Nucleus Segmentation Using Gaussian Mixture based Shape Models

Hyun-Gyu Lee and Sang-Chul Lee

***Abstract*—We identify cells in microscopy images with stained nuclei, using the following process: Candidate seeds for nuclei are identified as extrema in a Laplacian-of-Gaussian space, and weak candidates are eliminated from clusters obtained by ellipse fitting; a region of interest for each nucleus is then defined by combining local and global thresholding; and these regions are repeatedly merged and split by modeling the shape of a nucleus and measuring the roughness of the shared boundaries connected nuclei. This method showed superior abilities to detect the nucleus regions and to split the boundaries of connected nuclei. Our experiments show higher scores in comparison with five other techniques in terms of eight evaluation metrics.**

***Index Terms*—Gaussian mixture model, nucleus segmentation, reconstructed nuclei, ROI extraction, splitting nuclei.**

I. Introduction

AUTOMATIC methods of cell counting and tracking are important in cytometric analyses that classify cell types or analyze cell structure. Different microscopy imaging techniques and cell types require different segmentation methods, and these can be challenging to design, due to the wide variety of cell sizes and shapes, as well as noisy images of variable brightness. In particular, the boundaries between adjacent cells are sometimes ambiguous: even experts may be unable to decide whether a questionable object is a single cell or two overlapping cells.

Cells are typically detected using an intensity-based segmentation, such as Otsu's method [1] or a watershed transformation [2]. Recently, these two methods have been combined [3], but even when combined, these methods produce poor segmentations when there are local variations in contrast, noise, and a high density of overlapping cells, which occurs in microscopy images of stained nuclei. Energy minimization can yield better results in such conditions. The active contour model (ACM) [4] is the basis of two representative energy-minimization approaches: region-based [5] and edge-based [6] segmentation. Both of these methods are influenced by the accuracy of the initial boundaries, but more advanced energy-minimization algorithms, based on gradient flow tracking [7], [8], graph cut [9], [10], level sets [11], as well as the graph-based modular interactive nuclear segmentation (MINS) [12] method, produce promising segmentation results without requiring accurate initialization.

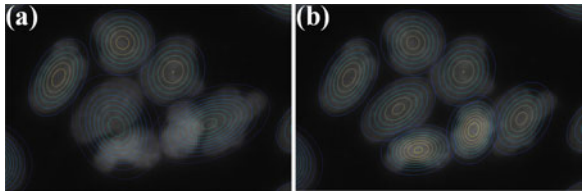

Fig. 1. Model parameters of nuclei identified by (a) merge and (b) split of nucleus shapes fitted by Gaussian mixture models.

Recently, machine learning has been applied to cell segmentation, using various methods of feature selection and designs of classifiers. Ilastik presented a method to use the features of each pixel's neighborhood in a learning step which builds a random forest classifier [13]. Supervised learning [14] can split touching cells in histopathological images using color-texture features obtained by local Fourier transforms. Detecting cells through binary classification by a support vector machine [15] is applicable to a range of datasets, including histopathological images of breast cancers, fluorescence images of HEK cells and phase-contrast images of HeLa cells. Bayesian and Kalman inference [16] also perform well in cell segmentation and tracking. Many learning-based approaches are satisfactory, but they all require ground-truth images for training.

Despite these advances, it remains challenging to resolve ambiguous boundaries between adjoining cells, and this also affects the availability of ground truth for training. We propose a new method for the segmentation of stained nuclei, based on the merge and split of Gaussian mixture based shape model (GMSM), as shown in Fig. 1. Based on nucleus blob detection [9], [17], we find a point in each cell, and then split cells by using an expectation-maximization (EM) [18] clustering method that identifies plausible boundaries. This method makes two significant contributions to cell segmentation:

1) Reliable splitting of a region of interest (ROI), extracted by seed-based integrated thresholding, using merge and split of nucleus shapes fitted by GMSM, even when the number of components in the mixture model fails to match the number of actual nuclei.

Manuscript received August 3, 2016; revised January 10, 2017 and April 19, 2017; accepted April 24, 2017. Date of publication May 2, 2017; date of current version January 3, 2018. This work was supported by Inha University Research Grant. *(Corresponding author: Sang-Chul Lee.)*

The authors are with the Department of Computer Science and Engineering, Inha University, Incheon 22212, Korea (e-mail: hg_lee@inha.edu; sclee@inha.ac.kr).

Digital Object Identifier 10.1109/JBHI.2017.2700518

2) Estimation of boundaries between adjoining nuclei, which produces boundaries close to the ground truth obtained by human inspection. This automated estimation is based on differences in intensity between the reconstructed nucleus and the original nucleus, and roughness of the estimated boundaries.

The remainder of the paper deals with several topics. In Section II we introduce our method of seed detection, which combines two processes: 1) extrema detection by finding the minima in a Laplacian-of-Gaussian (LoG) space; and 2) weak seed suppression by ellipse fitting. In Section III we introduce ROI extraction and show how it is integrated with global and local thresholding. In Section IV we describe nucleus division using three processes: 1) feature description based on the ascent of intensity gradients; 2) ROI clustering by GMSM; and 3) the merge and split of these models. In Section V we present experimental results from two microscopy image datasets, containing 758 images and 48 images respectively, and compare them with results from five other methods, using eight metrics. In Section VI we conclude the paper and outline plans for future work.

II. Seed Detection

A. Extrema Detection in LoG Space

Nucleus segmentation begins with seed-points, and therefore segmentation algorithms [9], [17] require reliable seed detection, because the seeds identify the approximate locations and scales of the nuclei. Following previous work [9], [17], we use a scale-normalized blob detector [19] in LoG space to find extrema, which become candidate seeds. We define a candidate seed $\hat{\mathbf{x}} = (x, y)$ as the position of a minimum in the LoG space $\mathrm{LoG}(x, y, \sigma)$. This minimum is actually found in a discrete 3D LoG space, in which the spatial x- and y-axes are discretized into pixels of unit size; and the third axis is scale σ, which is discretized into increments of σ_{step}. A candidate seed is located at a voxel which has a lower value than its 26 neighbors. Values of scale range from $\sigma_{\min}$ to $\sigma_{\max}$, and these limits are determined from the expected radius of a nucleus, such that $r = \sqrt{2}\sigma$. This blob detector can find candidate seeds close to the centers of circular objects and can also determine their approximate scales.

Algorithm 1: Iterative Ellipse Fitting.

procedure ELLIPSE_FITTING($\hat{\mathbf{x}}, \Lambda_{\hat{\mathbf{x}}}$)

Variable description and initialization:

N_{iter} : the number of iterations

r^1: radius $\sqrt{2}\sigma$ for the scale of $\hat{\mathbf{x}}$

δ_ε: tolerance for termination

w: a weight for ellipse expansion

$$\mathbf{R}(\theta) = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}$$

Ellipse fitting on the extreme:

for $k \leftarrow 1, N_{\mathrm{iter}}$ **do**

$\mathbf{x}^k = \{\mathbf{x} \in \Lambda_{\hat{\mathbf{x}}} : \|\mathbf{x} - \hat{\mathbf{x}}\|_2 < r^k\}$

$\mathbf{x}_{\mathrm{c}}^k = \mathbf{x}^k - \bar{\mathbf{x}}^k, x_{\mathrm{c}}, y_{\mathrm{c}} \in \mathbf{x}_{\mathrm{c}}^k$

if $k = 1$ then $\alpha = 1, \beta = 1$

$$\mathbf{x}_{\mathrm{e}}^k \in \mathbb{Z}^{n\times 2} = \left\{ x, y \in \mathbf{x}_{\mathrm{c}}^k : \left\| \mathbf{x}_{\mathrm{c}}^{\mathbf{k}} \right\| < \left\| \mathbf{R}(\theta) \begin{bmatrix} w\alpha\cos(\theta) \\ w\beta\sin(\theta) \end{bmatrix} \right\|, \theta = \tan^{-1}\left(\frac{y}{x}\right) \right\}$$

$$\mathbf{C}_{n\times 5} = \begin{bmatrix} x_1^2 & x_1y_1 & y_1^2 & x_1 & y_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n^2 & x_ny_n & y_n^2 & x_n & y_n \end{bmatrix}$$

$[a, b, c, d, e] = (n\mathbf{C}^{\mathrm{T}}\mathbf{C})^{-1} \sum_{i=1}^{n} \mathbf{C}_{i\times 5}$

$\theta_{\mathrm{e}} = \frac{1}{2}\tan^{-1}\left(\frac{b}{c-a}\right)$

$$\begin{bmatrix} a' \\ c' \\ d' \\ e' \end{bmatrix} = \begin{bmatrix} \cos^2\theta_{\mathrm{e}} & \sin^2\theta_{\mathrm{e}} & 0 & 0 \\ -\cos\theta_{\mathrm{e}}\sin\theta_{\mathrm{e}} & \cos\theta_{\mathrm{e}}\sin\theta_{\mathrm{e}} & 0 & 0 \\ \sin^2\theta_{\mathrm{e}} & \cos^2\theta_{\mathrm{e}} & 0 & 0 \\ 0 & 0 & \cos\theta_{\mathrm{e}} & \sin\theta_{\mathrm{e}} \\ 0 & 0 & -\sin\theta_{\mathrm{e}} & \cos\theta_{\mathrm{e}} \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} a \\ b \\ c \\ d \\ e \end{bmatrix}$$

$F = 1 + \frac{d'^2}{4a'} + \frac{e'^2}{4c'}$

$\alpha^k = \sqrt{\frac{F}{a'}}, \beta^k = \sqrt{\frac{F}{c'}}$

$r^{k+1} = r^k + 2(\alpha^k - \alpha^{k-1})$

if $|\alpha^k - \alpha^{k-1}| < \delta_\varepsilon$ then **exit for**

end for

return $\{\mathbf{R}(\theta_{\mathrm{e}}), \alpha^k, \beta^k\}$

end procedure

B. Weak Seed Suppression Using Ellipse Fitting

Many candidate seeds are typically found in an image, some of which may lie in the same nucleus. We select the seed that is closest to the center of a good representation of nucleus and the scale of that nucleus, using weak seed suppression. This eliminates seeds with larger LoG values from groups of seeds with overlapping radii. The candidate seeds are selected by the scale-normalized blob detector, on the basis that they are the centers of circular objects; but the stained nuclei in microscopy images are generally closer to ellipse shapes.

We estimate the parameters $\mathbf{R}$, α, β of an ellipse by least-squares fitting to the pixels in an object that may be a cell, where $\mathbf{R}$ is a 2×2 rotation matrix that determines the orientation of the ellipse, and α and β are the lengths of its major and minor axes. This method of fitting an ellipse to an object within a ROI is summarized as Algorithm 1. It grows a circular disk with the radius of the initial scale obtained from (1), up to the nucleus boundary, which is assumed to be an elliptical, and then finds the parameters of the ellipse which fits that nucleus [Fig. 2(a) and (b)]. $\Lambda_{\hat{\mathbf{x}}}$ is a set of positions of the ROI near the candidate seed $\hat{\mathbf{x}}$, which is used as the foreground for ellipse fitting. Determination of the ROI is described in Section III. Using the ellipse parameters estimated by Algorithm 1,we cluster candidate seeds which correspond to overlapping ellipses. A cluster C containing $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{x}}_j$ is de-

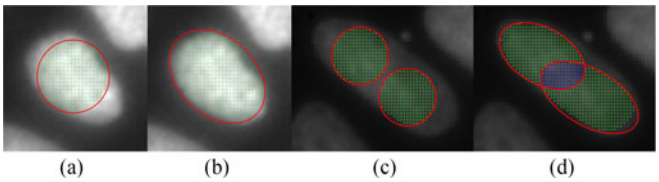

Fig. 2. Examples of (b) fitting an ellipse to an object from (a) an initial scale; (c) two objects which appear to be disjoint when fitted by circles, but (d) overlap when fitted by ellipses.

fined as follows:

$$C = \{\hat{\mathbf{x}}_i, \hat{\mathbf{x}}_j | D(\theta_{ij}) > \|\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_j\|_2,\ i \neq j\},$$
$$D(\theta_{ij}) = \left\| \mathbf{R_i} \begin{bmatrix} \alpha_i \cos(\theta_{ij}) \\ \beta_i \sin(\theta_{ij}) \end{bmatrix} \right\| + \left\| \mathbf{R_j} \begin{bmatrix} \alpha_j \cos(\theta_{ij}) \\ \beta_j \sin(\theta_{ij}) \end{bmatrix} \right\|, \quad (1)$$

where $D(\theta_{ij}) > \|\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_j\|_2$; θ_{ij} is the angle between $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{x}}_j$ as $\tan^{-1} \frac{\hat{y}_i - \hat{y}_j}{\hat{x}_i - \hat{x}_j}$; $D(\theta_{ij})$ is the sum of the radii of the ith and jth ellipses in the direction determined by θ_{ij}.

If two candidate seeds are located in a single nucleus, their ellipses may overlap [the blue area in Fig. 2(d)], and the seed with the larger local minimum is discarded between the two seeds in this case. Finally, we select the seed $\tilde{\mathbf{x}}$ with the minimum value in LoG space among all the seeds in cluster C.

III. Extraction of the Regions of Interest

One of the characteristics of microscopy images with stained nuclei is that there are many areas of black background that contain no nuclei. Our ellipse fitting and nucleus division processes need to be applied to nucleus area, and therefore we have to extract a region of interest (ROI) corresponding to this area. These ROIs have various intensity distributions depending on capture techniques and cell types. The intensity ranges of the nuclei and background areas must be normalized before we identify the ROI of the nucleus regions. We use one of two linear normalization processes, depending on the datasets: 1) normalization based on the maximum intensity of each image $\left(\tilde{I}(\mathbf{x}) = I(\mathbf{x})/\max(I)\right)$; or 2) average normalization based on the average intensity of an image from which the background has been eliminated. The background can be identified by the large difference between adjacent bars in the image histogram (see Algorithm 2). δ_H is threshold for difference between frequencies of ith and $(i-1)^{\text{th}}$ categories. By this threshold, we can find the intensity where histogram frequencies are little changed from. Average normalization is effective for a wide range of foreground intensities, even when these are close to the background intensities.

After normalization, the ROI is separated from the image by a combination of global thresholding and local thresholding around the seeds, using Otsu's method [1]. The binary images of the global ROI Ψ_{global} and a local ROI Ψ_{local} are produced using global thresholding δ_{global} and local thresholding δ_{g_i}

Algorithm 2: Background-eliminated Normalization.

procedure AVERAGE_NORMALIZATION(I)
Variable description:
H : An intensity histogram with n bins for image I, $\mathbb{N}^{1\times n}$
H_{s} : A histogram sorted by H descending, $\mathbb{N}^{1\times n}$
i_{s} : An index vector of H, $\mathbb{N}^{1\times n}$
Foreground selection:
$\{H_{\text{s}}, i_{\text{s}}\} = \text{sort}(H)$,
for $i \leftarrow 1, n$ **do**
$D(i) = H_{\text{s}}(i) - H_{\text{s}}(i+1)$,
end for
$i_{\text{f}} = \min(i)\ \text{s.t.}\ D(i) < \delta_H$,
Average normalization:
$H'_{\text{s}} = \frac{H_{\text{s}}(i_{\text{f}}:n)}{\sum_{i=i_{\text{f}}}^{n} H_{\text{s}}(i)}$
$\tilde{I} = \frac{I}{\sum_{i=i_{\text{f}}}^{n} i_{\text{s}}(i) H'_{\text{s}}(i-i_{\text{f}}+1)}$
return $\tilde{I}$
end procedure

respectively, as follows:

$$\Psi_{\text{global}}(x,y) = \begin{cases} 1, \text{if } G(x,y,\sigma) > \delta_{\text{global}} \\ 0, \text{otherwise} \end{cases},$$
$$\Psi_{\text{local}}(x,y) = \sum_{i=1}^{n} \begin{cases} 1, \text{if } g_i(x,y,\sigma) > \delta_{g_i} \\ 0, \text{otherwise} \end{cases}, \quad (2)$$
$$\text{s.t. } \hat{x}_i - r_i \leq x \leq \hat{x}_i + r_i,$$
$$\hat{y}_i - r_i \leq y \leq \hat{y}_i + r_i, g_i \subseteq G, (\hat{x}_i, \hat{y}_i) \in \hat{\mathbf{x}}_i,$$

where δ_{global} is the global threshold produced by applying Otsu's method to a Gaussian image G with a scale σ. Each local threshold δ_{g_i} is produced by applying Otsu's method to a subset g_i of a square image containing $4r_i^2$ pixels, centered on $\hat{\mathbf{x}}_i$, in the Gaussian image G. The value of r_i is determined from by the scale of the ith seed.

We can obtain an approximate ROI from the whole image by thresholding. However, parts of this ROI may actually be background, due to the low contrast between the nuclei and background. Thresholding using local contrast values can correct most of these errors. The final ROI resulting from both global and local thresholding can be expressed as follows:

$$\Psi(x,y) = \begin{cases} 1, \text{if } \Psi_{\text{blur}}(x,y) > \delta \\ 0, \text{otherwise} \end{cases}, \quad (3)$$
$$\Psi_{\text{blur}} = (\Psi_{\text{local}} + \Psi_{\text{global}}) * B(\sigma),$$

where $B(\sigma)$ is the Gaussian kernel $(B(\sigma) \sim \mathcal{N}(0,\sigma))$ that produces a smooth boundary, by a combination of local and global thresholding [see Fig. 3 (d)]. The threshold δ determines the extent of the ROI. A high threshold produces a smaller ROI.

The set of pixels $\{(x,y) : \Psi(x,y) = 1\}$, which we call Λ, is the extent of the ROI. The candidate seed $\hat{\mathbf{x}}$ is used for local thresholding of Ψ_{local} to extract the ROI Λ for ellipse fitting; the

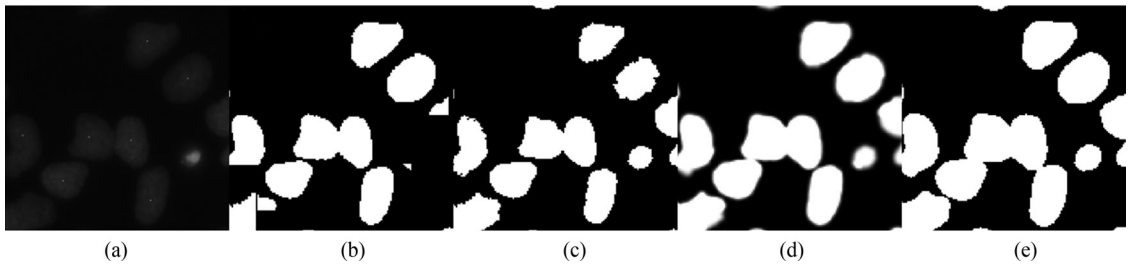

Fig. 3. Stages in the determination of a ROI: (a) the original image with seeds (red), (b) the image after local thresholding, (c) the image after global thresholding, (d) the combination of images (b) and (c), and (e) the final ROI with δ of 0.25.

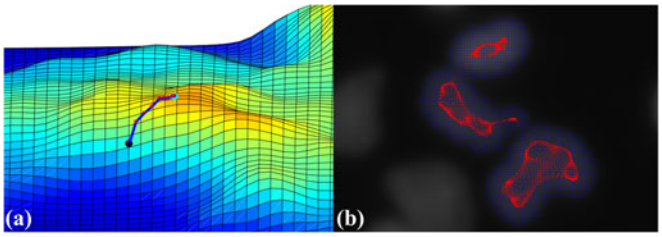

Fig. 4. (a) An example of the gradient ascent method, and (b) examples of a position $\mathbf{x}$ (blue) of a ROI and the position $\mathbf{x}_\nabla$ (red) of its local maximum.

final seed $\tilde{\mathbf{x}}$ is used for ROI to be used for nucleus division. Since the final seed $\tilde{\mathbf{x}}$ is not known during weak seed suppression, $\hat{\mathbf{x}}$ is only used in creating the ROI, which is then used for suppression. This ROI produced by $\tilde{\mathbf{x}}$ is used to split the GMSM. The stages in determining a ROI are shown in Fig. 3.

IV. Nucleus Division

The ROI is split into several nuclei based on the assumptions that the shapes of stained nuclei follow a multivariate Gaussian distribution, and that the shape of the ROI of the adjoining nuclei follows a GMSM. In this section, we introduce the concept of features in a GMSM, and the EM-based merge and split methods that we use to find K plausible components of a GMSM.

A. Feature Description

We consider each pixel in an ROI Λ, which is created by $\tilde{\mathbf{x}}$, to be a feature of a GMSM. When this probability is obtained from a GMSM, and a pixel is closer to the center of another nucleus than that of its own nucleus, the probability that the pixel belongs to the other nucleus may be higher than the probability that the pixel belongs to its own nucleus. These some pixels may be erroneously classified as belonging to the wrong nuclei.

We address this misclassification problem using the method of gradient ascent shown in Fig. 4(a). Starting from each position $\mathbf{x}$ in the ROI, the method ascends to a local maximum through the pixels on a path expressed by $\mathbf{x}^{k+1} = \mathbf{x}^k + \frac{r_{\text{near}}}{2} \nabla \mathrm{G}(\mathbf{x}^k)$, where $\nabla \mathrm{G}(\mathbf{x}^k)$ is the gradient at $\mathbf{x}^k$ in the Gaussian image G, k is the index of the iteration, and r_{near} is a radius obtained from the scale of the seed nearest to the position of the initial $\mathbf{x}$. Ascent terminates when the Euclidean distance between $\mathbf{x}^k$ and $\mathbf{x}^{k+1}$ converges within a tolerance the position $\mathbf{x}_\nabla$ of the local maximum of $\mathbf{x}$ is obtained. We expect that $\mathbf{x}_\nabla$ will be closer to the center of its own nucleus than a pixel $\mathbf{x}$ which is located near to a boundary between nuclei. By moving each seed to the center of its own nucleus, we can rectify the misclassification that is otherwise caused by elliptical nuclei.

Fig. 4(b) shows how the $\mathbf{x}_\nabla$ pixels near boundaries are closer to the nuclear centers than the pixels $\mathbf{x}$. However, since the positions of the $\mathbf{x}_\nabla$ pixels will be very different from the original shape of the nucleus, the shape of the GMSM obtained using $\mathbf{x}_\nabla$ alone may be distorted by the positions of the nuclei in the original image. To preserve the information in the original shape, we combine each pixel $\mathbf{x}$ in an ROI and the corresponding local maximum $\mathbf{x}_\nabla$ into a single feature $\Phi = [\mathbf{x}, \mathbf{x}_\nabla]^{\mathrm{T}}$ of the GMSM.

B. Clustering by Gaussian Mixture Based Shape Models

We optimize Φ for each segment of the ROI using the expectation-maximization (EM) method [18], which repeatedly finds a *maximum a posterior* (MAP) distribution. Our EM method involves initialization, followed by the E and the M steps. During initialization, the number of components K in the mixture model is changed by the merge and split processes. Each feature Φ is initially allocated to a class determined by the K-means algorithm [20], in which the position of the final seed $\tilde{\mathbf{x}}$ is used as the center of each of K clusters, as shown in Fig. 5(a). The mean μ and covariance Σ of the features in each class can then be obtained, and these are used as the parameters θ of the initial mixture model. The prior probability $P(c)$ is the proportion of all features that are in each class c.

In the E step, the expected MAP of all features in each class is updated using the current estimated parameters θ, as follows:

$$P(c_j|\Phi_i) = \frac{P(c_j)P(\Phi_i|\theta_j)}{\sum_{k=1}^{K} P(c_k)P(\Phi_i|\theta_k)}, \tag{4}$$

where, we assume, the likelihood follows a multivariate Gaussian distribution, i.e., $P(\Phi_i|\theta_j) \sim N(\mu, \Sigma)$. The class of each feature is then updated to the most probable class.

In the M step, the values of $P(c)$ and θ are updated to maximize the expected MAP, as follows:

$$P(c_j) = \frac{\sum_{i=1}^{N_\Phi} P(c_j|\Phi_i)}{\sum_{j=1}^{K}\sum_{i=1}^{N_\Phi} P(c_j|\Phi_i)}, \mu_j = \frac{\sum_{i=1}^{N_\Phi} P(c_j|\Phi_i)\Phi_i}{\sum_{i=1}^{N_\Phi} P(c_j|\Phi_i)},$$
$$\Sigma_j = \frac{\sum_{i=1}^{N_\Phi} P(c_j|\Phi_i)(\Phi_i - \mu_j)^{\mathrm{T}}(\Phi_i - \mu_j)}{\sum_{i=1}^{N_\Phi} P(c_j|\Phi_i)}, \tag{5}$$

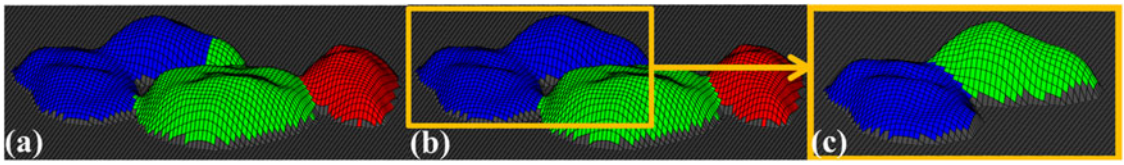

Fig. 5. Clustering using (a) K-means, (b) partitioning of the Gaussian mixture model of (a), and (c) split of (b). Each color shows a different class and $K = 3$.

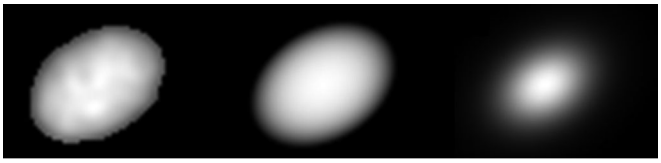

Fig. 6. (a) An original nucleus, and (b) a nucleus reconstructed using (7), and (c) a nucleus reconstructed using the normalized Gaussian distribution.

where N_Φ is the total number of features, and K is the number of classes, which is also the number of seeds.

C. Merge and Split of Gaussian Mixture Based Shape Models

It is difficult to find the best seeds in small merged regions, which contain many nuclei. If the result of seed detection is incomplete and the number of seeds is used as the number of the components K of the GMSM, then the parts of the GMSM produced by the procedure described in Section IV.B will be inaccurate. In this section, we solve this problem by merge and split operations that find a more accurate value of K by analysis of the similarity between the reconstructed nuclei and the regions connected by the original nuclei.

During merge, GMSM with K components, from 1 to the number of $\tilde{\mathbf{x}}$ seeds, are considered in the unlabeled local region. When K is less than the number of seeds n extracted from the local region, all combinations of the remaining seeds are used as initial seeds. Therefore, we have $\sum_{K=1}^{n} \frac{n!}{K!(n-K)!}$ candidate GMSMs in the ROI of the adjacent nuclei contain the n detected seeds. To find the best value of K among the candidate GMSMs, a reconstructed nucleus V is created for each model parameter θ of the GMSM. We transform the intensity away from the center of a reconstructed nucleus using a whitening transformation in order to produce intensities, of the reconstructed nucleus, similar to those on the surface of the original nucleus. The reconstructed nucleus is more similar to the original nucleus than that produced by the general Gaussian distribution, as shown in Fig. 6. We define a cost function to express the error between the original nucleus G and the reconstructed nucleus V generated by the parameter of the GMSM, as follows:

$$E = \sum_{i=1} |\mathbf{V}(\mathbf{x}_i, \Theta) - G(\mathbf{x}_i, \sigma)| + \lambda \sum_{j=1} \big(v_j - f(u_j, \boldsymbol{\beta})\big),$$

$$\mathbf{V}(\mathbf{x}_i, \Theta) = \max_{\theta_k} \big(V(\mathbf{x}_i, \theta_k)\big), \theta_k \in \Theta,$$

$$\text{s.t. } 1 \le k \le K \le n, \tag{6}$$

where $\mathbf{V}(\mathbf{x}_i, \Theta)$ is the maximum intensity at each position in k reconstructed nuclei, produced by model parameters Θ of GMSM, $G(\mathbf{x}_i, \sigma)$ is an intensity of the Gaussian image produced from the original image at each position, under the condition that $P(\mathbf{x}_c|\theta) < t$, and $\boldsymbol{\beta}$ is the coefficient set which fit points of connected boundary. Hence, f is $\beta_0 + \beta_1 u + \beta_2 u^2$ as a quadratic polynomial curve fitted by a least-squares procedure. u and v are positions indicating the shared boundary of two connected nuclei. λ adjusts the weights between the first term, and control the error between the original shape and the reconstructed shape; and the second term controls the roughness of the boundaries. By finding the value of K that minimizes (6), we can determine a value of K that is close to the actual number of nuclei.

The reconstructed nucleus $V(\mathbf{x}_i, \theta)$ is the intensity obtained by the normalized exponent corresponding to the distance $\|\mathbf{x}_c \mathbf{W}\|$, as follows:

$$V(\mathbf{x}, \theta) = \frac{\max_{\mathbf{x}_c} \left(e^{2\|\mathbf{x}_c \mathbf{W}\|}\right) - e^{2\|\mathbf{x}_c \mathbf{W}\|}}{\max_{\mathbf{x}_c} \left(e^{2\|\mathbf{x}_c \mathbf{W}\|}\right) - \min_{\mathbf{x}_c} \left(e^{2\|\mathbf{x}_c \mathbf{W}\|}\right)} + \delta_t,$$

$$\text{s.t. } P(\mathbf{x}_c|\theta) > \delta_t,$$

$$\delta_t = \underset{0<t\le 1}{\operatorname{argmin}} \left| \left(\sum_{\mathbf{x}_c} P(\mathbf{x}_c|\theta) \right) - \gamma \right|, \text{ s.t. } P(\mathbf{x}_c|\theta) < t, \tag{7}$$

where $P(\mathbf{x}_c|\theta)$ is a multivariate Gaussian distribution, θ are the parameters of the GMSM separated from ROI Λ, and $\mathbf{x}_c$ is the distance in pixels between the given $\mathbf{x}$ and μ in θ. The term δ_t is the minimum probability of $P(\mathbf{x}_c|\theta)$ obtained from γ that determines the size of the reconstructed nucleus, and $\mathbf{W}$ is a 2×2 whitening matrix used in the whitening transformation.

In the split process, we assume that each of the K components in a GMSM determined by the merge process is in a new GMSM with K components. Each component of the original GMSM produced by the merge process is split into between 1 and $\Omega/\Omega_{\min}$ components, where Ω is the number of samples of the component, and $\Omega_{\min}$ is the minimum size of a nucleus obtained from the dataset to be used in our experiments. Finally, an improved GMSM with K' components is created as described above.

These merge and split processes are implemented locally in each segment that consists of several nuclei.

V. Experimental Results

A. Datasets

We evaluated our method of nuclei segmentation using two datasets: BBBC006v1 from the Broad Bioimage Benchmark Collection [21] and hand-segmented U20S cells published in ISBI 2009 [22].

The BBBC dataset consists of 768 microscopy images and corresponding ground-truth images, produced by experts. The resolution of each image has 696×520, and pixels are 16-bit grayscale. In this dataset, the contrast between the background and nuclei varies from image to image: in our image the nuclei may have high intensity; in another their intensity might be low. Ten images were excluded from our experiments because their ground truth images did not contain any labeled nuclei.

The ISBI dataset consists of 48 microscopy images and their ground-truth images segmented by hand. The resolution of each image is 1349×1030 and pixels are 16-bit RGB. To allow these two datasets to be analyzed in a similar way, the RGB data was converted to grayscale, and the image size was reduced to 365×279. The sizes of nuclei in these reduced images were similar to the sizes of nuclei in the BBBC dataset. In this IBSI dataset, the contrast between background and nuclei varies between images, like the BBBC data, but this second dataset is more challenging because the nuclei tend to exhibit bumpy surface and irregular intensities. Moreover, some nuclei have shapes which are neither circular nor elliptical.

B. Algorithms Tested

We identify the stages in our method as follows: Local and Global: using local and global thresholding respectively. EM: splitting an unlabeled region into a number of components, corresponding to the number of seeds, using the expectation-maximization method. Merge: merging the components of the GMSM. Split: splitting a component of the GMSM into more than one component. Smoothing: using the error metric of (6). If there is no smoothing in a method name, (6) is still used, except for the second term, which relates to boundary roughness. The scales of all Gaussian kernels in our method were set to 1.6. The MATLAB source code of our method is available in the project webpage.[1]

Our method was compared with five other methods: graph cut-based segmentation (GC) [9], MINS [12], Otsu's segmentation [1], interactive learning and segmentation toolkit (ilastik) [13], and segmentation using non-overlapping extremal regions (ER) [15]. In the learning-based approaches such as ilastik and ER, classifiers were trained for each dataset. The parameters required by these methods were manually selected using a subset of the data, and their values are given in Table I. GC and Otsu's segmentation are fully automated. Using the ilastik method, the mean, variance, and a histogram of the neighborhood of each pixel were selected from the list provided with the method. The ER method is a supervised learning approach, and its performance was measured using 10-fold cross-validation. ER has a lot of parameters, which we did not test; we used the values recommended by its authors.

[1] http://imageinfo.inha.ac.kr/software/nucleus_seg/

TABLE I
Parameter Values Set for the Methods Being Compared

Methods	Parameters	Values
ilastik [13]	σ	3
	Threshold	0.8
	Neighborhood size	30
MINS [12]	nucleus diameter	35 pixels
	Image noise level	'Hard'
	Image smoothing kernel	1.2
Proposed	δ_H, $\delta_{\hat{\mathbf{x}}}$, $\delta_{\tilde{\mathbf{x}}}$, δ	0.25, 0.8, 0.25, 100
	γ	0.08
	$\sigma_{\min}$, $\sigma_{\max}$	7, 20

$\delta_{\hat{\mathbf{x}}}$ is used for the ellipse fitting, and $\delta_{\tilde{\mathbf{x}}}$ is used for the nucleus division.

Our method was implemented in MATLAB, and all experiments were performed on a PC with an Intel i7 CPU running at 3.40 GHz and with 16 GB of memory. In order to fairly measure computation times, we compared our method with ER and Otsu techniques that were implemented in MATLAB. Average computation times, which include training process, of our method, ER, and Otsu were 34.80, 37.53, and 0.03 sec. for each image in BBBC and ISBI datasets, respectively.

C. Evaluation Metrics

We evaluated the results obtained from all twenty methods using eight metrics that measure the effectiveness of clustering and segmentation. The normalized sum of distances (NSD), the Hausdorff metric (HM), and the Jaccard index (JI) followed the criteria described by Coelho *et al.* [22]. The NSD measures the distance between pixels in a segmented border and these in a reference border, and ranges from 0 to 1. The HM measures the shortest distance between a point on one boundary to another boundary, at the point where that distance is a maximum. The JI is the ratio of the number of identical members (pixels in this case) in two sets to the number of different members (pixels). The adjusted Rand index (AR) is the ratio of pair of identical member (pixels) in two sets to the total number of possible pairs; it ranges from 0 to 1, and is more practical to compute than the Rand index [23]. These evaluation metrics consider the background area as a kind of segmented object. To assess the nuclei alone, we eliminate the background region from the segmented objects. We also used the error-counting metrics introduced by Coelho *et al.* [22], which classify errors as split, merged, added, or missed nuclei.

Since the labeling of the segmented nuclei cannot be related to the labeling of the ground truth, and the evaluation subject is area, we are not evaluating a classification process, and therefore, we did not use accuracy metrics such as precision and recall, which specifically relate to classification.

D. Segmentation of the BBBC Dataset

In this experiment, the max normalization was applied to the dataset, and λ was set to 46.

1) Quantitative Performance Evaluation: Our method (Local, Global, Merge, Split, and Smooth) outperformed the

TABLE II
RESULTS OF SEGMENTING THE BBBC DATASET

Methods	NSD(L)	HM(L)	AR(H)	JI(H)	Average count	Split	Merged	Added	Missed
Local+EM	0.074	12.320	0.705	1.588	4.801	1.939	5.354	0.067	11.843
Local+Merge+Smooth	0.076	12.452	0.707	1.590	4.496	1.369	6.115	0.062	10.439
Local+Split+Smooth	**0.069**	12.056	0.697	1.581	4.578	2.976	3.393	0.098	11.843
Local+Merge+Split	0.108	11.999	0.646	1.519	5.284	7.691	2.834	0.170	10.439
Local+Merge+Split+Smooth	**0.068**	12.136	0.701	1.584	4.213	2.488	3.835	0.091	10.439
Global+EM	0.107	12.461	0.466	1.167	2.654	1.751	6.573	0.092	2.202
Global+Merge+Smooth	0.105	12.443	0.469	1.171	2.349	1.220	6.861	0.109	1.206
Global+Split+Smooth	0.074	**11.826**	0.464	1.170	2.022	4.038	1.656	0.194	2.202
Global+Merge+Split	0.119	**11.730**	0.418	1.129	3.181	10.299	0.967	0.251	1.206
Global+Merge+Split+Smooth	**0.072**	**11.809**	0.466	1.174	**1.698**	3.555	1.815	0.215	1.206
Local+Global+EM	0.115	12.902	0.758	1.673	2.956	1.943	7.682	0.715	1.483
Local+Global+Merge+Smooth	0.116	12.977	**0.761**	**1.678**	2.730	1.123	8.330	0.854	0.612
Local+Global+Split+Smooth	**0.069**	12.111	**0.762**	**1.685**	2.117	3.672	2.313	1.000	1.483
Local+Global+Merge+Split	0.103	11.948	0.706	1.615	3.019	9.071	1.078	1.317	0.612
Local+Global+Merge+Split+Smooth	**0.068**	12.151	**0.766**	**1.691**	**1.826**	2.927	2.617	1.145	0.612
OTSU [1]	0.290	14.920	0.413	1.143	12.252	2.094	5.145	0.181	4.918
GC [9]	0.158	12.834	0.491	1.259	3.162	4.235	1.898	5.625	0.889
MINS [12]	0.118	13.500	0.518	1.288	5.592	0.166	12.805	9.070	0.328
ilastik [13]	0.473	17.280	0.188	0.870	8.885	0.011	19.109	5.070	11.351
ER [15]	0.126	12.908	0.365	1.064	**1.715**	1.131	1.939	3.024	0.765

An appended L, in brackets, indicates that a lower value corresponds to a better segmentation result; an appended H indicates the opposite. The values in red, blue, and green are the best, second-best, and third-best, respectively.

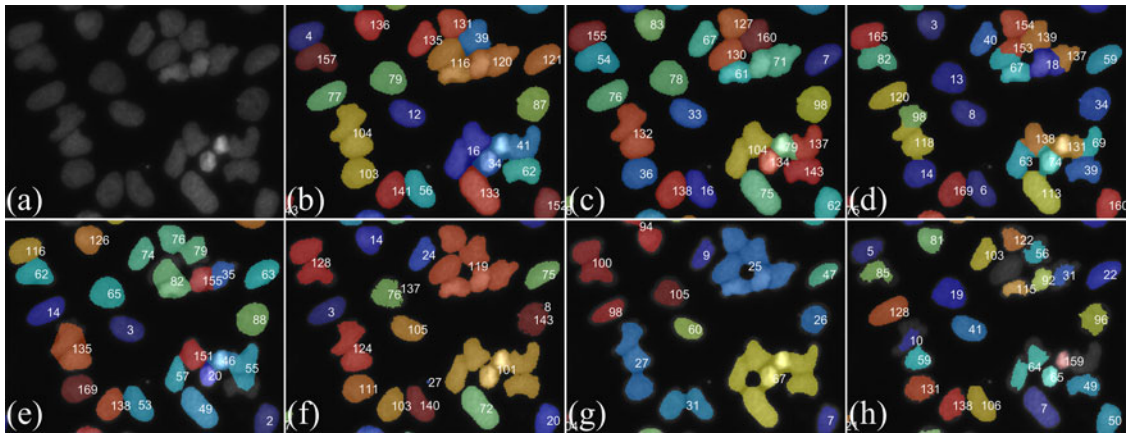

Fig. 7. Examples of segmentations of the BBBC dataset. (a) An original image and (b) the result of nuclei identified by the ground truth image; results produced by (c) The Local+Global+Merge+Split+Smooth version of our method, (d) GC, (e) MINS, (f) OTSU, (g) ilastik, and (h) ER. Each nucleus was randomly assigned a unique color and a number after segmentation.

four state-of-the-art methods and an Otsu-based segmentation method in terms of NSD, AR, JI, as shown in Table II . Furthermore, the average of the [22]'s four error counts was lower for our method them for most of the other methods. These results suggest that ROIs were extracted similar to ground-truth images without added background or missed foreground, and nuclei were separated properly. The large number of errors classified as split can be related to ambiguous boundaries between adjacent nuclei in the ground-truth images [see Fig. 7 (b)]. The MINS, ilastik and ER methods could not separate adjacent nuclei and therefore yielded few errors classified incorrectly split or merged nuclei.

When we compare different versions of our method, we find that combining the methods with Merge, Split, Smooth processes given the best results in terms of average error counts, regardless of the type of thresholding. In particular, the roughness score improves the method's ability to separate nuclei correctly. Since Hausdorff metric is only computed within a ROI, and increases as nuclei are added, versions with global thresholding produce higher HMs than versions with combined thresholding. However, the Local+Global+Merge+Split+Smooth process with the max normalization yielded the best scores overall in the condition with high contrast.

2) Qualitative Performance Evaluation: Fig. 7 compares results from our method (Local+Global+Merge+Split+Smooth) with results from five other methods, implemented by us. The ROI detected by GC was similar to the ground truth, but the boundaries identified yielded unexpected nuclei such as 18, 98, 131, and 153 in Fig. 7(d). MINS produced boundaries that were closer to the ground truth, but some ROIs contained errors,

TABLE III
RESULTS OF SEGMENTING THE ISBI DATASET

Methods	NSD(L)	HM(L)	AR(H)	JI(H)	Average count	Split	Merged	Added	Missed
Local+EM	0.092	12.214	0.398	1.207	2.198	0.250	1.083	0.458	7.000
Local+Merge+Smooth	0.098	12.383	0.401	1.209	1.943	0.083	1.479	0.458	5.750
Local+Split+Smooth	0.086	12.128	0.401	1.210	2.135	0.313	0.771	0.458	7.000
Local+Merge+Split	0.209	12.127	0.286	1.100	3.141	5.521	0.813	0.479	5.750
Local+Merge+Split+Smooth	0.089	12.254	0.405	1.215	1.839	0.146	0.979	0.479	5.750
Global+EM	0.103	12.341	0.679	1.789	1.260	0.354	1.875	0.917	1.896
Global+Merge+Smooth	0.105	12.245	0.678	1.784	**1.021**	0.104	2.000	1.000	0.979
Global+Split+Smooth	0.083	12.014	0.679	1.788	1.167	0.688	1.104	0.979	1.896
Global+Merge+Split	0.216	**11.719**	0.483	1.496	2.620	7.458	0.938	1.104	0.979
Global+Merge+Split+Smooth	**0.079**	12.014	0.680	1.788	**0.865**	0.375	1.021	1.083	0.979
Local+Global+EM	0.101	12.519	**0.697**	**1.854**	1.406	0.292	1.813	0.833	2.688
Local+Global+Merge+Smooth	0.105	12.381	0.687	1.836	1.161	0.083	1.958	0.896	1.708
Local+Global+Split+Smooth	**0.081**	12.213	**0.699**	**1.856**	1.307	0.521	1.104	0.917	2.688
Local+Global+Merge+Split	0.196	11.911	0.515	1.565	2.448	6.208	0.854	1.021	1.708
Local+Global+Merge+Split+Smooth.	0.085	12.022	0.687	1.836	**1.047**	0.375	1.146	0.958	1.708
OTSU [1]	0.389	**11.814**	0.561	1.594	2.875	0.083	2.000	8.500	0.917
GC [9]	0.260	**11.276**	0.576	1.616	3.276	9.458	0.854	2.125	0.667
MINS [12]	0.096	12.009	0.608	1.612	1.089	0.917	0.688	1.646	1.104
ilastik [13]	0.274	12.881	0.204	1.181	2.188	0.021	3.979	0.208	4.542
ER [15]	**0.080**	12.692	**0.711**	**2.146**	2.385	0.000	1.688	0.000	7.854

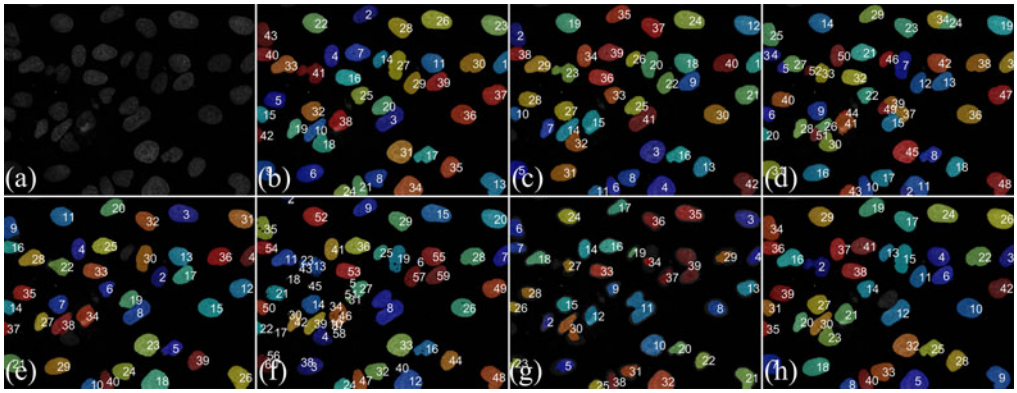

Fig. 8. Examples of segmentations of the ISBI dataset. (a) Original image and (b) nuclei identified from the ground-truth image; results produced by (c) our method (Local+Global+Merge+Split+Smooth), (d) GC, (e) MINS, (f) OTSU, (g) ilastik, and (h) ER. Each nucleus was randomly assigned a unique color and a number after segmentation.

such as 55 in Fig. 7(c). Although Otsu method was published some forty years ago, and it cannot separate adjoining nuclei, it was better at detecting ROIs than most of the other methods. The ilastik split boundaries with the best accuracy. ER was poor at ROI detection, but in general it split boundaries well.

The ROIs produced by our method appeared more similar to the ground-truth images than those produced by any of the other five methods. Our method also yielded boundaries closer to those intuitively inferred from the original image [Fig. 7(a)] than other methods. However, the boundaries between some adjacent nuclei in the ground-truth images (Fig. 7(b): 16, 41, 103, 104, 116, and 120] did not correspond to the boundaries in the original image. The other methods frequently failed to split such nuclei. The precise segmentation performed by our method increased the number of split errors, and reduced the number of split errors for the other methods.

E. Segmentation of the ISBI Dataset

Our method works well when there is high contrast between background and foreground regions, and nuclei are, which is the situation in the BBBC dataset. The ISBI dataset provides more challenging conditions, with reduced contrast between background and foreground regions, and nuclei with bumpier surface. We tested our method against this dataset, using average normalization to address the large differences in intensity between images, and low contrast between those nuclei and the background. The value of λ was set to 21.

1) Quantitative Performance Evaluation: The ER method was performed best overall, although our method with global thresholding did nearly as well as shown in Table III. It had the lowest average of split, merged, added and missed nuclei, a good ability to classify the boundaries between connected nuclei. Although our reconstructed nucleus and the metric evaluating error between the original nucleus and the recon-

structed nucleus were proposed for segmentation of a nucleus with smooth surfaces and a nucleus explicitly discriminated from background, our method showed good performances for identification of nuclei.

Among the variants of our method, the techniques using the Merge, Split, and Smooth operations performed better in terms of average error count, regardless of the type of thresholding. Following the average normalization of this dataset, global thresholding produced ROIs that were more similar to those from the ground-truth image than the combination of local and global thresholding. Combined thresholding of the cropped nuclei produced smaller ROIs than those found by global thresholding. This is the main reason for the increased number of missing nuclei. We therefore recommend the global thresholding with average normalization in the condition with low contrast between background and some nuclei.

2) Qualitative Performance Evaluation: Fig. 8 shows segmentations of data from the ISBI dataset produced by our method and other methods. GC showed the some adequate ability to detect ROIs as it did with the BBBC dataset, but the nucleus boundaries were again incorrect. MINS was unable to detect some ROIs, but the ROIs that it found, were adequate. The results produced by the Otsu's method had holes due to the bumpy surface of the nuclei. ilastik detected most of the nuclei, but the corresponding regions were very small. Some of the regions found by ER were incorrect, but in general, it showed superior ability to detect ROIs and find boundaries.

Overall, GC, ER, and our method showed excellent performance in terms of ROI detection. However, GC split nuclei, such as 38, 20, and 10 in Fig. 8(b), were split by GC, producing incorrect additional nuclei such as 41, 42; 39, 49; and 26, 51 in Fig. 8(d). ER failed to detect some nuclei, such as 20 in Fig. 8(b). Most of the nuclei were detected by our method, as shown Fig. 8(c), and the resulting image is very similar to the ground truth of Fig. 8(b).

VI. Conclusion

We have presented a new method of segmenting stained nuclei in microscopy images. Our seed detection reliably chose an initial seed point and its scale by fitting ellipses. Local and global thresholding were combined to overcome large differences in intensity between nuclei in an image and to find a ROI similar to the corresponding ground truth image. A reliable method of generating seed points identifies nuclei, and their scale is determined by ellipse fitting. The number of components found by Gaussian mixture models can be somewhat different from the actual number of nuclei, but nucleus boundaries are well identified by merge and split of these models. Our method performs best in the images with a contrasting foreground and nuclei of smooth surface. However, it also worked well on low-contrast images and nuclei with bumpy surface. Our method can estimate boundaries between adjoining nuclei, similar to those recognized by a human observer, using differences of intensity between the reconstructed nucleus and the original nucleus, and the roughness of the estimated boundaries. Future work will include finding ROIs in images with more complicated backgrounds.

References

[1] N. Otsu, "A threshold selection method from gray-level histograms," *Automatica*, vol. 11, no. 285–296, pp. 23–27, 1975.

[2] K. Jiang, Q.-M. Liao, and S.-Y. Dai, "A novel white blood cell segmentation scheme using scale-space filtering and watershed clustering," in *Proc. Int. Conf. Mach. Learn. Cybern.*, 2003, vol. 5, pp. 2820–2825.

[3] R. Beare and G. Lehmann, "The watershed transform in ITK-discussion and new developments," *Insight J.*, vol. 1, pp. 1–24, 2006.

[4] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *Int. J. Comput. Vis.*, vol. 1, no. 4, pp. 321–331, 1988.

[5] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Trans. Image Process.*, vol. 10, no. 2, pp. 266–277, Feb. 2001.

[6] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," *Int. J. Comput. Vis.*, vol. 22, no. 1, pp. 61–79, 1997.

[7] G. Li *et al.*, "3D cell nuclei segmentation based on gradient flow tracking," *BMC Cell Biol.*, vol. 8, no. 1, pp. 1–10, 2007.

[8] T. Liu, J. Nie, G. Li, L. Guo, and S. T. Wong, "ZFIQ: A software package for zebrafish biology," *Bioinformatics*, vol. 24, no. 3, pp. 438–439, 2008.

[9] Y. Al-Kofahi, W. Lassoued, W. Lee, and B. Roysam, "Improved automatic detection and segmentation of cell nuclei in histopathology images," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 4, pp. 841–852, Apr. 2010.

[10] H. Sharma *et al.*, "Determining similarity in histological images using graph-theoretic description and matching methods for content-based image retrieval in medical diagnostics," *Diagnostic Pathology*, vol. 7, no. 1, pp. 134–153, 2012.

[11] C. Li, C. Xu, C. Gui, and M. D. Fox, "Distance regularized level set evolution and its application to image segmentation," *IEEE Trans. Image Process.*, vol. 19, no. 12, pp. 3243–3254, Dec. 2010.

[12] X. Lou, M. Kang, P. Xenopoulos, S. Munoz-Descalzo, and A.-K. Hadjantonakis, "A rapid and efficient 2D/3D nuclear segmentation method for analysis of early mouse embryo and stem cell image data," *Stem Cell Rep.*, vol. 2, no. 3, pp. 382–397, 2014.

[13] C. Sommer, C. Straehle, U. Koethe, and F. A. Hamprecht, "Ilastik: Interactive learning and segmentation toolkit," in *Proc. IEEE Int. Symp. Biomed. Imag., Nano Macro*, 2011, pp. 230–233.

[14] H. Kong, M. Gurcan, and K. Belkacem-Boussaid, "Partitioning histopathological images: An integrated framework for supervised color-texture segmentation and cell splitting," *IEEE Trans. Med. Imag.*, vol. 30, no. 9, pp. 1661–1677, Sep. 2011.

[15] C. Arteta, V. Lempitsky, J. A. Noble, and A. Zisserman, "Learning to detect cells using non-overlapping extremal regions," in *Proc. Med. Image Comput. Comput.-Assisted Intervention*, 2012, pp. 348–356.

[16] A. Arbelle, N. Drayman, M. Bray, U. Alon, A. Carpenter, and T. R. Raviv, "Analysis of high-throughput microscopy videos: Catching up with cell dynamics," in *Proc. Med. Image Comput. Comput.-Assisted Intervention*, 2015, pp. 218–225.

[17] J. Stegmaier *et al.*, "Fast segmentation of stained nuclei in terabyte-scale, time resolved 3D microscopy image stacks," *PloS One*, vol. 9, no. 2, 2014, Art. no. e90036.

[18] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, USA: Springer, 2006.

[19] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[20] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Proc. 5th. Berkeley Symp. Math. Statist. Probab.*, Oakland, CA, USA, 1967, vol. 1 pp. 281–297.

[21] V. Ljosa, K. L. Sokolnicki, and A. E. Carpenter, "Annotated high-throughput microscopy image sets for validation," *Nature Methods*, vol. 9, no. 7, pp. 637–638, 2012.

[22] L. P. Coelho, A. Shariff, and R. F. Murphy, "Nuclear segmentation in microscope cell images: A hand-segmented dataset and comparison of algorithms," in *Proc. IEEE Int. Symp. Biomed. Imag., Nano Macro*, 2009, pp. 518–521.

[23] K. Y. Yeung and W. L. Ruzzo, "An empirical study on principal component analysis for clustering gene expression data," Dept. Comput. Sci. Eng., Univ. Washington, Seattle, WA, USA, *Tech. Rep.* UW-CSE-2000-11-03, Nov. 2000.