli, and Stephen J. Lockett*

***Abstract*—Communications between cells in large part drive tissue development and function, as well as disease-related processes such as tumorigenesis. Understanding the mechanistic bases of these processes necessitates quantifying specific molecules in adjacent cells or cell nuclei of intact tissue. However, a major restriction on such analyses is the lack of an efficient method that correctly segments each object (cell or nucleus) from 3-D images of an intact tissue specimen. We report a highly reliable and accurate semi-automatic algorithmic method for segmenting fluorescence-labeled cells or nuclei from 3-D tissue images. Segmentation begins with semi-automatic, 2-D object delineation in a user-selected plane, using dynamic programming (DP) to locate the border with an accumulated intensity per unit length greater that any other possible border around the same object. Then the two surfaces of the object in planes above and below the selected plane are found using an algorithm that combines DP and combinatorial searching. Following segmentation, any perceived errors can be interactively corrected. Segmentation accuracy is not significantly affected by intermittent labeling of object surfaces, diffuse surfaces, or spurious signals away from surfaces. The unique strength of the segmentation method was demonstrated on a variety of biological tissue samples where all cells, including irregularly shaped cells, were accurately segmented based on visual inspection.**

***Index Terms*—Confocal microscopy, dynamic programming, segmentation, three-dimensional image analysis.**

Manuscript received June 15, 2007; revised November 5, 2007. This project was funded in whole or in part with federal funds from the National Cancer Institute (NCI), National Institutes of Health under Contract N01-CO-12400. The content of this publication does not necessarily reflect the views or policies of the Department of Health and Human Services and nor does mention of trade names, commercial products or organizations imply endorsement by the U.S. Government. NCI-Frederick is accredited by AAALAC International and follows the Public Health Service Policy for the Care and Use of Laboratory Animals. Animal care was provided in accordance with the procedures outlined in the "Guide for Care and Use of Laboratory Animals" (National Research Council; 1996; National Academy Press, Washington, D.C.). This research was supported [in part] by the Intramural Research Program of the NIH, National Cancer Institute, Center for Cancer Research. *Asterisk indicates corresponding author.*

D. P. McCullough is with High Performance Technologies, Inc., Reston, VA 20190 USA (e-mail: dmccullough@hpti.com).

P. R. Gudla and J. A. Collins are with the Image Analysis Laboratory, Advanced Technology Program, SAIC—Frederick, National Cancer Institute, Frederick, MD 21702 USA (e-mail: reddyg@ncifcrf.gov; jas.a.collins@verizon.net).

B. S. Harris was with the Image Analysis Laboratory, Advanced Technology Program, SAIC—Frederick, National Cancer Institute, Frederick, MD 21702 USA. He is now with Carl Zeiss, Inc., Thornwood, NY 10594 USA (e-mail: bharris@zeiss.com).

K. J. Meaburn and T. Misteli are with the Cell Biology of Genomes Group, National Cancer Institute, Bethesda, MD 20892 USA (e-mail: meaburnk@mail.nih.gov; mistelit@mail.nih.gov).

M. Nakaya was with the Cancer and Developmental Biology Laboratory, National Cancer Institute, Frederick, MD 21702 USA. He is now with the Department of Histology and Embryology, Graduate School of Medical Science, Kanazawa University 13-1, Takara-machi, Kanazawa 920-8640, Japan (e-mail: mnakaya@basic.m.kanazawa-u.ac.jp).

T. Yamaguchi is with the Cancer and Developmental Biology Laboratory, National Cancer Institute, Frederick, MD 21702 USA (e-mail: tyamaguchi@ncifcrf.gov).

*S. J. Lockett is with the Image Analysis Laboratory, Advanced Technology Program, SAIC—Frederick, National Cancer Institute, P.O. Box B, Frederick, MD 21702 USA (e-mail: slockett@ncifcrf.gov).

Digital Object Identifier 10.1109/TMI.2007.913135

I. Introduction

COMMUNICATION between cells in multicellular organisms is fundamental to the development and homeostasis of organisms and tissues within an organism. Furthermore, altered cell-to-cell communication underlies many disease-related processes, such as tumorigenesis. Attempts to understand the nature of such communications and their downstream effects on cell behavior are widespread, and are often performed using optical microscopy of samples fluorescence labeled for specific molecules of interest. However, interpretation of microscope images, which is done predominantly by qualitative visual examination, is limited. Quantitative analysis of 3-D microscopy images on the other hand provides a much richer source of information leading to much more in depth understandings of cell–cell communication processes and potentially leading to improved diagnosis and treatment of major human diseases, such as cancer. The information derived from quantitative analysis for example can serve as input for testing hypothesized mathematical models of complex cell–cell processes from which fundamental kinetic parameters of underlying molecular interactions can be determined. While recently there have been several reports of mathematical modeling of cell communication processes, they have been restricted to theoretical or idealized representations of the actual cells in the tissue [1]. We believe that this restriction is in large part due to the lack of efficient and accurate computational algorithms for segmenting (delineating) individual cells from within tissue samples. Following segmentation, it is possible to quantitatively measure features relating to cell morphology, spatial organization of cells, and the distribution of specific molecules inside and on individual cells.

Early work on cell segmentation concentrated on the automatic detection of cell nuclei in 2-D images from the standpoint of automating and improving disease diagnosis of cyto-

logical and histological samples. Significant interest in the 3-D segmentation of cells emerged with the advent of confocal, fluorescence microscopy approximately 15 years ago, which enabled researchers to directly acquire 3-D images of biological samples. It was quickly realized that computational 3-D analysis was essential for the accurate, individual and contextual quantification of cells.

Concurrent with the introduction of confocal microscopy, several advanced image analysis approaches were being developed for other applications that could potentially improve the accuracy of the segmentation of cells in culture and tissue samples. Dynamic programming (DP) was one of the first of these advanced segmentation approaches, which has the particular advantage of finding the globally optimal path between two points in 2-D space based on an *a priori* figure-of-merit and with some modest constraints on the curvature of the path. The global search makes the method extremely robust against noise, enabling it to correctly identify paths that are significantly corrupted by noise, missing segments, and low resolution. In this sense, DP is similar to the geodesic active contour method that minimizes a path integral with an implicit border constraint [2]. DP was applied early on to optimally detect curves in 2-D noisy pictures [3], and its first utilization in cell segmentation was for automatically determining the optimum dividing paths between clustered cell nuclei in 2-D images [4], [5]. Since then DP has been used for a variety of other applications, including finding networks of lines in 2-D images from which neuritis could be traced and cells segmented in heart tissue [6], [7], tracking individual fluorescence particles in time-lapse microscopy [8], and tracking living cells in 4-D time-lapse images [9]. Recently we have developed a semi-automated DP-based segmentation method for detecting whole cells or cell nuclei in tissue [10], and an algorithm based on 2-D DP for segmenting 3-D electron tomogram images of HIV particles was reported by Bartesaghi *et al.* [11]. A 3-D version was developed for delineating surfaces of cells in 3-D images [12], but it had not been formally proven that it finds the globally optimal solution, neither had the method been demonstrated in practical applications involving cells in tissue.

In parallel with the utilization of DP based techniques, several other advanced image analysis approaches have been adopted for cell segmentation. Gradient-curvature driven flow and level sets and related approaches became widely known in the 1990s [13], [14]. They can identify edges of objects in significantly degraded images and have the major advantage of extending to three or more dimensions. Consequently, several applications for fluorescence microscope images based on this approach have emerged. One of the first was segmentation of whole cells and cell nuclei in tissue sections [15]. Since then the approach has been used to track the dynamics of living cells [16], and recently Appleton and Talbot reported a 3-D segmentation technique based on finding globally minimal surfaces by continuous maximal flows [17]. Other promising approaches for segmentation of cells in microscope images include utilization of the watershed algorithm [18]–[21], the "active snake" method [22], statistical partitioning of images into objects [23]–[27], the Markov chain Monte Carlo method [28], Markov random fields [29], [30], and graph-cut methods [31].

Although a variety of promising approaches for segmenting cells in fluorescence microscope images exist, there is not yet a segmentation method that is highly reliable and accurate across the wide range of 3-D cell images encountered in biomedicine, as well as practical to use. Here, we report a highly reliable and accurate semi-automatic algorithmic method for segmenting 3-D images of individual whole cells and cell nuclei (objects) from inside tissue. The 3-D images must be acquired with high-resolution optical microscopy and the surfaces of the objects must be fluorescence labeled, or in the case of nuclei their volumes can be labeled instead. When considering the design of our algorithm, we accepted the "gold-standard" for correct segmentation as being the human visual system. In order to incorporate this gold-standard into the segmentation, the procedure must either permit retrospective visual verification and interactive correction of an automatic result, or alternatively incorporate prospective guidance by the human in the segmentation procedure for each object. The retrospective approach is tedious because it requires the user to search through all segmented objects to find and correct errors. Instead, we have chosen the prospective approach as being potentially more accurate and less demanding on the user, because it enables the algorithm to direct the user to each object being segmented one at a time. A similar reasoning had been used earlier in the design of an algorithm for neurite tracing [7] Furthermore, it means that erroneous segmentations are detected and corrected earlier in the process, which if left uncorrected could precipitate erroneous segmentations of subsequent objects. Additionally, it allows the segmentation of visually unsatisfactory objects to be trivially avoided. The other major design criterion for our method was that it should work on very noisy images that could arise from using low levels of fluorescent dyes and low excitation light intensity in order to minimize damage to living cells. We chose dynamic programming as the basis of our segmentation method, since it finds the border by globally maximizing an intuitive figure-of-merit, and, therefore, is extremely insensitive to noise. This was recently demonstrated to be the case by us when segmenting 2-D images [10] and earlier by Meijering *et al.* for neurite tracing [7].

The paper is organized as follows. The major part of the paper, Section II, outlines and illustrates the main steps involved in segmentation, and includes a detailed description and justification for each step. The latter part of this section describes the generation of computer simulations for assessing the performance of the segmentation approach, followed by a description of the preparation and image acquisition of cell and tissue samples used for testing the segmentation procedure. Section III presents the accuracy of the segmentation using simulated objects with noise levels significantly greater than expected in confocal microscope images and with objects closely juxtaposed to each other in order to rigorously test our method under realistic tissue-like conditions. This is followed by segmentation results from biological samples. The paper ends with a discussion of future enhancements to the segmentation approach and concludes with ideas of biological questions that could be answered following the segmentation of individual cells or cell nuclei in their tissue context.

II. Materials and Methods

A. Overview and Operation of the Segmentation Algorithm

The goal of the segmentation algorithm is to find the optimal surface around each selected object in the image. This section starts with a description of the operation of the algorithm, which is summarized in a flow diagram in Fig. 1(a). This is followed by detailed explanations of each step of the algorithm.

The input to the algorithm is a 3-D image of a biological sample where the borders of objects of interest have higher intensity than other features in the image. Typically, the objects will be whole cells or cell nuclei in intact tissue. Contrast is achieved either by fluorescence labeling the surfaces of the objects of interest [Fig. 1(b)], or by labeling the object's volume so that changes in intensity across the object's surface are high [Fig. 2(g)]. In the latter case, the image is converted to an equivalent image of surface labeling by calculating the directional gradient at every voxel [Fig. 2(h)] (see Section II-B-5 for details). Given these labeling schemes where the surface is at a higher intensity than elsewhere, the optimal surface enclosing an object is defined as having an accumulated intensity per unit surface area greater than all other surfaces.

Segmentation commences with the user visually selecting a planar x–y slice through the object and interactively marking a point inside (P_C) and a point on the border (P_B) of the object [Fig. 1(a) box 1 and Fig. 1(b)]. Next, a subvolume of the image centered on P_C is spherically transformed (Fig. 1(a) box 2) and then DP automatically calculates a border around the object in the selected x–y slice that surrounds P_C and passes through P_B (Fig. 1(a) box 3). Thus, P_B is guaranteed to be on the border. This border has an accumulated intensity per unit length greater than any other border surrounding P_C and intersecting P_B [Fig. 1(c)] [10]. The user visually verifies this 2-D segmentation (Fig. 1(a) box 4) and can add control points for correction, for example, when the border encompasses two objects [Fig. 1(a) box 5, Fig. 1(d) and (e)]. Following satisfactory 2-D segmentation of an object in the selected x–y slice, the complete surface of the object in the 3-D image that includes the segmented 2-D border is automatically found using DP in conjunction with combinatorial searching (Fig. 1(a) boxes 6–8). The user visually inspects this surface (Fig. 1(a) box 9) either in x–y slice images (Fig. 1(f) left), or in planar images at different angles of elevation about the internal point P_C (Fig. 1(f) right), and may correct the surface by adding control points in any displayed image (Fig. 1(a) box 10). The above steps are repeated for segmenting subsequent objects (Fig. 1(a) box 11). Fig. 1(f) shows the graphical user interface.

The algorithm was implemented in C and GTK-2.0 libraries were used for the graphical user interface. The source code is available from the first or last author under a licence from the National Cancer Institute, USA. The application was compiled using GNU C Compiler (GCC, version 3.4.x) and all assessments and segmentation results described in the following sections were carried out on a SGI-Altix platform. Computation time is a few seconds per object with the rate limiting step being the careful visual inspection, resulting in times of 15–60 s per object. However, the total time for segmenting all objects is still small compared to the overall time needed for undertaking a great many biological studies in their entirety.

B. Details of the Segmentation

1) Spherical Transformation: The automatic calculation of the 2-D border and 3-D surface surrounding an object is performed in a spherically-transformed subvolume of the image where the origin (O) is set to the user marked internal point P_C (Fig. 1(a) box 2). Spherical transformation simplifies the calculation of 2-D borders and 3-D surfaces enclosing objects, and allows segmentation of objects with moderately-sized concavities. We represent the spherically transformed subvolume by $S(\ominus, \phi, r)$, where $0 \leq \ominus < 2\pi$ is the azimuth, $0 \leq \phi \leq \pi$ is the elevation angle (also known as the "polar angle"), and r is the radius [Fig. 2(a)]. In the digitized, spherically-transformed 3-D space [Fig. 2(b)], ϕ is uniformly divided into 128 elements and the voxel size in the radial dimension approximately equals the size in the x/y dimensions of the Cartesian image. (The size of voxels in the x and y dimensions are normally equal for confocal microscope images, but the voxel size in the z dimension is greater.) For $|\phi - 64| < 42$, the number of elements assigned to $\ominus$ is 256, but elsewhere the number of elements decreases as $|\phi - 64|$ increases in order to avoid excessive over-sampling as ϕ approaches 0 or 128. This spherical transformation results in a total of 24 946 voxels in each $(\ominus, \phi)$ plane. The number of elements assigned to $\ominus$ and ϕ could be readily changed. Voxel intensities in the spherical coordinate system are linearly interpolated from the closest eight voxels in the Cartesian image and are only calculated in the range $\alpha D < r < \beta D$, where D is the distance between P_C and the border point P_B. α was set to 0.3 and prevents the surface from being erroneously attracted to the internal point P_C by a few high intensity voxels close to P_C. β was set to 4.0 and limits the extent of the 3-D search space. In addition, any differences in the size of voxels in the z direction versus the x and y directions in the Cartesian image (aspect ratio) was accounted for during the spherical transformation.

2) Two-Dimensional Dynamic Programming Algorithm: Although we have published the 2-D DP algorithm for finding the border around an object in an x–y plane [10], we briefly explain it here for completeness (Fig. 1(a) box 3). The spherical coordinate transformation converts a looped border around the object in a specific x–y slice into a pair of nonstraight paths in (r, ϕ) planes at $\ominus = 0, 256$, and 128 [red paths in Fig. 2(b)]. We define the globally optimal path as having the highest accumulated intensity along its length compared to all other paths, subject to the constraints that the radial coordinate can at most change by one voxel between adjacent voxels in the ϕ direction (connectivity constraint) and that there is exactly one voxel on the path at each ϕ. Thus, the curve does not double back on itself. Mathematically, the optimal path is expressed as $\max_{\{p\}}(\sum_{\varphi} I(\varphi, r_p(\varphi)))$, where $r_p(\varphi)$ is the radial coordinate of the voxel on the path at ϕ and $I(\phi, r)$ is the voxel intensity at (ϕ, r). The connectivity constraint is $|r_p(\phi - 1) - r_p(\phi)| \leq 1$, where $\phi - 1$ is a notational shortcut for one step back in the digitized version of ϕ. DP is a recursive algorithm for determining the optimal path. It is based on the observation that the best partial path to $(\phi', r_p(\phi'))$ that passes through $(\phi' - 1, r_p(\phi' - 1))$ contains (as a subpath) the best

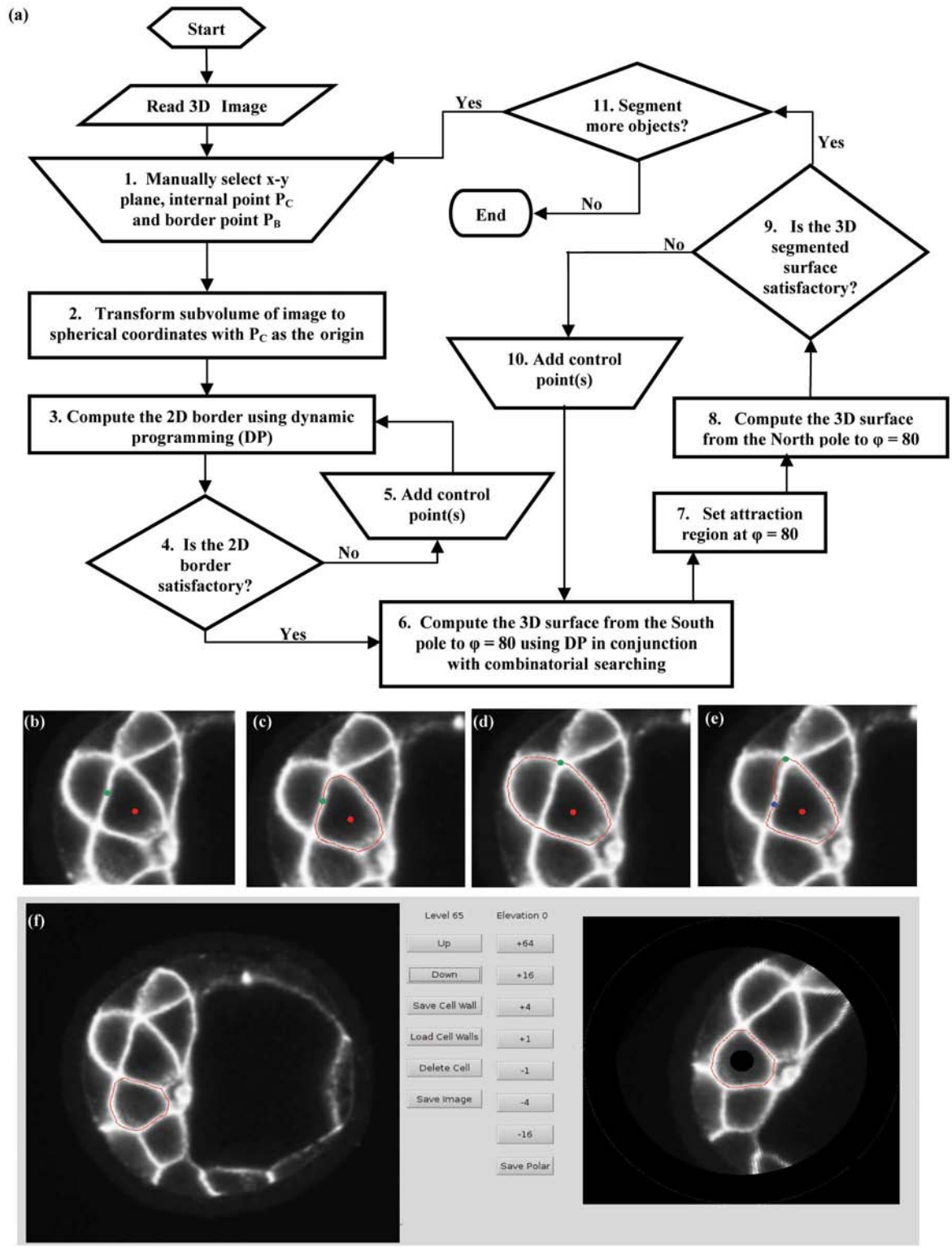

Fig. 1. Overview of the segmentation algorithm. (a) Flow diagram of the 3-D DP based segmentation procedure. (b)–(e) Segmentation of an object in a 2-D x–y slice. (b) First two control points are interactively marked inside the object (P_C, red dot) and on the border (P_B, green dot). (c) Two-dimensional border, red line, calculated by DP. (d) An incorrect border enclosing two objects and (e) interactive correction by adding a control point (blue dot). (f) Graphical user interface showing the buttons for controlling the displayed images, saving results, loading previous results, and deleting the segmented surface of a cell. "Up" and "down" change the xy-slice in the left image. The right column changes the angle of the elevation displayed in the right image.

partial path to $(\phi' - 1, r_p(\phi' - 1))$ [Fig. 2(c)]. The calculated path however may not agree with visual inspection [Fig. 1(d)], in which case control points can be interactively added that are forced to be included in the path (Fig. 1(a) boxes 4 and 5).

3) Three-Dimensional Algorithm Combining Dynamic Programming and Combinatorial Searching: The optimal surface of the 3-D object is defined as having the highest accumulated intensity per unit area compared to any other

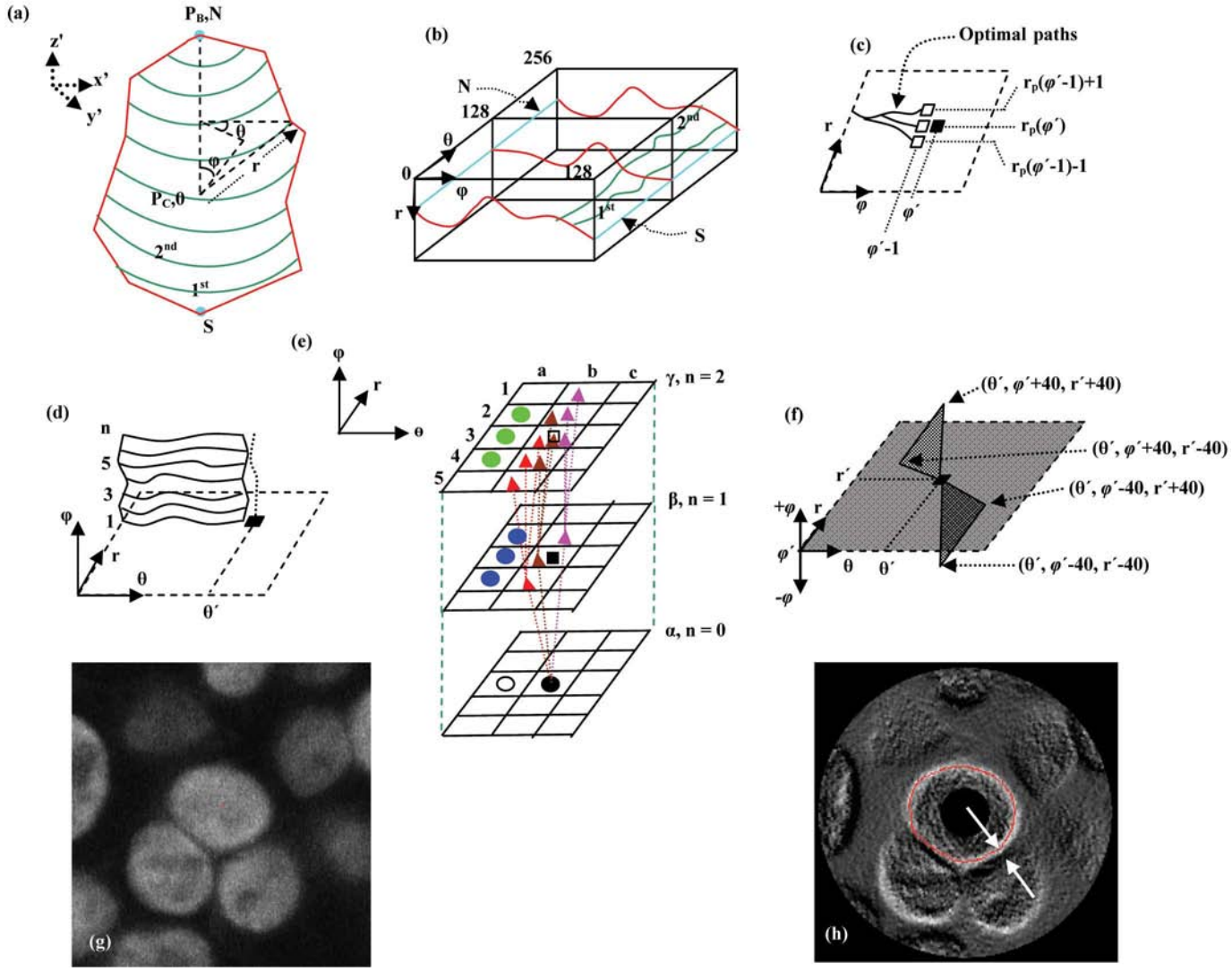

Fig. 2. The dynamic programming-based segmentation method. (a) Schematic of an object in Cartesian coordinates showing the 2-D border (red) and the 3-D surface (green), with the definition of the spherical coordinate system used in this study. x', y,' and z' are the dimensions of the Cartesian system after translation and rotation so that the internal point P_C is the origin and the border point P_B is on the positive z' axis. We also call P_B the "north pole" (N) and the "south pole" (S) is the intersection of the 2-D border and the negative z' axis. $\ominus$, ϕ and r are the azimuth, angle of elevation, and radius in the spherical transform system. The green curves represent the surface of the segmented object. (The curves around the back of the object ($\theta = \pi$ to 2π) are not shown.) (b) The object in digitized spherical coordinates. $N$and S convert to straight lines parallel to the θ axis (blue) and the 2-D border converts to two paths in (r, ϕ) planes at $\theta = 0, 256$ and at $\theta = 128$ (red). The 3-D surface is calculated as a sequence of paths (green lines). Each path, which is at a fixed ϕ, connects to the 2-D border at $\theta = 0$ and $\theta = 128$. (c) Two-dimensional DP is used for finding the optimal path in the (r, ϕ) plane. The partial solution at the solid square $(\phi', r_p(\phi'))$, equals the sum of the intensity at $(\phi', r_p(\phi'))$, $I(\phi', r_p(\phi'))$ and the maximum of the partial solutions amongst the three connected predecessors (open squares). (d) In 3-D, the goal is to calculate an optimal surface of height $\phi = n$ using DP in conjunction with combinatorial searching. For a given point under consideration in the θ dimension, θ' (solid square), there are 3^n possible paths (dotted line illustrates one of them) from $\phi = 1$ to $\phi = n$ under the connectivity constraint $|r_p(\theta, \phi-1) - r_p(\theta, \phi)| \leq 1$. (e) Candidate paths for the calculation of the set of partial solutions at $(\phi, \ominus) = (\beta, b)$ for a look ahead of $n = 2$. (f) The triangular regions within which voxels intensities are incremented so that the control point (r', θ', ϕ') is included in the surface. (g) and (h) Calculation of the gradient of volume-labeled objects. (g) Two-dimensional slice image of volume labeled nuclei. (h) Directional gradient image of the same slice showing high intensities at the border of the object being segmented (red boundary), but low intensities at the close-by borders of neighboring objects (arrows).

surface surrounding the internal point P_C and intersecting the border point P_B. Mathematically, it is expressed as $\max_{\{p\}}(\sum_{\varphi,\theta} I(\varphi, \theta, r_p(\varphi, \theta)))$, where $r_p(\varphi, \theta)$ is the radial coordinate of the voxel on the path at $(\phi, \ominus)$ and $I(\phi, \ominus, r)$ is the intensity at $(\phi, \ominus, r)$. However, we believe that determination of a truly global optimum of this expression is impractical for two reasons. First, before calculating the above expression, manual initialization is required to ensure that the algorithm finds a surface enclosing only the selected object. This involves marking the border point P_B, but there is no guarantee that P_B lies on the globally optimal surface. Similarly, there is no guarantee that the subsequently calculated 2-D border is part of the globally optimal surface. Second, the connectivity constraints, which are necessary to guarantee that the surface is connected to itself at every point, limit the maximum curvature (κ) of the discretized surface to $\kappa = 1$ in spherical coordinates. This is because the radial coordinate can change by at most ± 1 for each increment of ϕ or $\ominus$. However, the true surface may exceed this curvature limit. As a consequence of these limitations to calculation of a globally optimal surface, we

have adopted a suboptimal method, which we show through simulations achieves a very close approximation to the global optimum.

Our 3-D segmentation algorithm (Fig. 1(a) box 6) utilizes the border found by the 2-D DP algorithm [red border in Fig. 2(a)] as a set of control points for calculating the complete surface of the object, where the initially marked border point, P_B is set to the "north pole" (N) where $\phi = 0$. The point at $\phi = 128$ is designated the "south pole" (S). Determination of the 3-D surface starts from S and is calculated as a succession of paths at fixed elevation angles (ϕ) from $\phi = 128$ to $\phi = 80$ and where each path [green curves in Fig. 2(b)] starts and ends on the 2-D border (Fig. 1(a) box 6). Then the surface is calculated starting from N ($\phi = 0$) to $\phi = 80$ (Fig. 1(a) box 8). The surface is subject to the above-mentioned connectivity constraint meaning that the radial coordinate can at most change by one between adjacent voxels on the surface. Mathematically, this is $|r_p(\phi - 1, \ominus) - r_p(\phi, \ominus)| \leq 1$ and $|r_p(\phi, \ominus - 1) - r_p(\phi, \ominus)| \leq 1$, which limits the number of candidate values of $r_p(\phi, \ominus)$ to 3 for a given value of $r_p(\phi-1, \ominus)$ or $r_p(\phi, \ominus-1)$. Starting from both poles, rather than proceeding from one pole to the other avoids a problem close to the destination pole ($|\phi - 64| > 42$) where reassignment of the path to fewer elements in the ϕ direction can violate connectivity constraints.

The simplest suboptimal solution to the calculation of the succession of paths is known as the "greedy" algorithm [32]. It finds the 2-D path at each ϕ successively with the only constraint that the path is connected to its preceding path. However, it does not necessarily find the best overall solution, because it can not anticipate the location of subsequent paths. This problem was alleviated here by extending the "greedy" algorithm to "look ahead" n subsequent paths. Thus, the look ahead method finds a "ribbon" around the object of n voxels wide in the ϕ dimension, where the solution is globally optimal for the ribbon and the ribbon is connected to the previously determined ribbon [Fig. 2(d)]. Since this method determines the surface over the large number of voxels in the ribbon, it is highly robust against image noise and gaps in the true surface. Thus, it closely approximates the global optimum. This was our primary reason for choosing this method.

Calculation of the optimum surface for each ribbon, $\max_{\{p\}}(\sum_{\varphi,\theta} I(\varphi, \theta, r_p(\varphi, \theta)))$ is complicated because DP does not straightforwardly extend to more than two dimensions due to the loss of the natural ordering of surface voxels on which the recursion depends. Nevertheless, DP can still be utilized in the $\ominus$ dimension since the best partial surface to $\{r_p(\phi, \ominus), r_p(\phi+1, \ominus), \ldots, r_p(\phi+n-1, \ominus)\}$ that is connected to $\{r_p(\phi, \ominus - 1), r_p(\phi + 1, \theta - 1), \ldots, r_p(\phi + n - 1, \ominus - 1)\}$ contains the best partial surface to $\{r_p(\phi, \ominus - 1), r_p(\phi+1, \ominus - 1), \ldots, r_p(\phi + n - 1, \ominus - 1)\}$.

In the ϕ dimension, however, a combinatorial search must be employed. Fig. 2(d) illustrates the combinatorial approach, where for a given point under consideration in the θ dimension, θ' (solid square), there are 3^n possible paths (dotted line illustrates one of them) from $\phi = 1$ to $\phi = n$ under the connectivity constraint $|r_p(\theta', \phi - 1) - r_p(\theta', \phi)| \leq 1$. The partial solution is calculated for each of these paths, and it is the summation of voxel intensities in the path plus the maximum of the partial solutions from the connected paths at $\theta = \theta' - 1$. The set of partial solutions is recorded for subsequent calculation of the path at $\theta' + 1$. We point out that a combinatorial search in both the θ and ϕ dimensions would result in 3^{n^2} possible permutations, which would be computationally prohibitive to calculate. Thus, DP is a vital component to our 3-D segmentation algorithm. In practice, we set n to 4 to limit computation time.

We present the combinatorial search in further detail for $n = 2$, noting that calculations for $n > 2$ follow straightforwardedly. Referring to Fig. 2(e), we explain the calculation of the set of possible solutions along the line at $(\phi, \ominus) = (\beta, b)$. This line is equivalent to the dashed line from $(\phi, \ominus, r) = (0, \theta', 0)$ to the solid square in Fig. 2(d)and extending beyond the solid square in the $+r$ direction. We assume that the voxels $(\phi, \ominus, r) = (\alpha, b, 3)$ (solid black circle) and $(\alpha, a, 3)$ (open circle) in Fig. 2(e) have already been determined to be on the surface. The partial solutions that are connected to $(\alpha, b, 3)$ for all the pairs of voxels, where $\phi = \beta$ for one voxel and $\phi = \gamma$ for the other voxel must be calculated and stored for the subsequent calculation at $\ominus = c$. There are nine pairs of partial solutions indicated by the upward arrows from $(\alpha, b, 3)$. For each pair, if the two voxels are connected, the partial solution is calculated by adding to the sum of the intensities of the two voxels, the maximum accumulated intensity from the set of partial, connected solutions calculated previously at $\ominus = a$. For example, for the pair of voxels: $(\beta, b, 3)$ and $(\gamma, b, 3)$ [indicated by solid and open squares, respectively, in Fig. 2(e)], all permutations of one blue voxel connected to one green voxel will be inspected for the maximum partial solution.

Following the calculation of an optimal "ribbon," the solution for $\{r_p(\phi, \ominus) | 0 \leq \ominus < 256\}$ is kept, and the calculation is repeated for the "ribbon" $\{r_p(\phi+1, \ominus), r_p(\phi+2, \ominus), \ldots, r_p(\phi+n, \ominus) | 0 \leq \ominus < 256\}$.

4) "Control" Points Must Be on the Surface: Interactively marked "control" points must become surface voxels even if they are far from the previously computed surface (Fig. 1(a) box 10). To guarantee this, "attraction" regions are set up that guide the evolving surface to the control point in the ϕ direction. This is achieved by incrementing the intensities of the control point and all voxels in adjacent, triangular regions in the $+\phi$ and $-\phi$ directions by a large number [Fig. 2(f)]. 6400 was selected as being sufficiently large and the attraction range was set to 40. Both numbers can be changed if necessary. These attraction regions do not bias the surface, because adding the same (large) constant to voxels inside the triangle does not change the selection of surface voxels inside the triangle. There is no need to attract the surface in the θ dimension to a control point or in 2-D segmentation, because the global search of DP guarantees inclusion of the control point. For control points near the north or south poles, the triangular region is in the negative or positive directions only.

Automatically set control points are used to ensure that the two evolving surfaces from the north and south poles agree at $\phi = 80$ (Fig. 1(a) box 7). When the evolving surface from the South pole reaches $\phi = 80$, an attraction region is set up by incrementing the intensity of all surface voxels at $\phi = 80$ and all voxels in a triangular region in the negative ϕ direction. This attraction region guides the evolving surface from the north pole to the surface from the south pole at $\phi = 80$.

5) Calculation of the Gradient of Volume-Labeled Objects: For the case of volume-labeled objects [Fig. 2(g)], it is necessary to transform images so that the intensities at the object's surface becomes higher than elsewhere. Rather than applying a standard gradient magnitude filter, we take advantage of the user-defined internal point P_C to calculate the gradient in the outward direction from the object. Mathematically, the gradient of a voxel at radial position, r along an outward line from the origin (O) is calculated as $I_{r-2} + 2I_{r-1} - 2I_{r+1} - I_{r+2}$. This directional derivative has the major advantage that voxels on the surface of the object being segmented will generally receive positive values, because the intensities inside the object (I_{r-2} and I_{r-1}) are usually greater than intensities outside the object (I_{r+2} and I_{r+1}). In contrast, voxels on the surface of close-by objects receive negative values [Fig. 2(h)]. The result is that the computed surface is much more likely to follow the surface of the object being segmented rather than "jump across" to the surface of a close-by adjacent object.

C. Assessment Using Simulated Objects

Three types of simulated objects were prepared: a sphere, a cube, and a sphere juxtaposed to the concave surface of a dish-like object. In all cases, different levels of Gaussian noise were added to the image, and none of the segmented surfaces were interactively corrected.

1) Spheres: Spheres were used to evaluate the accuracy of the algorithm in the presence of random noise and to assess the increased accuracy gained by using "look ahead." They were created directly in the spherical coordinate system with the ϕ and $\ominus$ dimensions discretized each into 256 steps. For all spheres, voxel intensity values were assigned according to the expression $150 - |25 - r'|$ for $0 \leq r' \leq 50$. This resulted in an increasing intensity from $r' = 0$ to $r' = 25$ and then decreasing to $r' = 50$. Different levels of Gaussian noise were added with zero mean and standard deviation (σ) equaling: 5, 10, 15, 20, 30, and 40. These noise levels are significantly greater than those expected in confocal microscope images where $\sigma \ll 10$. The center and border points were set to the origin and to a random point at $r' = 25$, respectively. For each noise level, the segmented surface was determined with the look ahead (n) set to 1, 2, 3, and 4 for 20 trials. The accuracy of the segmented surface, $r_p(\phi, \ominus)$ was parameterized by the root mean square (rms) deviation from the true surface ($r = 25$) using the expression $\sqrt{(\sum_{\varphi,\theta}(r_p(\varphi,\theta) - 25)^2/24\,946)}$, where 24 946 is the total number of radii in the image. In addition, we calculated the average intensity of the voxels on each segmented surface (AVSCR) in order to assess bias of the algorithm towards voxels that have high intensity due to noise, but are not on the true surface.

2) Cubes: Cubes were used to model closely packed cells in tissue and to assess segmentation accuracy at edges and corners. Each simulation consisted of a $100 \times 100 \times 67$ voxel image, with the z axis aspect ratio of 1.5. The cube was embedded in the image with faces at $x = 15, x = 84, y = 15, y = 84, z = 10, z = 56$. All voxels had a minimum intensity of 100 and those voxels on the faces were set to 120. Voxels within four units of one of the faces were given values between 100 and 120 to simulate the finite spatial resolution of the confocal microscope resulting in apparent thickness of cell boundaries. Gaussian noise was added with zero mean and σ equaling: 5, 10, 15, 20, 30, and 40.

In the spherically transformed space, the surfaces of cubes and other objects have some adjacent coordinates (ϕ and $\ominus$) that differ by more than one in the radial dimension. We call these coordinate transitions "large steps," and they cannot by definition be modeled exactly by the segmentation algorithm and, therefore, are a source of error.

Fifty trials were run for each value of σ. Each trial consisted of the random selection of a central slice uniformly distributed in the range $z = [27, 38]$, an internal point where both x and y were uniformly distributed in the range [40, 59], and an edge point with $x = 15$ and y uniformly distributed in the range [15, 84]. Each trial was assessed using the rms deviation along radial lines between the true border and the segmented border in the initial 2-D slice image (2DRMS), the rms deviation between the true and segmented surface of the segmented cube (3DRMS), the maximum absolute difference between the true and segmented surfaces averaged over the trials (MAXDIF) and the mean voxel intensity of each segmented surface (AVSCR).

3) Sphere Juxtaposed to the Concave Surface of a Dish-Like Object: Certain types of cells (and cell nuclei) in tissue, which are in close proximity to each other frequently possess concavities. In order to demonstrate the performance of the segmentation method for proximal objects and at concavities, 3-D images were generated simulating a sphere juxtaposed to the concave surface of a dish-like object, as defined in Fig. 3(a).

The pair of objects simulated either volume labeled cells by setting internal voxels to 255 or surface labeled cells by setting only surface voxels to 255. Other voxels were set to zero. Blurring of the microscope was simulated by convolving the images with a Gaussian kernel of $\sigma = 2$ followed by adding Gaussian noise with a mean of zero and $\sigma = 90$ and renormalizing the intensity range to [0, 255] to simulate random noise significantly greater than expected from confocal microscopy. Objects were segmented before and after blurring and adding noise to the images. Segmentation accuracy was assessed by calculating the rms and mean deviations between the segmented and true objects over the entire surfaces and only in the concave contact region [red box in Fig. 3(a)].

D. Preparation and Image Acquisition of Biological Samples

1) Mouse Embryo: Early somite stage (E3.0–5) mouse embryos were obtained from NIH Swiss female mice. Embryos were fixed with 2% paraformaldehyde/phosphate buffered saline (PBS) for 30 min at room temperature (RT) and labeled with the fluorescent dye, Alexa 633-phalloidin (Molecular Probes, Eugene, OR), which labels cortical actin concentrated on the cell surfaces. Images of samples mounted in Vectashield were acquired using a 40X, 1.3 numerical aperture (NA) oil objective lens, and pinhole of 1 airy unit on an LSM 510 confocal microscope (Carl Zeiss Inc., Thornwood, NY). Excitation was with 633-nm laser light and emitted light above 650 nm was acquired. Voxel size was 0.22 μm in the x and y dimensions

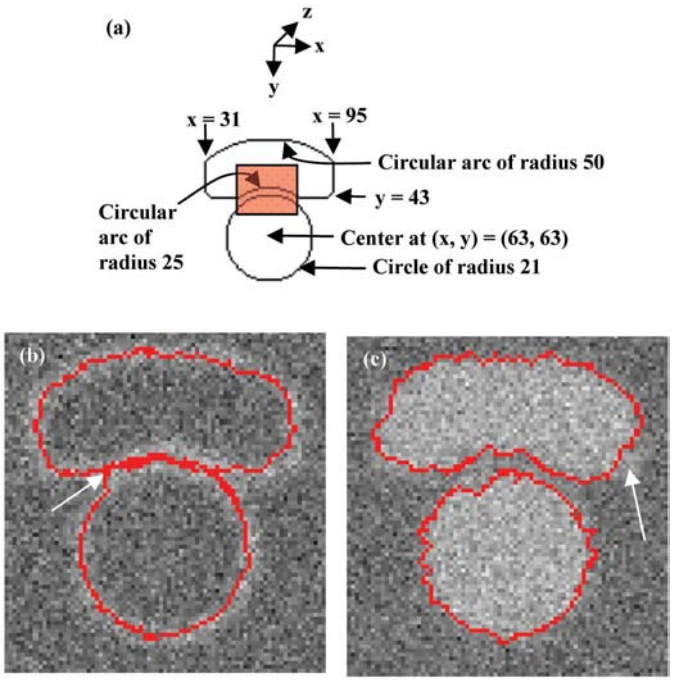

Fig. 3. Computer simulation of a sphere juxtaposed to the concave surface of a dish-like object. (a) Central slice through the 3-D image showing coordinates of the objects. The image is $127 \times 127 \times 127$ voxels and the objects are circularly symmetric about the axis $(x, z) = (63, 63)$. All circular arcs are centered at $(x, y, z) = (63, 63, 63)$. The red box indicates the contact region between the objects. (b) and (c) Overlay of the original image and segmented surface in the central slice image through the object. (b) Simulation of surface-labeled objects. Arrow indicates "jumping across" of the segmented surface to the adjacent object. (c) Simulation of solid-labeled objects. Arrow indicates truncation of the segmented object in regions of high curvature.

and 0.44 μm in the z dimension resulting in 20–40 z slices per cell. The 3-D image was $512 \times 512 \times 167$ voxels.

2) MCF-10A Acinus: Mature acini were grown for 14 days from single MCF-10A cells on Basement Membrane Extract (Trevigen Inc., Gaithersburg, MD), fixed in 4% paraformaldehyde in PBS for 10 min at RT and labeled with the DNA dye, 4-in, 6-diamidino-2-phenylindole (DAPI). Samples were mounted in Vectashield. Images were acquired using a 63X, 1.4 NA oil objective lens, and pinhole of 1 airy unit on an LSM 510 confocal microscope (Carl Zeiss, Inc.). Excitation was with 405-nm laser light and emitted light between 420 and 480 nm was acquired. Voxel size was 0.14 μm in the x and y dimensions and 0.50 μm in the z dimension, resulting in 10–20 z slices per nucleus. The 3-D image was $1024 \times 1024 \times 70$ voxels.

III. Results

A. Assessment Using Simulated Objects

1) Spheres: Table I(a) reports results for the segmentation of spheres averaged over the trials for different levels of noise and "look ahead." As expected, rms deviations increase with increasing additive noise, but remain small (less than 2 voxels) even for very high noise levels ($\sigma = 40$). This accuracy is particularly impressive given that the effective signal-to-noise ratio (SNR) is much less than 1 close to the true border where mean intensities change by one per radial unit, but σ is between 10 and 40. The typical level of noise in a confocal microscope image is equivalent to $\sigma \ll 10$ where the rms deviation is very acceptable at just over one voxel averaged over the sphere's surface. The rms deviation reduces for increasing look ahead and appears to be reaching a limit at a look ahead of 4, because the reduction in rms deviation from look ahead 3 to look ahead 4 is much smaller than the rms deviation. For example, at $\sigma = 10$, the rms decreases by 1.6% from 1.072 to 1.051 for look ahead 3 versus look ahead 4. This suggests that no further quantitative improvement in accuracy could be achieved by increasing the look ahead above 4 and thus the accuracy at look ahead 4 is very close to the accuracy of a globally optimal search for the surface.

Table I(a) additionally reports the mean voxel intensity on each segmented surface (AVSCR). As expected, AVSCR increases with increasing σ due to the inherent property of the algorithm to seek high intensity voxels. However, the algorithm has still sought the best estimate of the surfaces of the objects given the limited information available in the noisy images.

TABLE I

Results From Simulations. (a) rms Deviation Between the Known Surfaces of Simulated Spheres and the Segmented Surfaces and the Mean Voxel Intensity of Segmented Surfaces (AVSCR) Averaged Over Multiple Trials for Different Levels of Gaussian Noise (σ) and Different Amounts of "Look Ahead." (b) Results for the Segmentation of Cubes. "Large Steps" are Adjacent Points in the ϕ and $\ominus$ Dimensions on the True Surface Where the Radial Distances Differ by More Than One. 2DRMS is the rms Deviation Between the True and Segmented Border in the Initial 2-D Slice Image, and 3DRMS is the Deviation Between the Entire True and Segmented Surfaces. MAXDIF is the Maximum Absolute Difference Between the True and Segmented Surfaces, and AVSCR is the Mean Voxel Intensity of the Segmented Surface. All Results are the Average Over 50 Trials. (c) Results for Segmentation of the Sphere Juxtaposed to the Concave Surface of a Dish-Like Object. RMSALL is the RMS Deviation Averaged Over the Entire Surface of the Object and RMSCONT is the RMS Deviation Averaged Over Only the Contact Region Between the Two Objects

(a)

Noise (σ)	Look Ahead 1		Look Ahead 2		Look Ahead 3		Look Ahead 4	
	RMS	AVSCR	RMS	AVSCR	RMS	AVSCR	RMS	AVSCR
5	0.986	152.12	0.899	152.42	0.872	152.49	0.865	152.51
10	1.269	155.17	1.105	155.85	1.072	156.03	1.051	156.11
15	1.387	158.36	1.259	159.32	1.223	159.62	1.195	159.76
20	1.575	161.46	1.406	162.76	1.382	163.16	1.343	163.36
30	1.816	167.75	1.656	169.61	1.595	170.34	1.550	170.65
40	1.961	174.13	1.770	176.65	1.746	177.57	1.710	177.96

(b)

Noise (σ)	Large Steps	2DRMS	3DRMS	MAXDIF	AVSCR
5.0	121.78	.949	1.168	9.0	119.58
10.0	117.60	1.233	1.461	10.3	120.75
15.0	132.45	1.632	2.426	14.9	122.96
20.0	128.32	1.656	2.544	15.5	122.83
30.0	117.44	2.059	3.606	18.5	125.07
40.0	125.40	2.932	4.725	21.2	127.09

(c)

Object	Labeling (volume / surface)	Blurring (σ)	Noise (σ)	RMSALL	RMSCONT
sphere	volume	0	0	1.23	1.17
sphere	surface	0	0	0.77	0.59
sphere	volume	2	2	1.6	1.59
sphere	surface	2	2	1.57	1.49
dish	volume	0	0	1.05	0.91
dish	surface	0	0	0.96	0.69
dish	volume	2	2	2.08	1.62
dish	surface	2	2	1.35	1.14

2) Cubes: Table I(b) reports the results for cubes, which were used to simulate the close packing of cells in tissues. As expected, both the rms deviation in the initial 2-D slices (2DRMS) and the rms deviation for the entire 3-D surfaces (3DRMS) increase with increasing noise. However, the deviations are greater than spheres for similar levels of noise. This is due to a combination of "large steps" where, by design the algorithm cannot follow the true surface and high noise causing the truncation of corners (Fig. 4 in [10]). Truncation at corners leads to relatively large deviations between segmented and true surfaces (MAXDIF), but can be corrected by subsequent interactive correction. Nevertheless, rms deviations are satisfactory at noise levels expected in confocal microscope images ($\sigma \ll 10$). Also similar to spheres, the mean voxel intensity (AVSCR) of the segmented surfaces increases with increasing noise.

3) Sphere Juxtaposed to the Concave Surface of a Dish-Like Object: Table I(c) reports the rms deviations between the true and segmented surfaces of the two objects averaged over the entire surface (RMSALL) and averaged over only the "contact" region (RMSCONT) [red box in Fig. 3(a)]. Fig. 3(b) and (c) shows overlays of the objects and segmented surfaces in 2-D images through the center of the objects for the cases where blurring and noise have been applied. As expected, rms deviations increased in the presence of blurring and noise, but remained acceptable in all tests (generally < 2). The sources of error were truncation in regions of high curvature [arrow in Fig. 3(c)] as was the case for cubes and a tendency of the border to "jump across" to the adjacent object [arrow in Fig. 3(b)]. These errors could be removed by interactive correction. Notably, rms deviations for both objects were improved slightly in the contact regions compared to the entire surfaces, demonstrating that close proximity of objects did not adversely affect segmentation accuracy. This result is particularly significant, because it demonstrates the robustness of the method for analyzing clustered cells in intact tissue.

B. Assessment Using Biological Samples

Fig. 4 shows the segmentation results for the 3.5 day mouse embryo and acinus of MCF-10A cells grown for 14 days in 3-D culture, respectively, in evenly-spaced xy and yz slices overlaying the original 3-D image and as surface renderings of the segmented cells [Fig. 4(i) and (r)]. The contiguous areas of red in Fig. 4(a)–(d) and (j)–(m) are locations where the segmented surface is parallel to the xy plane of the image. These areas are more frequent in the xy slices than in yz slices, because of the coarser sampling in the z dimension. Although some interactive user correction was required, all 20 whole cells and all 64 cell nuclei were accurately segmented based on visual judgment from the embryo and acinus, respectively. These results, for the first time, demonstrate the correct 3-D segmentation (based on visual assessment) of every whole cell in an organism that was surface labeled and every cell nucleus in an organo-typic cell culture that was volume labeled.

IV. Discussion

We have developed a highly robust method for segmenting whole cells, cell nuclei and other objects from 3-D optical microscopic images of biological samples, thus removing a major bottleneck preventing quantitative and contextual analysis of cells in tissue samples. DP was utilized for calculating the surface, because of its robustness to noise. A modest and acceptable level of *a priori* user interaction was involved in the segmentation process in order to guarantee 100% correct segmentation of each object based on visual assessment. For the vast majority of nuclei [Fig. 4(j)–(r)], only the minimum number of mouse clicks were needed for each nucleus. In a modified user-interface, which was written after completion of this study, the minimum number of clicks was three. In addition, user interaction readily enabled unsatisfactory objects to be excluded.

There were several minor limitations to our approach, which caused small inaccuracies in the segmented surfaces. One of these limitations was the restriction to point convex objects that was inherent in our method, because the segmented surface must have exactly one radius value for each pair of angles (ϕ and θ) in the spherically transformed space. In the Cartesian image, this translates to the requirement that any radial line from the user marked internal point P_C must intersect the surface exactly once. In practice, however, this restriction is not severe since it still allows the detection of objects with concavities (Figs. 3 and 4). Inaccuracies in the segmented surface also occurred at places where the angle between the tangent to the surface and the line from the surface to the internal point P_C approached zero. At these places, the radial distance increased by more than one unit for changes ϕ or θ by ± 1, defying the connectivity constraint imposed by the algorithm. However, this error could readily be reduced by increasing the number of elements assigned to ϕ and θ, although it comes at the cost of less "look ahead."

High noise caused distortion in the surface for two reasons. First, by design the algorithm maximized the accumulated intensity (or intensity gradient) of the segmented surface, favoring voxels with high intensity due to noise to be included in the surface. This was the reason for the increase in AVSCR with increasing noise in Table I. There is no solution to this limitation because more accurate information was not available from the images except by exploiting additional *a priori* information about the shape of the objects. Second, high noise had significant effect at places of high curvature, where the segmented surface tended to "cut the corner" [Fig. 3(c)]. On the other hand, such errors can be interactively corrected. An additional source of error is the spatial resolution of the confocal microscope that is poorer in the axial versus lateral dimensions. This introduced segmentation errors, by biasing the surface inwards or outwards at locations with convex or concave curvature respectively. Nevertheless, such distortions were considered insignificant. Coarse voxel sampling in the x, y, or z dimensions, which may occur when imaging living samples, also decreases segmentation accuracy. However, it is always possible to obtain segmentation results as good as visual judgment by using the interactive correction option.

We envision several improvements to our method to solve some of the limitations mentioned above. The restriction to point convex objects could be removed by enabling interactive marking of multiple centers resulting in the segmentation of multiple partial objects that are then merged together. The

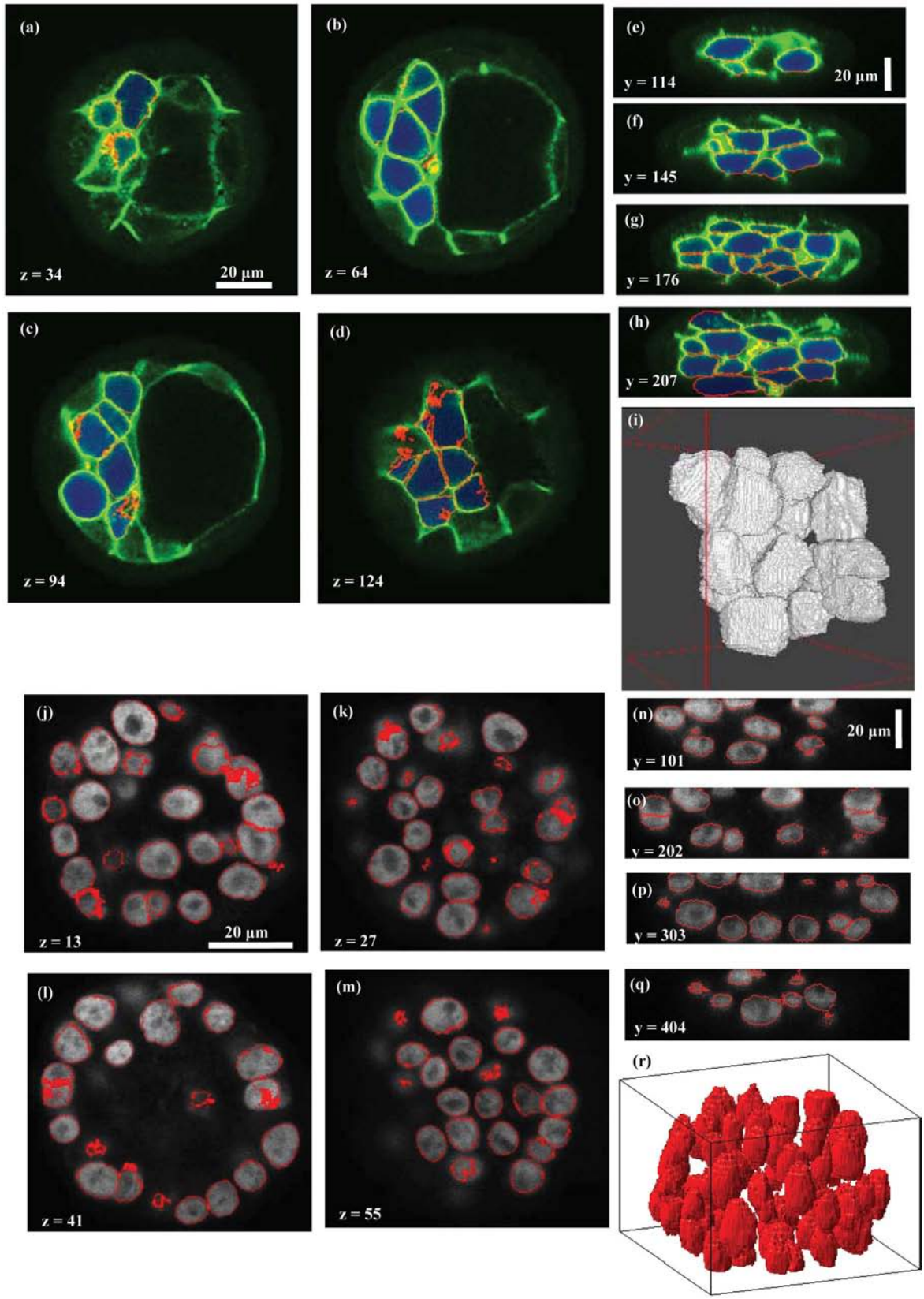

Fig. 4. Biological Results. (a)–(i) Segmentation results of whole cells in a mouse embryo labeled with Alexa633-phalloidin against cortical actin (green). The red overlay shows the surfaces of the segmented cells and blue shows regions determined to be inside the segmented cells by the algorithm. (a)–(d) xy slices equally spaced in the z dimension through the embryo. (e)–(h) xz slices equally spaced in the y dimension through the embryo. (i) Surface rendering of the segmented cells. (j)–(r) Segmentation results of cell nuclei in an acinus of MCF-10A cells labeled with DAPI. The red overlay shows the surfaces of the segmented nuclei. (j)–(m) xy slices equally spaced in the z dimension through the acinus. (n)–(q) xz slices equally spaced in the y dimension through the acinus. (r) Surface rendering of the segmented nuclei.

spatial resolution of the imaging system is an inherent limitation, which although can be improved by deconvolving the images prior to segmentation, imposes an upper limit on the curvature of the object surfaces as represented in the images. This upper limit could be exploited as a constraint on the segmented surface and thus further increasing the robustness of the segmentation.

Although other segmentation methods calculate an actual global optimum [17], [31], whereas ours closely approximates a global optimum, other errors and restrictions that are common to all methods exist. These include the situation when the calculated surface differs from visual judgment due to poor choice of cost function, when the optimum surface encloses more than one object and when objects are not point convex. It is likely that a combination of different image analysis approaches will ultimately prove more accurate and thus lead to further automation than the use of only one approach. For example, Park and Keller [33] combined four image analysis approaches: watershed algorithm, snakes, multiresolution analysis, and dynamic programming to segment 2-D images of cells. Zeng *et al.* [34] combined dynamic programming with anisotropic diffusion for finding 3-D sulcal ribbons in magnetic resonance images. Also a significant subset of segmentation algorithms applied to cells and nuclei are model based in order to exploit shape and texture features of the objects [26]. Thus, it is likely that a merging of DP with a model-based method will increase robustness and thus reduce user interaction. A further consideration is the close packing of cells in tissue that to a large extent determines cell and nuclear shapes. This property of tissue has not yet been taken full advantage of in image analysis and implies that algorithms designed for segmenting multiple objects simultaneously will be more robust than segmenting each object separately [16].

We believe our 3-D segmentation method will be most useful in applications where tens to hundreds of cells per sample require segmentation, with the confidence that the segmentation of each cell chosen for segmentation is accurate based on visual judgment. Following 3-D segmentation, it is possible to analyze the structural and molecular properties of individual cells using morphological and texture features [35]. However, not only is it possible to analyze the properties of individual cells, but it is possible to quantitatively analyze the contextual relationships between cells in their true tissue environment using spatial statistical methods [36]. The combination of individual and contextual quantitative feature measurement is a rich source of information that will lead to a much deeper understanding of numerous biological processes in multicellular organisms, particularly those related to tissue development and solid tumorigenesis. Examples include understanding genetic variations from cell to cell that arise in solid tumors [37], interactions between drug-resistant and drug-sensitive cells [38], and cell polarity in tissue development and homeostasis [1].

In conclusion, we have developed an accurate, practical, and robust method for segmenting fluorescence labeled cells from 3-D confocal microscope images. This method enables the quantitative analysis of the individual properties of cells and their contextual relationships inside intact tissue samples.

REFERENCES

[1] K. Amonlirdviman, N. A. Khare, D. R. Tree, W. S. Chen, J. D. Axelrod, and C. J. Tomlin, "Mathematical modeling of planar cell polarity to understand domineering nonautonomy," *Science*, vol. 307, no. 5708, pp. 423–426, Jan. 2005.

[2] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," *Int. J. Comp. Vis.*, vol. 22, pp. 61–79, 1997.

[3] U. Montanari, "Optimal detection of curves in noisy pictures," *Commun. ACM*, vol. 14, pp. 335–345, 1971.

[4] S. J. Lockett and B. Herman, "Automatic detection of clustered, fluorescent-stained nuclei by digital image-based cytometry," *Cytometry A*, vol. 17, pp. 1–12, 1994.

[5] H. I. Christiansen and J. L. Crowley, *Experimental Environments for Computer Vision and Image Processing*. Singapore: World Scientific, 1994, pp. 107–126.

[6] J. M. Geusebroek, A. W. M. Smeulders, and H. Geerts, "A minimum cost approach for segmenting networks of lines," *Int. J. Comput. Vis.*, vol. 43, pp. 99–111, 2001.

[7] E. Meijering, M. Jacob, J. C. Sarria, P. Steiner, H. Hirling, and M. Unser, "Design and validation of a tool for neurite tracing and analysis in fluorescence microscopy images," *Cytometry A*, vol. 58A, no. 2, pp. 167–176, Apr. 2004.

[8] D. Sage, F. R. Neumann, F. Hediger, S. M. Gasser, and M. Unser, "Automatic tracking of individual fluorescence particles: Application to the study of chromosome dynamics," *IEEE Trans. Image Process.*, vol. 14, no. 9, pp. 1372–1383, Sep. 2005.

[9] A. Dufour , V. Shinin, S. Tajbakhsh, N. Guillén-Aghion , J.-C. Olivo-Marin, and C. Zimmer, "Segmenting and tracking fluorescent cells in dynamic 3-D microscopy with coupled active surfaces," *IEEE Trans. Image Process.*, vol. 14, no. 9, pp. 1396–1410, Sep. 2005.

[10] D. Baggett, M. A. Nakaya, M. McAuliffe, T. P. Yamaguchi, and S. Lockett, "Whole cell segmentation in solid tissue sections," *Cytometry*, vol. 67a, no. 2, pp. 137–143, Oct. 2005.

[11] A. Bartesaghi, G. Sapiro, and S. Subramaniam, "An energy-based three dimensional segmentation approach for the quantitative interpretation of electron tomograms," *IEEE Trans. Image Process.*, vol. 14, no. 9, pp. 1314–1323, Sep. 2005.

[12] J.-P. P. Starink and J. J. Gerbrands, "Three-dimensional object delineation by dynamic programming," *Bioimaging*, vol. 2, pp. 204–211, 1994.

[13] S. Osher and J. Sethian, "Fronts propagating with curvature-dependent speed-algorithms based on Hamilton–Jacobi formulations," *J. Comput. Phys.*, vol. 79, pp. 12–49, 1988.

[14] J. A. Sethian, *Level Set Methods and Fast Marching Methods*. Cambridge, U.K.: Cambridge Univ. Press, 1999.

[15] C. Ortiz De Solorzano, R. Malladi, S. A. Lelievre, and S. J. Lockett, "Segmentation of nuclei and cells using membrane protein markers," *J. Microsc.*, vol. 201, pp. 404–415, Mar. 2001.

[16] C. Zimmer and J. C. Olivo-Marin, "Coupled parametric active contours," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 11, pp. 1838–1842, Nov. 2005.

[17] B. Appleton and H. Talbot, "Globally minimal surfaces by continuous maximal flows," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 1, pp. 106–118, Jan. 2006.

[18] R. Beare, "A locally constrained watershed transform," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, pp. 1063–1074, Jul. 2006.

[19] B. Fang, W. Hsu, and M. L. Lee, "On the accurate counting of tumor cells," *IEEE Trans. Nanobiosci.*, vol. 2, pp. 94–103, Jun. 2003.

[20] H. T. 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.

[21] A. Pinidiyaarachchi and C. Wählby, "Seeded watersheds for combined segmentation and tracking of cells," presented at the ICIAP 2005: Int. Conf. Image Anal. Process., Cagliari, Italy, 2005.

[22] A. I. Dow, S. A. Shafer, J. M. Kirkwood, R. A. Mascari, and A. S. Waggoner, "Automatic multiparameter fluorescence imaging for determining lymphocyte phenotype and activation status in melanoma tissue sections," *Cytometry A*, vol. 25, no. 1, pp. 71–81, Sep. 1996.

[23] J. Fehr, O. Ronneberger, H. Kurz, and H. Burkhardt, "Self-learning segmentation and classification of cell-nuclei in 3D volumetric data using voxel-wise gray scale invariants," in *Patt. Recog. Proc.*, 2005, vol. 3663, pp. 377–384.

[24] J. X. Hu, A. Razdan, G. M. Nielson, G. E. Farin, D. P. Baluch, and D. G. Capco, "Volumetric segmentation using weibull E-SD fields," *IEEE Trans. Vis. Comput. Graph.*, vol. 9, no. 3, pp. 320–328, Jul./Sep. 2003.

[25] K. C. Strasters and J. J. Gerbrands, "Three-dimensional image segmentation using a split, merge and group approach," *Pattern Recognit. Lett.*, vol. 12, pp. 307–325, 1991.

[26] G. Lin, M. K. Chawla, K. Olson, J. F. Guzowski, C. A. Barnes, and B. Roysam, "Hierarchical, model-based merging of multiple fragments for improved three-dimensional segmentation of nuclei," *Cytometry A*, vol. 63, pp. 20–33, 2005.

[27] J. H. Wang, A. Trubuil, C. Graffigne, and B. Kaeffer, "3-D Aggregated object detection and labeling from multivariate confocal microscopy images: A model validation approach," *IEEE Trans. Syst., Man Cybern., Part B*, vol. 33, no. 4, pp. 572–581, Aug. 2003.

[28] F. Al-Awadhi, C. Jennison, and M. Hurn, "Statistical image analysis for a confocal microscopy two-dimensional section of cartilage growth," *J. R. Stat. Soc. Ser. C*, vol. 53, pp. 31–49, 2004.

[29] B. L. Luck, K. D. Carlson, A. C. Bovik, and R. R. Richards-Kortum, "An image model and segmentation algorithm for reflectance confocal images of in vivo cervical tissue," *IEEE Trans. Image Process.*, vol. 14, no. 9, pp. 1265–1276, Sep. 2005.

[30] J. L. Marroquin, E. A. Santana, and S. Botello, "Hidden markov measure field models for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 11, pp. 1380–1387, Nov. 2003.

[31] Y. Boykov and G. Funka-Lea, "Graph cuts and efficient N-D image segmentation," *Int. J. Comput. Vis.*, vol. 70, pp. 109–131, 2006.

[32] P. E. Black, "Greedy algorithm," in *Dictionary of Algorithms and Data Structures*, P. E. Black, Ed. Gaithersburg, MD: NIST, Feb. 2, 2005.

[33] J. Park and J. M. Keller, "Snakes on the watershed," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 10, pp. 1201–1205, Oct. 2001.

[34] X. Zeng, L. H. Staib, R. T. Schultz, H. Tagare, L. Win, and J. S. Duncan, "A new approach to 3-D sulcal ribbon finding from MR images," in *2nd Int. Conf. Med. Image Comput. Computer-Assisted Intervention*, Cambridge, U.K., Sep. 19–22, 1999, pp. 148–157.

[35] T. Zhao, M. Velliste, M. V. Boland, and R. F. Murphy, "Object type recognition for automated analysis of protein subcellular location," *IEEE Trans. Image Process.*, vol. 14, no. 9, pp. 1351–1359, Sep. 2005.

[36] D. Knowles, C. Ortiz de Solorzano, A. Jones, and S. J. Lockett, D. L. Farkas and R. C. Leif, Eds., "Analysis of the 3-D spatial organization of cells and subcellular structures in tissue," in *Opt. Diagnostics Living Cells III*, 2000, vol. 3921, Proc. SPIE, pp. 66–73.

[37] K. Chin, C. Ortiz de Solorzano, D. Knowles, A. Jones, W. Chou, E. G. Rodriguez, W. L. Kuo, B. M. Ljung, K. Chew, K. Myambo, M. Miranda, S. Krig, J. Garbe, M. Stampfer, P. Yaswen, J. W. Gray, and S. J. Lockett, "In situ analyzes of genome instability in breast cancer," *Nat. Genet.*, vol. 36, pp. 984–988, 2004.

[38] A. Starzec, D. Briane, M. Kraemer, J.-C. Kouyoumdjian, J.-L. Moretti, R. Beaupain, and O. Oudar, "Spatial organization of three-dimensional cocultures of adriamycin-sensitive and -resistant human breast cancer MCF-7 cells," *Biol. Cell*, vol. 95, pp. 257–264, 2003.