

Improved Automatic Detection and Segmentation of Cell Nuclei in Histopathology Images

Yousef Al-Kofahi, Wiem Lassoued, William Lee, and Badrinath Roysam*, Senior Member, IEEE

Abstract—Automatic segmentation of cell nuclei is an essential step in image cytometry and histometry. Despite substantial progress, there is a need to improve accuracy, speed, level of automation, and adaptability to new applications. This paper presents a robust and accurate novel method for segmenting cell nuclei using a combination of ideas. The image foreground is extracted automatically using a graph-cuts-based binarization. Next, nuclear seed points are detected by a novel method combining multiscale Laplacian-of-Gaussian filtering constrained by distance-map-based adaptive scale selection. These points are used to perform an initial segmentation that is refined using a second graph-cuts-based algorithm incorporating the method of alpha expansions and graph coloring to reduce computational complexity. Nuclear segmentation results were manually validated over 25 representative images (15 *in vitro* images and 10 *in vivo* images, containing more than 7400 nuclei) drawn from diverse cancer histopathology studies, and four types of segmentation errors were investigated. The overall accuracy of the proposed segmentation algorithm exceeded 86%. The accuracy was found to exceed 94% when only over- and undersegmentation errors were considered. The confounding image characteristics that led to most detection/segmentation errors were high cell density, high degree of clustering, poor image contrast and noisy background, damaged/irregular nuclei, and poor edge information. We present an efficient semiautomated approach to editing automated segmentation results that requires two mouse clicks per operation.

Index Terms—Image cytometry, cell nuclei, histopathology, segmentation.

I. INTRODUCTION

THE GOAL of this study is to develop efficient and accurate algorithms for detecting and segmenting cell nuclei in 2-D histological images. This is commonly a first step to counting cells, quantifying molecular markers (antigens) of

Manuscript received February 18, 2009; revised June 26, 2009. First published October 30, 2009; current version published March 24, 2010. This work was supported by the U.S. Army Breast Cancer Research Program under Grant W81XWH-07-1-0325 BC061142, by the National Institute of Biomedical Imaging and Bioengineering under Grant R01 EB005157, and by the National Science Foundation (NSF) under Grant EEC-9986821 to the Center for Subsurface Sensing and Imaging Systems. Asterisk indicates corresponding author.

Y. Al-Kofahi is with the Department of Electrical, Computer and Systems Engineering (ECSE), Rensselaer Polytechnic Institute, Troy, NY 12180 USA (e-mail: alkofy@rpi.edu).

W. Lassoued is with Dr. William's Lee Laboratory, Abramson Cancer Center, University of Pennsylvania, Philadelphia, PA 19104 USA (e-mail: wiemllas@yahoo.fr).

W. Lee is with the Division of Hematology-Oncology, Abramson Cancer Center, University of Pennsylvania, Philadelphia, PA 19104 USA (e-mail: leemingf@mail.med.upenn.edu).

*B. Roysam is with the Department of Electrical, Computer and Systems Engineering (ECSE), Rensselaer Polytechnic Institute, Troy, NY 12180 USA (e-mail: roysam@ecse.rpi.edu).

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/TBME.2009.2035102

interest in healthy and pathologic specimens [1], [2], and also for quantifying aspects of normal/diseased tissue architecture [1]. The cell nuclei may be stained using fluorescent markers [e.g., 4',6'-diamidino-2-phenylindole (DAPI)], or with histochemical stains (e.g., hematoxylin). It is important in these applications to be able to detect the correct number of cells with high accuracy, and to delineate them accurately with utmost automation and minimal human effort. It is also helpful to be able to easily adapt the software algorithms to images of different tissues captured under differing imaging conditions.

Automated segmentation of cell nuclei is now a well-studied topic for which a large number of algorithms have been described in the literature [2]–[18], and newer methods continue to be investigated. The main challenges in segmenting nuclei in histological, especially pathological tissue specimens, result from the fact that the specimen is a 2-D section of a 3-D tissue sample. The 2-D sectioning can result in partially imaged nuclei, sectioning of nuclei at odd angles, and damage due to the sectioning process. Furthermore, sections have finite thickness resulting in overlapping or partially superposed cells and nuclei in planar images. The end result of these limitations is a set of image objects that differ considerably from the ideal of round blob-like shapes. Their sizes and shapes in images can be irregular, and not always indicative of their 3-D reality. There is natural variability among nuclear shapes and sizes even when they are ideally sectioned. With pathological samples, nuclei can exhibit unnatural shapes and sizes. Variable chromatin texture is another source of segmentation error—highly textured nuclei are harder to segment, especially when they are densely clustered. Separation of densely clustered cell nuclei is a long-standing problem in this field. The presence of a large number of nuclei in the field (especially whole-slide images) necessitates methods that are computationally tractable, in addition to being effective. Finally, imaging noise in the background regions, especially for fluorescence data, and the presence of spectral unmixing errors in processed multispectral images results in additional errors.

Perhaps the most critical aspect of nuclear segmentation algorithms is the process of detecting a set of points in the image, usually one per cell nucleus and close to its center, that are variously referred to as “markers” or “seeds.” These points are used by subsequent algorithms to delineate the spatial extent of each cell nucleus. Indeed, the accuracy of the segmentation depends critically on the accuracy and reliability of the initial seed points. Several approaches have been used to detect seed points. The early work in this field [3], [19] relied upon the peaks of the Euclidean distance map. This method is often used in conjunction with the watershed algorithm [9] due to its computational efficiency and ready availability. However, it has

the widely acknowledged disadvantage of detecting too many seeds, leading to over-segmentation. Some efforts at addressing this limitation include filtering of seeds based on mutual proximity [3], incorporation of additional cues such as the image intensity gradient [9], and the use of region merging algorithms as a postprocessing step [10], [11]. Another technique is to detect local maxima points in the gray-scale image using the h -maxima transform [16], [20]. This method was found to be overly sensitive to image texture, and resulted in overseeding with our images. The Hough transform [21] has also been used for detecting seed points [2], [6]. This method is practical for nearly circular nuclei, and requires excessive computation. More recently, the very elegant iterative radial voting algorithm was presented in [22], and has been used in several papers [14], [23]. This method requires edge extraction based on gradient thresholding, and a careful choice of several parameters that proved impractical in the automated pathology context. In [24] a regularized centroid transform was used. This method only uses the binarized image and does not exploit additional cues present in the image intensity data. In [8], a gradient flow-tracking algorithm was used. Like the radial voting idea, this method is conceptually elegant. The difficulty with this method in our experiments was the rough chromatin texture that produces inaccurate flow values and/or directions.

In this paper, we present a method that overcomes many of the limitations of the aforementioned methods. It is based on the multiscale Laplacian-of-Gaussian (LoG) filter originally introduced by Lindeberg [25] as a generic blob detection method. Recently, Byun *et al.* [4] used a blob detector based on the LoG filter at a fixed scale (set empirically) to count cells in retinal images. This method offers important advantages, including computational efficiency, ability to exploit shape and intensity information, ease of implementation, especially the ability to specify the approximate expected sizes of nuclei, and robustness to variations. Building upon this study, and keeping in mind the challenges specific to histopathology images noted earlier, we propose a method combining the LoG filter with automatic and adaptive scale selection.

Aside from advances in seed detection, the field of automated image analysis has also witnessed the emergence of a new generation of image segmentation algorithms. Notable among these advances are methods based on graph cuts [26]–[30] that offer the important advantage of computing globally optimal solutions. Additional advances have been reported across the literature. Notwithstanding these advances, several needs have remained. For instance, the graph-cuts algorithm requires effective initialization. For this, we present a method in which the results of seed detection are processed by a new generation of fast clustering algorithms to generate an initial segmentation that is subsequently refined using the graph-cuts segmentation algorithm. Another important need is to be able to segment large connected clusters of nuclei efficiently and accurately. For this, we introduce a novel segmentation algorithm based on automatic graph coloring and the method of α -expansions. Overall, the effectiveness of the combination of these methods is demonstrated on breast histopathology images. Fig. 1 shows a flowchart illustrating the main steps of our method.

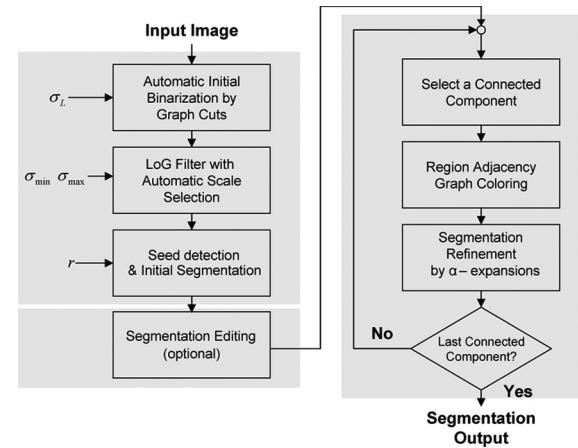


Fig. 1. Flowchart outlining the main steps of the proposed nuclear segmentation algorithm. The initial segmentation and refinement steps are illustrated in Fig. 2. The optional editing step is illustrated in Fig. 3.

II. MATERIALS AND METHODS

A. Histology and Nuclear Staining

For the *in vivo* tissue examples, deparaffinized 5 μM sections of formalin-fixed, paraffin-embedded human breast tissues were rehydrated, and stained with hematoxylin (Vector Laboratories, Burlingame, CA). For the *in vitro* tissue examples, 6 μM sections of OCT frozen blocks of cultured K1735 tumor cells were stained with DAPI (Vector Laboratories, Burlingame, CA).

B. Image Capture

Images of hematoxylin or DAPI stained histopathology slides were captured using a Nuance multispectral camera (CRI, Inc., Woburn, MA) mounted on a Leica epifluorescence microscope (Leica DMRA2). Images were captured using full resolution of the Nuance camera at 8 bits of data per pixel and with 10 nm spectral widths from 420 to 720 nm for brightfield images, and 440–480 nm for DAPI. Nuance software was used to unmix the chromogens and fluorophores in the data cube into a set of nonoverlapping channels based on user-provided reference spectra of the pure chromogens or fluorochromes, respectively. We denote the raw spectral data cube collected by the instrument $I(x, y, \lambda)$, where (x, y) are spatial coordinates of a pixel, and λ is the wavelength. The spectral unmixing procedure results in multiple nonoverlapping channels that are denoted as follows. The nuclear channel is denoted $I_N(x, y)$. This paper is primarily concerned with the processing of $I_N(x, y)$.

C. Automatic Image Binarization

The first step in nuclear segmentation is to separate the foreground pixels in the nuclear channel $I_N(x, y)$ from the background pixels. Several approaches have been presented in the literature, and a survey on image thresholding methods can be found in [31]. Common methods include histogram-based [32], clustering-based [33]–[35], and entropy-based [7] algorithms. More advanced techniques are based on graph-cuts [15] and level set [36] algorithms, but they require good

initialization/training. With this in mind, we propose a hybrid approach that starts with an initial binarization that is subsequently refined using the graph-cuts algorithm.

For the initial binarization, we compute the normalized image histogram, denoted $h(i)$, where i denotes the intensity of a pixel in the range $\{0, \dots, I_{\max}\}$. We found that 128 bins were adequate for these histograms. For the specimens studied here, the histograms were found to be bimodal as expected, and are modeled well by a mixture of two Poisson distributions. This modeling choice was supported by analysis of the image formation process [37], prior literature [37], and empirical comparison against the more commonly used mixture of Gaussians model [34], [38], [39]. We used the Poisson-distribution-based minimum error thresholding algorithm [38], [39]. The normalized image histogram for the mixture of Poisson distributions is written as

$$h(i) = P_0 \times p(i|0) + P_1 \times p(i|1) \quad (1)$$

where P_0 and P_1 are the *a priori* probabilities of the background and foreground regions, and $p(i|j)$, $j = 0, 1$ are Poisson distributions with means μ_j . For a threshold t , the Poisson mixture parameters are given by

$$\begin{aligned} P_0(t) &= \sum_{i=0}^t h(i), & \mu_0(t) &= \frac{1}{P_0(t)} \sum_{i=0}^t i \times h(i) \\ P_1(t) &= \sum_{i=t+1}^{I_{\max}} h(i), & \mu_1(t) &= \frac{1}{P_1(t)} \sum_{i=t+1}^{I_{\max}} i \times h(i). \end{aligned} \quad (2)$$

The optimal threshold t^* is chosen to minimize an error criterion [38], as follows:

$$\begin{aligned} t^* &= \arg \min_t \{ \mu - P_0(t)(\ln P_0(t) + \mu_0(t) \ln \mu_0(t)) \\ &\quad - P_1(t)(\ln P_1(t) + \mu_1(t) \ln \mu_1(t)) \} \end{aligned} \quad (3)$$

where μ is the mean intensity of the complete image. The result of thresholding $I_N(x, y)$ using t^* is refined by incorporating spatial continuity constraints. We seek the pixel labeling $L(x, y)$ that minimizes the following energy function:

$$\begin{aligned} E(L(x, y)) &= \sum_{(x, y)} D(L(x, y); I_N(x, y)) \\ &\quad + \sum_{(x, y)} \sum_{(x', y') \in N(x, y)} V(L(x, y), L(x', y')) \end{aligned} \quad (4)$$

where $N(x, y)$ is a spatial neighbor of pixel (x, y) . The globally optimal labeling is computed using the widely used graph-cuts algorithm [26]–[30], [40], [41]. The first term in (4) is the data term representing the cost of assigning a label to a pixel. It has two possible values depending upon whether the foreground or background model is used. Mathematically, this is written as follows:

$$D(L(x, y); I_N(x, y)) = -\ln p(I_N(x, y)|j = \{0, 1\}). \quad (5)$$

The second term is the pixel continuity term that penalizes different labels for neighboring pixels. Following [26], this is

written as follows:

$$\begin{aligned} V(L(x, y), L(x', y')) &= \eta(L(x, y), L(x', y')) \\ &\quad \times \exp \left(-\frac{[I_N(x, y) - I_N(x', y')]}{2\sigma_L^2} \right) \end{aligned} \quad (6)$$

where

$$\eta(L(x, y), L(x', y')) = \begin{cases} 1, & \text{if } L(x, y) \neq L(x', y') \\ 0, & \text{if } L(x, y) = L(x', y'). \end{cases}$$

The V -term penalizes different labels for neighboring pixels when $|I_N(x, y) - I_N(x', y')| < \sigma_L$. In our work, the scale factor σ_L is set empirically to values in the range 20–30 pixels. Lower values are used when the image is smooth, and higher values are used when the nuclear chromatin is highly textured. We used an implementation of the fast max-flow/min-cut algorithm described by Boykov and Kolmogorov [27]. The previous method results in accurate binarization results. Fig. 2(B) provides a visual example of the binarization results for the image in Fig. 2(A).

D. Automatic Seed Detection and Initial Segmentation

The graph-cuts binarization algorithm extracts connected clusters of nuclei that must be separated into individual nuclei. This requires identification of initial markers (a.k.a. seed points) such that there is one marker per cell. For the present work, multiscale LoG filter based approaches proved to be the most effective. The LoG filter is given by

$$\text{LoG}(x, y; \sigma) = \frac{\partial^2 G(x, y; \sigma)}{\partial x^2} + \frac{\partial^2 G(x, y; \sigma)}{\partial y^2} \quad (7)$$

where σ is the scale value, and $G(x, y; \sigma)$ is a Gaussian with 0 mean and scale σ . When applied to an image containing blob-like objects, this filter produces a scale-selective peak response at the center of each object with radius r when $r = \sigma\sqrt{2}$. The main advantage of this filter is that the locations of these peaks are robust to the chromatin texture that has a much smaller scale value compared to the nuclear blobs. The filtering results form a topographic surface that provides a basis for cell segmentation. In addition, as we describe shortly, it provides additional useful information about the boundaries of touching nuclei. A direct application of the multiscale LoG to images of nuclei would be naïve, since our tissue specimens contain a heterogeneous population of cell types with different nuclear sizes. For this, we propose a multiscale LoG with automatic scale selection, as described by Lindeberg [25]. While this multiscale method greatly improved upon the fixed-scale method, as expected, it was nevertheless inadequate, as illustrated in Fig. 2(C) and (D). In particular, this method fails over heterogeneous clusters of nuclei with different sizes, and weak separating edges. In these cases, it is possible for clusters of 2 or more small nuclei to be detected falsely as a single larger blob that may also encroach on smaller blobs in its vicinity. Overcoming this issue requires a more sophisticated control over the scale values. Our

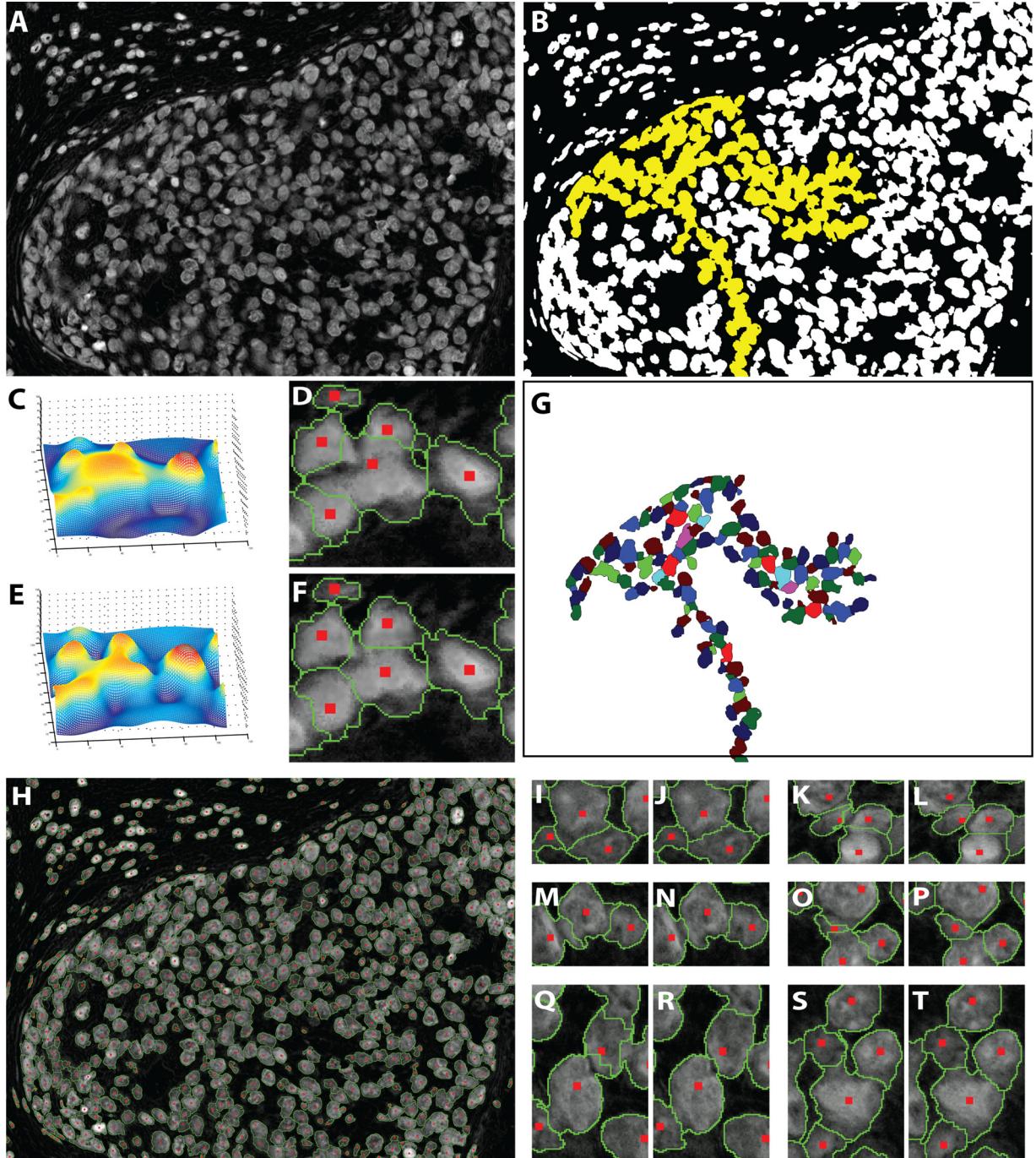


Fig. 2. Illustrating key steps of the proposed nuclear segmentation method. (A) Nuclear channel from spectral unmixing. (B) Foreground extraction results. Pixels marked yellow represent a large connected component. (C) Surface plot of the multiscale LoG filtering results for a small region. (D) Initial segmentation based on the LoG. (E) Surface plot of the distance-map-constrained multiscale LoG. (F) Improved initial segmentation resulting from the distance-constrained LoG. (G) Color coding of the yellow pixels in panel (B). (H) Final segmentation of the image in panel (A). Panels (I and J), (K and L), (M and N), (O and P), (Q and R), and (S and T) indicate initial and final segmentation closeups taken from different regions in the image shown in panel (H).

method to achieve such control relies on exploiting shape and size cues available in the Euclidean distance map $D_N(x, y)$ of the binarized image [42], [43]. Our method proceeds as follows. We compute the response of the scale-normalized LoG filter $\text{LoG}_{\text{norm}}(x, y; \sigma) = \sigma^2 \text{LoG}(x, y; \sigma)$ at multiple scales $\sigma = [\sigma_{\min}, \dots, \sigma_{\max}]$ in steps of 1. Then, we use the Euclidean distance map to constrain the maximum scale values when com-

bining the LoG filtering results across scales to compute a single response surface denoted $R_N(x, y)$ as follows:

$$R_N(x, y) = \arg \max_{\sigma \in [\sigma_{\min}, \sigma_{\max}]} \{\text{LoG}_{\text{norm}}(x, y; \sigma) * I_N(x, y)\} \quad (8)$$

where $\sigma_{\max} = \max\{\sigma_{\min}, \min\{\sigma_{\max}, 2 \times D_N(x, y)\}\}$.

In effect, the distance map constrains the maximum scale value *at each point*. The response $R_N(x, y)$ can be thought of as a topographical surface whose peaks indicate centers of individual nuclei—these are the seed points (nuclear markers). We identify the local maxima of $R_N(x, y)$, and impose a minimum size (based on the expected range of nuclear diameters) to filter out irrelevant minima, as described further in the following subsection. The effect of using the distance map constraint is illustrated in Fig. 2. For instance, panels (C) and (D) show a surface plot of the multiscale LoG and the corresponding initial segmentation (discussed shortly), respectively. Clearly, the central nucleus is oversmoothed and encroaches into its neighbors. The reason for the encroachment is the use of large σ_{\max} that is needed to detect large cells in other regions in the image. Fig. 2(E) and (F), respectively, shows surface plots of $R_N(x, y)$ and the corresponding initial segmentation. It is clear that the accuracy of seed locations and the initial cells boundaries are much improved by imposing the scale constraint.

Using $R_N(x, y)$ and the seed points detected, as described earlier, we compute an initial segmentation of the nuclei, as described shortly. The classical approach used by several authors (including ourselves) is based on the watershed algorithm, and its many variants and improvements [2], [3], [5], [9]–[11], [16], [44], [45]. This method has the advantage of speed, simplicity, absence of adjustable parameters and a degree of flexibility that results from being able to modify the underlying distance map. The main disadvantage of this algorithm for the present task is its sensitivity to even minor peaks in the distance map that results in over segmentation, and detection of tiny regions as objects. To address this problem, we propose an alternate method based on size-constrained clustering. The use of clustering for nuclear/cell segmentation is not new, and predates the watershed method, e.g., [44] and [46]. However, clustering methods have been computationally expensive and difficult to scale to large images. Recently, Wu *et al.* [47] described the local-maximum clustering algorithm [47] that overcomes the previous limitations, and paved the way for the present work. This algorithm has a resolution parameter r that is used to define a search area, denoted $A(x, y)$ of size $2r \times 2r$ around each pixel in $R_N(x, y)$. In a nutshell, this algorithm uses the seed points as cluster centers, and assigns each pixel in the foreground image to these centers to form clusters.

To illustrate the effect of varying the resolution parameter on the clustering (initial segmentation results), Fig. 3 shows two 1-D examples. The shown curve consists of several 1-D blobs with different sizes. When using a small resolution parameter ($r = 3$) all the three blobs are detected. The local maxima (seed points) are indicated in dark red, and the vertical dashed-red lines separate the blobs. The direction and length of the black arrows indicate the assignment of each point to its local maximum in a region (distance in 1-D) defined by the resolution parameter. In panel (B), we use a larger value of the resolution parameter ($r = 6$). Therefore, the small blob (center) is bypassed and pixels to its left are assigned to their local maxima points to its right. Only two blobs (with two corresponding seed points) were detected. Note that the separating lines pass through the minima between the two blobs. These points can be thought of as inflection

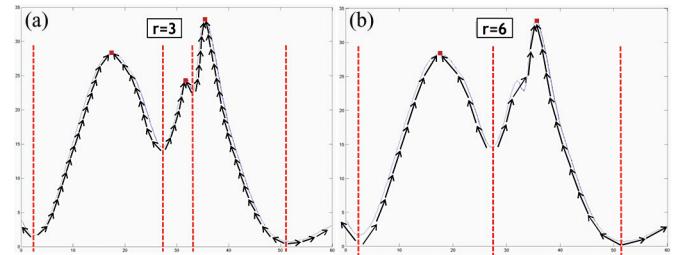


Fig. 3. Illustrating the local-maximum clustering method. A 1-D curve with three blobs is used. The blob in the middle is very small compared to the others. Two values for the resolution parameter are used. Using in panel (A) results in detecting all three blobs. In panel (B), the use of resulted in missing the small blob and merging it to the larger one on the right. The black arrows indicate the assignments of points to their local maxima. The detected seed points are displayed as red dots.

points, where one inflection point is present between blobs. In 2-D images, we have separating boundaries between 2-D blobs.

There are two major advantages of using this method over the watershed method [48]. First, the resolution parameter r provides the ability to avoid forming small clusters, as was clearly shown in the two synthetic examples of Fig. 3. Second, the clustering method works on foreground points only, which makes it faster. In our experiments, this algorithm was comparably fast to the watershed, and often faster. In our study, the parameter r was set empirically in the range of five pixels. Intuitively, r specifies the smallest size of the clusters that we are willing to accept for the next stage of processing.

E. Refinement of Initial Nuclear Segmentation using α -Expansions and Graph Coloring

The segmentation contours produced by the aforementioned cluster analysis are approximate because the clusters are formed using $R_N(x, y)$ rather than the original image, and therefore require further refinement using the image intensity.

The purpose of the refinement is to enhance the initial contours between touching nuclei to better delineate the true edges between them. To meet this goal, three requirements are needed in the segmentation refinement step. First, it should preserve the shape of the cell nucleus based on some shape model. Second, there should be some rules that prevent two or more nuclei from being merged. This happens if we allow two nonneighbor nuclei to encroach into a third one between them until they merge. Third, given the large number of cell nuclei found in real images, the refinement step should allow multiple cell nuclei need to be refined concurrently for efficiency.

As with the binarization refinement, this step is also formulated as an energy minimization that is solved using a graph-cuts algorithm. However, the problem here is more challenging since we have multiple labels, where the number of labels equals the number of cells in a connected component. In the binary case, the graph-cuts method finds the global minima in polynomial time. However, finding a multiway cut, such that the resulting labeling configuration minimizes the energy function is known to be NP-hard. Boykov *et al.* [29] introduced two algorithms, known as α -expansion and $\alpha - \beta$ swap, respectively, that can

efficiently find good approximate solutions to the multiway cut to within a known factor of the global minimum. In this study, the former is used. In the α -expansion method, we formulate the segmentation as an iterative binary labeling problem. At each iteration, one label is set to an integer α , and the rest of the labels are set to another value, denoted $\bar{\alpha}$, where $\bar{\alpha} \neq \alpha$. Then, a binary graph-cuts step (called an expansion) is carried out, in which pixel labels are allowed to change in one direction from $\bar{\alpha}$ to α . The border of the cell labeled with α is refined by expanding it into its neighbors until the energy function is minimized. In the ideal case, the energy function will reach its minimum when the segmentation contour delineates the true nucleus contour, at which the gradient is maximum. The data and smoothness terms of the energy function should be chosen carefully in order to achieve that goal. For the α -expansion method to work, the smoothness term denoted V has to be a metric that requires three conditions to hold [29]. Given any three pixel labels L_1 , L_2 , and L_3 , the three conditions are listed as follows:

- 1) $V(L_1, L_2) = 0 \Leftrightarrow L_1 = L_2$;
- 2) $V(L_1, L_2) = V(L_2, L_1) \geq 0$;
- 3) $V(L_1, L_2) \leq V(L_1, L_3) + V(L_3, L_2)$.

We used a spatially varying smoothness function similar to the one used in the binarization step

$$\begin{aligned} V(L(x, y), L(x', y')) &= \eta(L(x, y), L(x', y')) \\ &\quad \times \exp(-|I_N(x, y) - I_N(x', y')|) \end{aligned}$$

where

$$\eta(L(x, y), L(x', y')) = \begin{cases} \text{Const}, & \text{if } L(x, y) \neq L(x', y') \\ 0, & \text{if } L(x, y) = L(x', y'). \end{cases}$$

The previous smoothness function is reached when the labeling discontinuities occur at the edges between the nuclei. The data term at each pixel depends on the likelihood of assigning it to each label (nucleus). As mentioned earlier, the LoG output profile of each nucleus is roughly similar to a Gaussian. In addition, the elliptical shape of the cell is similar to that of the 2-D Gaussian. Hence, a Gaussian model is used to represent each cell. A maximum likelihood method (MLE) is used to estimate the Gaussian parameters. The inputs to the MLE are the (x, y) coordinates of the pixels of each nucleus, weighted by the pixelwise LoG responses. The likelihood for a pixel (x, y) to be assigned to cell i is $G(x, y; \mu_i, \Sigma_i)$, where μ_i and Σ_i are the mean and the covariance matrix of the i th Gaussian, respectively.

Unfortunately, the α -expansion method is not practical when the number of cells in a connected component is large (>20), leading to an excessive number of expansions that require an impractical amount of computer memory and time. To address this difficulty, we propose a novel method based on graph coloring that is described next. We start by noting that when the α -expansion procedure is applied to an initially segmented cell, it will only expand to its neighboring regions. This is because the expansion procedure will not assign a pixel to a distant cell. Therefore, we turn the problem into using a small number of labels, with each having a large number of cells expanding in

parallel. This is achieved by using a graph coloring approach similar to the one used in [13], but we differ in the use a two-level region adjacency graph. Using the initial segmentation, we build a region adjacency graph. Unlike our prior work [10], [11], we now use a two-level adjacency graph in which a cell is adjacent to its direct neighbors, and to the neighbors' neighbors as well. The second level of adjacency is added to reduce the possibility that two nonneighboring cells with the same color merge after an expansion. The graph is then colored sequentially such that no two adjacent cells have the same color. Choosing the number of colors is a challenge since the well-known four-color theorem [49] does not apply in our case because of our two-level structure. The problem of finding the minimal number of colors is nondeterministic polynomial-time hard (NP-hard). For these reasons, we use a sequential coloring method that is simple to implement, but does not necessarily yield the smallest number of colors. Fig. 2(G) shows the coloring output for the initial segmentation of a connected component. This connected component [also shown in yellow in the binarization, which is shown in Fig. 2(B)] contains 123 nuclei, but only eight colors are used.

The resulting colors are used as labels for the α -expansion step. At each iteration, all the nuclei with a given color are assigned the label α , while all others are assigned $\bar{\alpha}$. Then, α cells are expanded concurrently into $\bar{\alpha}$ cells. As a result, just a few (usually less than 10) expansions are needed regardless of the much larger number of cells in a connected cluster. The smoothness term described earlier is a pixel-level function, since it depends on the local gradient between adjacent pixels, and hence it is not affected by the grouping of cells based on graph coloring. On the other hand, the data term is a function defined on a cell level, since it is based on a Gaussian model of the cell. Therefore, it is modified in order to compute likelihoods to be assigned to groups (colors) rather than individual cells. Suppose that the number of colors assigned to a connected component with N_c cells is N_r , where $N_r \ll N_c$. The likelihood that a pixel (x, y) will be assigned the j th color is

$$p(L(x, y) = j) = \max \left\{ G\left(x, y; \mu_i, \sum_i\right) | C_i = j \right\}, \quad j = 1. \quad (9)$$

The corresponding data term that represents the penalty for assigning pixel (x, y) to color j is

$$D(L(x, y) = j; I_N(x, y)) = -\ln p(L(x, y) = j). \quad (10)$$

The segmentation refinement consists of multiple iterations of α -expansion up to a preset maximum number of iterations (usually 3), or until no change in any pixel label will reduce the energy function. Finally, the resulting objects are renumbered to achieve consistency with the numbers of the initial objects. In Fig. 2, panels (I and J), (K and L), (M and N), (O and P), (Q and R), and (S and T), respectively, represent initial and refined segmentation closeups taken from different regions in the image shown in panel (H) of the same figure.

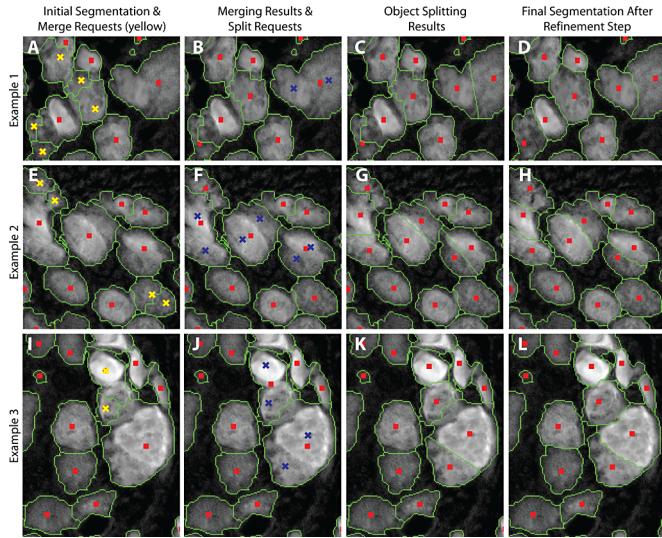


Fig. 4. Illustrating the impact of (optional) seed editing on the final segmentation. (A), (E), and (I) Initial segmentations are shown in the first column for three selected regions. Yellow crosses indicate locations of mouse clicks requesting pairs of segmented objects to be merged. (B), (F), and (J) Results of merging are shown in the second column, in addition to the user's requests to split objects indicated as blue crosses. The blue crosses are initial seeds locations for the two new objects. (C), (G), and (K) Results of splitting are shown in the third column. (D), (H), and (L) Final segmentation after refinement is shown in the fourth column.

F. Efficient Computer-Assisted Editing of Automated Segmentation Results

Automatic segmentation algorithms can provide fast and accurate segmentation of nuclei. However, segmentation errors cannot be avoided even when using optimal parameter values. Hence, human interaction might be needed to fix some of the segmentation errors in order to obtain the highest level of accuracy. Indeed, human interaction should be made minimal by tuning the segmentation parameters to reduce the number of errors. In addition, the editing method should be made easy and fast. In this study, two types of errors (defined in the next section) can be corrected using manual editing. The first type is over segmentation and is corrected by merging fragments of oversegmented nuclei. The merging is performed on pairs of neighbor objects by clicking on one point inside each one of them. Fig. 4(A), (E), and (I) shows closeups of initial segmentation results. User selected points (using mouse clicks) for pair of objects that need to be merged are shown in yellow. The merging results are shown in Fig. 4(B), (F), and (J).

The second type of errors that can be corrected is under segmentation. An undersegmented object is split into two objects by clicking at two points inside it. An automatic splitting method is used to draw an initial contour between the two new objects.

The splitting method starts by computing the approximate Euclidian distances from each point inside the undersegmented object to the manually selected points. Then, the splitting is done based on the minimum of the two distances at each point. Blue crosses in Fig. 4(B), (F), and (J) represent user selected pairs of points indicating objects that need to be split. Fig. 4(C), (G), and (K) shows the automatic splitting results. Changes to the initial

segmentation caused by editing are also applied on all the images needed in the segmentation refinement step. As illustrated previously, the refinement step uses a graph-cuts-based technique (α -expansion), where both the initial segmentation and the LoG output [$R_N(x, y)$] are needed. The editing methods mentioned earlier will update the labels in the initial segmentation. On the other hand, the LoG output $R_N(x, y)$ is updated as follows. In the case of oversegmentation, the LoG output profile of the oversegmented object is replaced by the inverted distance map from the center of the new object produced by merging. On the other hand, the LoG output profile of an oversegmented object is replaced by the inverted distance map from the centers of the two new objects resulting from splitting. Fig. 4(D), (H), and (L) shows the final segmentation after refinement. A related two-mouse-click based technique for interactive whole cell segmentation was presented in [50], where the user segments one cell at a time by clicking on a point at the center of cell and another one on its border. The image is then transformed into polar coordinates, a dynamic programming algorithm is used to find the optimal path on the cell border from left to right, and finally that path is mapped back into Cartesian coordinates.

One drawback of the presented editing tool is the need to scan the image visually to search for segmentation errors. This can be a time consuming task in very large images with large numbers of cells. One possible solution is to adopt the approach presented by the same group [51] in which a segmentation confidence score is computed for each segmented cell nucleus based on some morphological and intensity based features. The lower the confidence score, the more likely the segmentation error is. Then, segmented cell nuclei are sorted based on their confidence scores and the user starts inspecting those them starting from those with low confidence values. Yet another approach that is explored as part of the FARSIGHT project (www.farsight-toolkit.org) is to identify outliers (based on one or more features) to detect nuclei that require further inspection for potential editing. In general, the use of pattern analysis tools to guide the user for expedited editing is a topic of ongoing research.

III. RESULTS AND VALIDATION

The results of automated analysis for 25 representative images (15 *in vitro* images and ten *in vivo* images, containing more than 7400 nuclei in all) drawn from diverse cancer histopathology studies were inspected and scored manually with the goal of developing a conservative assessment of the frequency and types of segmentation errors. The manual scoring was recorded electronically, and a sample is displayed in Fig. 5(A). In this figure, the type of error is indicated as a color-coded dot. Seeds of correctly segmented nuclei are displayed as green dots. Undersegmentation errors (i.e., a failure to split a region into the correct number of nuclei) are indicated as dark blue dots. Oversegmentation errors (i.e., excessive splitting) are indicated as purple dots. In addition to these standard types of errors, we also looked for encroachment errors (indicated as yellow dots) that occur when the automated algorithms do not correctly place the boundary between a pair of touching nuclei. In other words, it is the error in delineating the true border between two

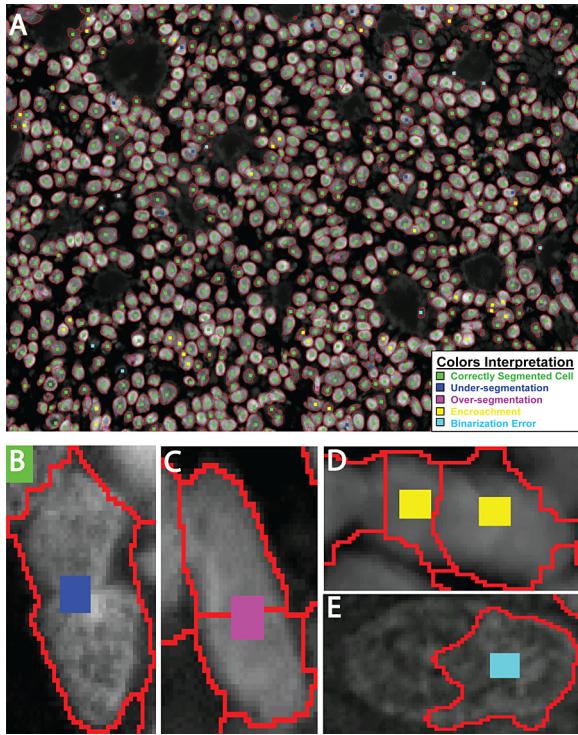


Fig. 5. Illustrating the results validation criteria. (A) Segmentation output of an image with color-coded seeds on each nucleus to identify whether it is correctly segmented or the type of segmentation error. (B) Example of an undersegmentation error. (C) Example of oversegmentation error. (D) Example of an encroachment error. (E) Example of a binarization error.

nuclei. The last type of segmentation errors is binarization errors. This type of error includes the case of nuclei encroaching on their neighbors, or nuclei encroached over by their neighbors. The main difficulty with quantification of encroachment errors is its innate subjectivity. Another difficulty is the acceptance threshold. In a strict sense, one could argue successfully that every adjacent pair of nuclei suffers from some encroachment error. In our work, a slight encroachment of a few pixels that does not change the nucleus shape or size significantly is not considered as an error. We only consider moderate to severe encroachment errors in which the error corresponds to at least 25% of the total nucleus area. Although this manual observation is still subjective and may vary from an observer to another, it can give a good approximation of the number of encroachment errors. Furthermore, we also examined errors from automatic binarization of the image data. The binarization is visualized as boundaries overlaid on the image [see Fig. 5(A)]. Incorrectly binarized nuclei are indicated with light blue dots in Fig. 5(A). We considered errors for which a cell nucleus or part of it is missed at the binarization step.

Finally, Fig. 6 shows six typical examples of segmented nuclear images. All of the scoring results are provided to the reader in the *electronic supplement* (available: www.ecse.rpi.edu/~roysam/TBME-2010-Supp/). Table I summarizes the error analysis using 25 images. The first 15 in the table are *in vitro* images while the last ten are *in vivo*. Overall, just considering under- and oversegmentation errors alone, our

fully automated algorithm achieved >94% accuracy. These data are helpful in comparing our algorithm to previously published methods [9], [18]. When encroachment and binarization errors are included, our algorithm showed an accuracy of more than 86%. The performance of our algorithm with regard to over- and undersegmentation errors can be described in terms of precision and recall measures. Specifically, the last two columns of these values are indicated in Table I. The overall *F*-measure ($2 \times \text{precision} \times \text{recall}) / (\text{precision} + \text{recall})$) for these data is 0.97. We studied the performance of our binarization refinement step by comparing its output with the initial binarization using twenty 2-D phantom images for which we have ground truth data. For each image, we compared the percentages of incorrectly labeled pixels before and after binarization refinement using graph cuts, as detailed in Table II. Fig. 7(A) and (B) shows a sample phantom image and the corresponding ground truth. Initial segmentation output is shown in panel (C), while the refinement output is shown in panel (D). It is clear that significant improvement is achieved after applying graph-cuts refinement.

Finally, we studied the complexity reduction achieved using graph coloring by comparing segmentation processing times with and without graph coloring for 15 automatically created phantom images. All the images have the same size (300×300), with only one connected component (cluster of nuclei), and a varying number of nuclei in each cluster (10–150). Table III shows a summary of the analysis. Increasing numbers of nuclei in the cluster results in rapidly increasing processing time when graph coloring is not used. That is because the number of required α -expansions is equal to the number of nuclei in the cluster. However, no significant increase in processing time is noted when graph coloring is used, since the number of α -expansions is equal to the number of colors, which is in the range of 5 to 10 colors. Three sample phantom images are shown in Fig. 8(A)–(C) containing 10, 70, and 130 nuclei, respectively. The segmentation results are shown as red outlines. A graphical representation of the results in Table III is shown in Fig. 8(D), which shows 2-D plots of the number of cells in the connected component (cluster) versus the processing time for both cases.

IV. DISCUSSION

The present work has built upon, integrated, and extended multiple recent advances in the biological image analysis field. The resulting algorithms have proved to be extremely robust and accurate. In our experience, the usually tricky task of choosing the optimal parameter settings for the proposed algorithm is both simple as well as intuitive. When errors do occur, our method of editing the seeds, followed by segmentation refinement is extremely efficient in practice. It requires minimal effort, and makes best use of the human observer's ability to discern complex patterns, and resolve ambiguities. The actual segmentation is best carried out computationally.

There are several known sources of the errors analyzed by us. Oversegmentation usually happens when a nucleus' chromatin is highly textured (especially true for large nuclei) or when the nucleus shape is extremely elongated. This is particularly common with nuclei that deviate significantly from a

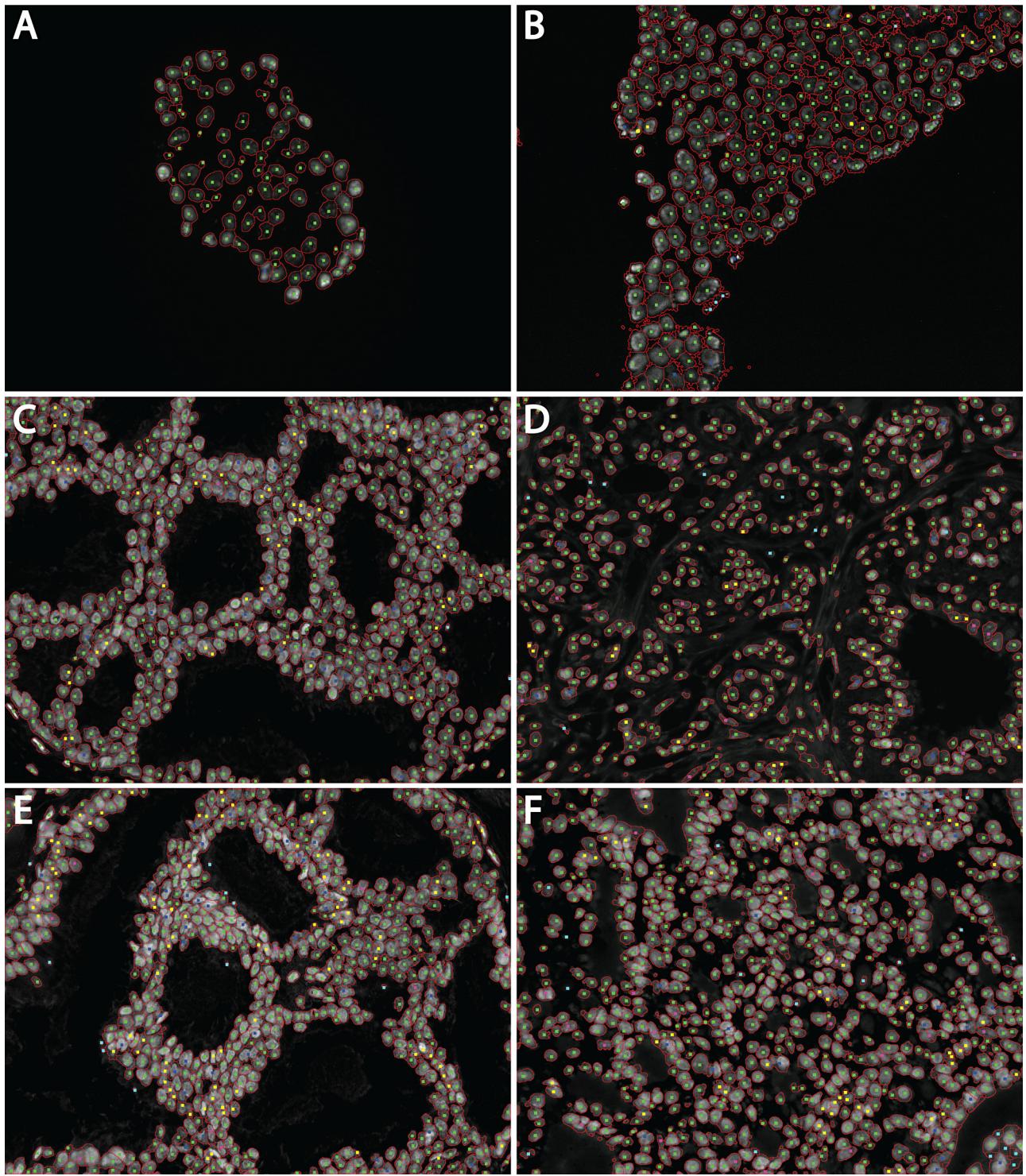


Fig. 6. Sample segmentation results of six 2-D nuclear images including [(A) and (B)] two *in vitro* images and [(C)–(F)] four *in vivo* images.

blob shape, as is the case with some vascular endothelial cells. Undersegmentation usually occurs when nuclei (especially small ones) are highly clustered with weak borders between the nuclei. The causes of encroachment errors were much more diverse, and most often caused by weak object separation cues in the image. The types of errors mentioned earlier are, to some extent, influenced by the choice of parameter settings. This is

discussed further shortly. Binarization errors were largely due to variations in the nuclear signal intensity, specifically, a weak signal resulted in most binarization errors.

A traditional difficulty with automated algorithms is the effort required to tune them by selecting appropriate parameter settings to new images and applications. In this regard, the algorithms we described are well-behaved and intuitive. The main

TABLE I
SUMMARY OF SEGMENTATION PERFORMANCE DATA FOR 25 SAMPLE IMAGES*

Image ID	Number of Cells	Correctly Segmented	Under-Segmented	Over-Segmented	Encroachment Errors	Binarization Errors	Precision	Recall
1	29	27	2	0	0	0	1.00	0.93
2	75	74	0	0	0	1	1.00	1.00
3	154	145	2	2	2	3	0.99	0.99
4	258	232	8	4	10	4	0.98	0.97
5	61	60	1	0	0	0	1.00	0.98
6	98	84	6	2	2	4	0.98	0.93
7	97	96	1	0	0	0	1.00	0.99
8	59	55	1	0	1	2	1.00	0.98
9	53	47	2	1	0	3	0.98	0.96
10	33	33	0	0	0	0	1.00	1.00
11	71	63	3	0	2	3	1.00	0.95
12	34	34	0	0	0	0	1.00	1.00
13	153	148	2	0	1	2	1.00	0.99
14	56	43	6	2	3	2	0.96	0.88
15	156	143	6	2	3	2	0.99	0.96
16	637	487	40	10	89	11	0.98	0.92
17	627	532	33	3	55	4	0.99	0.94
18	375	317	12	11	9	26	0.97	0.96
19	766	644	29	24	50	19	0.96	0.96
20	873	779	25	14	36	19	0.98	0.97
21	641	531	33	17	49	11	0.97	0.94
22	575	491	38	3	37	6	0.99	0.93
23	563	498	11	16	27	11	0.97	0.98
24	608	512	32	12	40	12	0.98	0.94
25	401	344	25	3	17	12	0.99	0.93
Total	7441	6419	312	125	430	155	0.98	0.95
Percentage	86.3%	4.2%	1.7%	5.8%	2.1%			

*All of the images and segmentation results are available as an electronic supplement to facilitate high-resolution viewing.

TABLE II
COMPARISON OF BINARIZATION ACCURACY BEFORE AND AFTER GRAPH-CUT REFINEMENT

Image ID	Error in initial binarization (%)	Error after refinement (%)
1	3.3	1.4
2	5.8	3.0
3	9.0	4.1
4	10.7	3.8
5	13.4	5.1
6	13.4	6.0
7	14.0	6.2
8	13.2	5.5
9	12.8	6.5
10	12.1	5.6
11	12.3	6.2
12	11.5	5.4
13	11.7	6.3
14	12.7	6.1
15	11.2	5.2
16	13.2	6.2
17	11.1	5.3
18	12.4	5.1
19	12.8	6.2
20	13.5	6.4
Average	11.4	5.3

parameters that must be provided to the software include the minimum scale for the LoG filter σ_{\min} , the maximum scale value σ_{\max} , which define the expected range of sizes of the nuclei. In our experiments, we used values in the range of four to eight pixels for σ_{\min} , and 10–20 pixels for σ_{\max} . Although our algorithms are multiscale by design, the choice of these

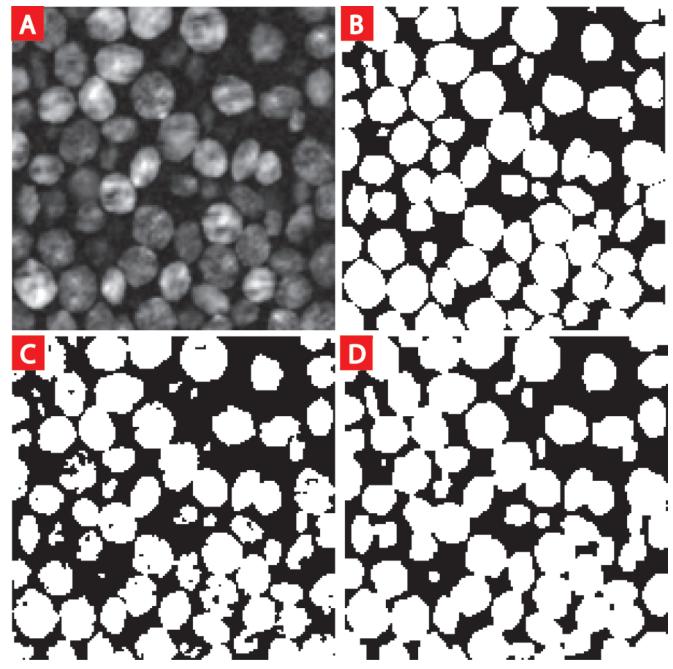


Fig. 7. Comparing initial and graph-cut refined binarization results using a phantom image for which the ground truth is known. (A) 2-D phantom image. (B) Binarization ground truth. (C) Initial binarization output. (D) Results of binarization refinement using graph cuts.

TABLE III
ILLUSTRATING THE COMPLEXITY AND PROCESSING TIME REDUCTION
AFTER USING GRAPH COLORING

Image ID	True number of cells	Number of Detected cells	Number of Colors Used	Segmentation Time (seconds)	
				Without coloring	With coloring
1	10	10	5	5.047	4.782
2	20	20	7	6.235	5.094
3	30	30	6	8.204	5.422
4	40	35	7	7.422	5.281
5	50	47	8	9.031	5.828
6	60	58	9	15.282	6.031
7	70	67	8	19.188	6.844
8	80	79	9	20.516	7.003
9	90	93	8	20.626	6.672
10	100	97	10	27.204	7.547
11	110	104	10	26.001	8.625
12	120	113	8	32.563	7.672
13	130	124	10	37.267	8.235
14	140	137	9	40.032	8.328
15	150	142	9	43.079	8.156

parameters affects the balance of over- and undersegmentation errors to a small extent. Between these two parameters, σ_{\min} is more influential. Specifically, if the value of σ_{\min} is much smaller than the expected minimum size of the nuclei, then the incidence of oversegmentation increases. Smaller values of this parameter are also needed to account for small fragments of nuclei that are characteristic of 2-D sections of 3-D tissue. On the other hand, if the value of σ_{\max} is too low, oversegmentation errors become more prevalent. An overly high value of σ_{\max} is much more benign in nature because it is used in combination with the distance map—it can result in undersegmentation or encroachment errors when exceptionally large and highly clustered groups of nuclei are encountered. The clustering resolution parameter r was generally chosen in the range of 3–5 pixels, and the weighting parameter σ_L for the graph-cuts segmentation algorithm was in the range of 20–30.

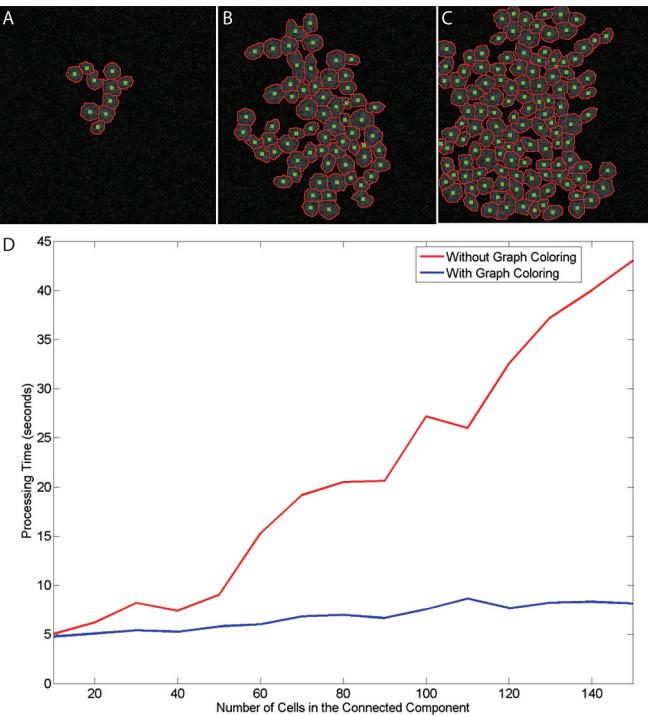


Fig. 8. Illustrating the effect of graph coloring using 15 phantom images with the same size and one nuclear cluster, but with different number of nuclei (x -axis). Three examples are shown in (A)–(C) containing 10, 70, and 130 nuclei, respectively. Detected seeds are shown as green dots and nuclear segmentation results are shown as red outlines. (D) Number of cell nuclei in the cluster versus segmentation processing time (without graph coloring in red and with graph coloring in blue).

The algorithm described here is incorporated into the FARSIGHT toolkit [51] that is designed to analyze multiparameter histopathology images. This software system and implementations of the algorithms reported here are available to interested colleagues from the corresponding author.

ACKNOWLEDGMENT

The authors wish to thank Dr. S. Nath for helpful discussions.

REFERENCES

- [1] C. Bilgin, C. Demir, C. Nagi, and B. Yener, "Cell-graph mining for breast tissue modeling and classification," in *Proc. 29th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. (EMBS 2007)*, pp. 531–5314.
- [2] C. Ortiz de Solorzano, E. Garcia Rodriguez, A. Jones, D. Pinkel, J. W. Gray, D. Sudar, and S. J. Lockett, "Segmentation of confocal microscope images of cell nuclei in thick tissue sections," *J. Microsc.*, vol. 193, no. 3, pp. 212–226, Mar. 1999.
- [3] H. Ancin, B. Roysam, T. E. Dufresne, M. M. Chestnut, G. M. Ridder, D. H. Szarowski, and J. N. Turner, "Advances in automated 3-D image analyses of cell populations imaged by confocal microscopy," *Cytometry*, vol. 25, no. 3, pp. 221–234, Nov. 1, 1996.
- [4] J. Y. Byun, M. R. Verardo, B. Sumengen, G. P. Lewis, B. S. Manjunath, and S. K. Fisher, "Automated tool for the detection of cell nuclei in digital microscopic images: Application to retinal images," *Mol. Vis.*, vol. 12, no. 105–107, pp. 949–960, Aug. 16, 2006.
- [5] M. K. Chawla, G. Lin, K. Olson, A. Vazdarjanova, S. N. Burke, B. L. McNaughton, P. F. Worley, J. F. Guzowski, B. Roysam, and C. A. Barnes, "3D-catFISH: a system for automated quantitative three-dimensional compartmental analysis of temporal gene transcription activity imaged by fluorescence in situ hybridization," *J. Neurosci. Methods*, vol. 139, no. 1, pp. 13–24, Oct. 15, 2004.
- [6] C. O. De Solorzano, R. Malladi, S. A. Lelievre, and S. J. Lockett, "Segmentation of nuclei and cells using membrane related protein markers," *J. Microsc. Oxford*, vol. 201, pp. 404–415, Mar. 2001.
- [7] P. R. Gudla, K. Nandy, J. Collins, K. J. Meaburn, T. Misteli, and S. J. Lockett, "A high-throughput system for segmenting nuclei using multiscale techniques," *Cytometry A*, vol. 73, no. 5, pp. 451–466, May 2008.
- [8] G. Li, T. M. Liu, J. X. Nie, L. Guo, J. Malicki, A. Mara, S. A. Holley, W. M. Xia, and S. T. C. Wong, "Detection of blob objects in microscopic zebrafish images based on gradient vector diffusion," *Cytometry A*, vol. 71A, no. 10, pp. 835–845, Oct. 2007.
- [9] G. Lin, U. Adiga, K. Olson, J. F. Guzowski, C. A. Barnes, and B. Roysam, "A hybrid 3D watershed algorithm incorporating gradient cues and object models for automatic segmentation of nuclei in confocal image stacks," *Cytometry A*, vol. 56, no. 1, pp. 23–36, Nov. 2003.
- [10] G. Lin, M. K. Chawla, K. Olson, C. A. Barnes, J. F. Guzowski, C. Bjornsson, W. Shain, and B. Roysam, "A multi-model approach to simultaneous segmentation and classification of heterogeneous populations of cell nuclei in 3D confocal microscope images," *Cytometry A*, vol. 71, no. 9, pp. 724–736, Sep. 2007.
- [11] 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, no. 1, pp. 20–33, 2005.
- [12] S. K. Nath, F. Bunyak, and K. Palaniappan, "Robust tracking of migrating cells using four-color level set segmentation," *Lect. Notes Comput. Sci. (ACIVS)*, vol. 4179, pp. 920–932, 2006.
- [13] S. K. Nath, K. Palaniappan, and F. Bunyak, "Cell segmentation using coupled level sets and graph-vertex coloring," in *Proc. Med. Image Comput. Comput.-Assisted Intervention (MICCAI)*, 2006, vol. 4190, pp. 101–108.
- [14] B. Parvin, Q. Yang, J. Han, H. Chang, B. Rydberg, and M. H. Barcellos-Hoff, "Iterative voting for inference of structural saliency and characterization of subcellular events," *IEEE Trans. Image Process.*, vol. 16, no. 3, pp. 615–623, Mar. 2007.
- [15] C. Russell, D. Metaxas, C. Restif, and P. Torr, "Using the P^n Potts model with learning methods to segment live cell images," in *Proc. IEEE 11th Int. Conf. Comput. Vis.*, 2007, pp. 1–8.
- [16] C. Wahby, I. M. Sintorn, F. Erlandsson, G. Borgefors, and E. Bengtsson, "Combining intensity, edge and shape information for 2D and 3D segmentation of cell nuclei in tissue sections," *J. Microsc. Oxford*, vol. 215, pp. 67–76, Jul. 2004.
- [17] M. Wang, X. Zhou, F. Li, J. Huckins, R. W. King, and S. T. Wong, "Novel cell segmentation and online SVM for cell cycle phase identification in automated microscopy," *Bioinformatics*, vol. 24, no. 1, pp. 94–101, Jan. 1, 2008.
- [18] G. Li, T. Liu, A. Tarokh, J. Nie, L. Guo, A. Mara, S. Holley, and S. T. Wong, "3D cell nuclei segmentation based on gradient flow tracking," *BMC Cell Biol.*, vol. 8, p. 40, 2007.
- [19] N. Malpica, C. O. de Solorzano, J. J. Vaquero, A. Santos, I. Vallcorba, J. M. Garcia-Sagredo, and F. del Pozo, "Applying watershed algorithms to the segmentation of clustered nuclei," *Cytometry*, vol. 28, no. 4, pp. 289–297, Aug. 1, 1997.
- [20] P. Soille, *Morphological Image Analysis: Principles and Applications*. Berlin, Germany: Springer-Verlag Telos, 1999.
- [21] D. H. Ballard, "Generalizing the hough transform to detect arbitrary shapes," *Pattern Recognit.*, vol. 13, no. 2, pp. 111–122, 1981.
- [22] Q. Yang and B. Parvin, "Perceptual organization of radial symmetries," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2004, pp. 320–325.
- [23] H. Chang, Q. Yang, and B. Parvin, "Segmentation of heterogeneous blob objects through voting and level set formulation," *Pattern Recognit. Lett.*, vol. 28, no. 13, pp. 1781–1787, Oct. 1, 2007.
- [24] Q. Yang and B. Parvin, "Harmonic cut and regularized centroid transform for localization of subcellular structures," *IEEE Trans. Biomed. Eng.*, vol. 50, no. 4, pp. 469–475, Apr. 2003.
- [25] T. Lindeberg, "Feature detection with automatic scale selection," *Int. J. Comput. Vis.*, vol. 30, no. 2, pp. 79–116, Nov. 1998.
- [26] Y. Boykov and G. Funka-Lea, "Graph cuts and efficient N-D image segmentation," *Int. J. Comput. Vis.*, vol. 70, no. 2, pp. 109–131, Nov. 2006.
- [27] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Pattern Anal. Mach.*, vol. 26, no. 9, pp. 1124–1137, Sep. 2004.

- [28] Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images," in *Proc. Int. Conf. Comput. Vision*, 2001, pp. 1025–1112.
- [29] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1222–1239, Nov. 2001.
- [30] V. Kolmogorov and R. Zabih, "What energy functions can be minimized via graph cuts?" *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 147–159, Feb. 2004.
- [31] M. Sezgin and B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," *J. Electron. Imag.*, vol. 13, no. 1, pp. 146–168, Jan. 2004.
- [32] R. Guo and S. M. Pandit, "Automatic threshold selection based on histogram modes and a discriminant criterion," *Mach. Vis. Appl.*, vol. 10, no. 5–6, pp. 331–338, Apr. 1998.
- [33] J. Kittler and J. Illingworth, "On threshold selection using clustering criteria," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-15, no. 5, pp. 652–655, Sep. 1985.
- [34] J. Kittler and J. Illingworth, "Minimum error thresholding," *Pattern Recognit.*, vol. 19, no. 1, pp. 41–47, 1986.
- [35] N. Otsu, "Threshold selection method from Gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-9, no. 1, pp. 62–66, Jan. 1979.
- [36] J. Lie, M. Lysaker, and X. C. Tai, "A binary level set model and some applications to Mumford-Shah image segmentation," *IEEE Trans. Image Process.*, vol. 15, no. 5, pp. 1171–1181, May 2006.
- [37] N. R. Pal and S. K. Pal, "Image model, Poisson distribution and object extraction," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 5, no. 3, pp. 459–483, 1991.
- [38] J. L. Fan, "Notes on Poisson distribution-based minimum error thresholding," *Pattern Recognit. Lett.*, vol. 19, no. 5–6, pp. 425–431, Apr. 1998.
- [39] N. R. Pal and D. Bhandari, "On object background classification," *Int. J. Syst. Sci.*, vol. 23, no. 11, pp. 1903–1920, Nov. 1992.
- [40] Y. Boykov, O. Veksler, and R. Zabih, "Markov random fields with efficient approximations," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 1998, pp. 648–655.
- [41] D. Geiger, A. Gupta, L. A. Costa, and J. Vlontzos, "Dynamic programming for detecting, tracking, and matching deformable contours (vol 17, pg 294 1995)," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 5, pp. 575–575, May 1996.
- [42] J. Gill, H. Breu, D. Kirkpatrick, and M. Werman, "Linear time Euclidean distance transform algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 5, pp. 529–533, May 1995.
- [43] O. Cuisenaire, "Fast Euclidean distance transformations by propagation using multiple neighbourhoods," *Comput. Vis. Image Understanding*, vol. 76, no. 2, pp. 163–172, Nov. 1999.
- [44] R. W. Mackin, B. Roysam, T. J. Holmes, and J. N. Turner, "Automated three-dimensional image analysis of thick and overlapped clusters in cytologic preparations. Application to cytologic smears," *Anal. Quant. Cytol. Histol.*, vol. 15, no. 6, pp. 405–417, Dec. 1993.
- [45] R. W. Mackin, Jr., L. M. Newton, J. N. Turner, and B. Roysam, "Advances in high-speed, three-dimensional imaging and automated segmentation algorithms for thick and overlapped clusters in cytologic preparations. Application to cervical smears," *Anal. Quant. Cytol. Histol.*, vol. 20, no. 2, pp. 105–121, Apr. 1998.
- [46] B. Roysam, H. Ancin, A. K. Bhattacharjya, M. A. Chisti, R. Seegal, and J. N. Turner, "Algorithms for automated characterization of cell populations in thick specimens from 3-D confocal fluorescence microscopy data," *J. Microsc.*, vol. 173, no. 2, pp. 115–126, Feb. 1994.
- [47] X. W. Wu, Y. D. Chen, B. R. Brooks, and Y. A. Su, "The local maximum clustering method and its application in microarray gene expression data analysis," *EURASIP J. Appl. Signal Process.*, vol. 2004, no. 1, pp. 53–63, Jan. 1, 2004.
- [48] 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.
- [49] N. Robertson, D. P. Sanders, P. Seymour, and R. Thomas, "Efficiently four-coloring planar graphs," in *Proc. 28th Annu. ACM Symp. Theory Comput.*, 1996, pp. 571–575.
- [50] D. Baggett, M. A. Nakaya, M. McAuliffe, T. P. Yamaguchi, and S. Lockett, "Whole cell segmentation in solid tissue sections," *Cytometry A*, vol. 67, no. 2, pp. 137–143, Oct. 2005.
- [51] C. S. Björnsson, G. Lin, Y. Al-Kofahi, A. Narayanaswamy, K. L. Smith, W. Shain, and B. Roysam, "Associative image analysis: a method for automated quantification of 3D multi-parameter images of brain tissue," *J. Neurosci. Methods*, vol. 170, no. 1, pp. 165–178, May 15, 2008.



Yousef Al-Kofahi received the B.S degree in computer engineering from Yarmouk University, Irbid, Jordan, in 2003, and the M.S. degree in 2005 from the Electrical, Computer and Systems Engineering (ECSE) Department, Rensselaer Polytechnic Institute (RPI), Troy, NY, where he is currently working toward the Ph.D. degree in computer systems engineering.

He is currently engaged in developing computer vision and machine-learning algorithms for object-level mapping of complex biological tissues from 2-D/3-D multispectral images. His current research interests include computer vision, pattern recognition, and image processing/analysis.



Wiem Lassoued received the Master's degree in sciences from the University of Sciences of Tunis, Tunis, Tunisia, in 2001.

She has been a Specialist Researcher C with Dr. William's Lee Laboratory, Abramson Cancer Center, University of Pennsylvania, Philadelphia, PA, for the past 5 years. Her current research interests include development of computerized quantitative analysis of histopathology tumor specimen based on image analysis, and look at different mechanisms involved to develop cancer.



William Lee received the Ph.D. and M.D. degrees from the University of Chicago, Chicago, IL, in 1974 and 1975, respectively.

He is currently an Associate Professor of medicine with the Division of Hematology–Oncology, University of Pennsylvania, Philadelphia, PA, where he also the Co-leader of the Tumor Biology Program in the Abramson Cancer Center. He is an Associate Editor of the *Journal of Clinical Investigation*. His current research interests include development of methods for quantitative analysis of human tumor histopathology and the biological events and processes preserved in these specimens.



Badrinath Roysam (SM'89) received the B.Tech. degree in electronics engineering from the Indian Institute of Technology Madras, Chennai, India, in 1984, and the M.S. and D.Sc. degrees from Washington University, St. Louis, in 1987, and 1989, respectively.

Since 1989, he has been with the Rensselaer Polytechnic Institute, Troy, New York, where he is currently a Professor in the Electrical, Computer and Systems Engineering Department. He is an Associate Director of the Center for Subsurface Sensing and Imaging Systems (CenSSIS)—a multiuniversity National Science Foundation (NSF) sponsored engineering research center, and the Co-director of the Rensselaer Center for Open Source Software. He also holds an appointment in the Biomedical Engineering Department. His current research interests include 2-D, 3-D, and 4-D biomedical image analysis, biotechnology automation, optical instrumentation, high-speed and real-time computing architectures, and parallel algorithms.

Dr. Roysam is a Senior Member the Microscopy Society of America, the International Society for Analytical Cytology, the Society for Neuroscience, and the Association for Research in Vision and Ophthalmology. He is an Associate Editor for the *IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING* and the *IEEE TRANSACTIONS ON INFORMATION TECHNOLOGY IN BIOMEDICINE*.