Biochemistry
From the Bench
Cite This: Biochemistry XXXX, XXX, XXX-XXX
pubs.acs.org/biochemistry

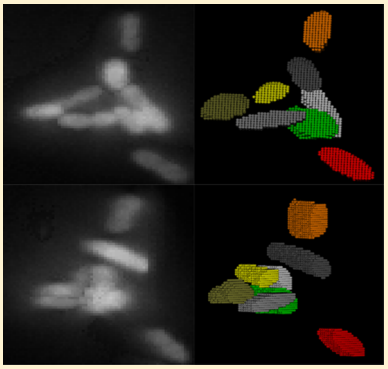
An Automated Image Analysis Method for Segmenting Fluorescent Bacteria in Three Dimensions

Matthew A. Reyer,[†] Eric L. McLean,[‡] Shriram Chennakesavalu,[§] and Jingyi Fei*[,†,∥]

[†]Institute for Biophysical Dynamics, [‡]Department of Molecular Genetics and Cell Biology, [§]College of the University of Chicago, and [∥]Department of Biochemistry and Molecular Biology, The University of Chicago, Chicago, Illinois 60637 United States

Supporting Information

ABSTRACT: Single-cell fluorescence imaging is a powerful technique for studying inherently heterogeneous biological processes. To correlate a genotype or phenotype to a specific cell, images containing a population of cells must first be properly segmented. However, a proper segmentation with minimal user input becomes challenging when cells are clustered or overlapping in three dimensions. We introduce a new analysis package, Seg-3D, for the segmentation of bacterial cells in three-dimensional (3D) images, based on local thresholding, shape analysis, concavity-based cluster splitting, and morphology-based 3D reconstruction. The reconstructed cell volumes allow us to directly quantify the fluorescent signals from biomolecules of interest within individual cells. We demonstrate the application of this analysis package in 3D segmentation of individual bacterial pathogens invading host cells. We believe Seg-3D can be an efficient and simple program that can be used to analyze a wide variety of single-cell images, especially for biological systems involving random 3D orientation and clustering behavior, such as bacterial infection or colonization.



Single-cell fluorescence microscopy has become a powerful method for studying the stochasticity of cellular activities and heterogeneity within a population.[1,2] To achieve single-cell resolution, an efficient and accurate segmentation method is a critical tool in the data analysis. Bacteria serve as model systems for investigating the fundamental mechanisms of many biological processes. In many experiments, bacteria are immobilized in two-dimensional (2D) surfaces or trapped and aligned in microfluidic devices;[2] therefore, many automatic segmentation methods are available to identify and track bacterial cells lying in two dimensions.[3−5] However, many interesting behaviors of microbes may not be observed in a 2D setting. Bacterial pathogens invading host cells[6] or bacteria undergoing a 2D to 3D transition during biofilm formation,[7] for instance, are often clustered with different 3D orientations. In such examples, changes in the phenotypes and cellular activities at the single-cell level may be not only time-dependent but also 3D space-dependent. Therefore, a 3D segmentation method designed for these specific conditions is necessary. Currently, a fully automatic 3D segmentation method for bacterial cells with high cell identification and segmentation accuracy is not available. We considered several well-cited programs used for 3D segmentation, including ImageJ,[8] Imaris (Bitplane, St. Paul, MN), Icy Spot Detector,[9] and Cell Profiler,[10] and found two general shortcomings. (1) Most of the commercial software can perform automatic 2D segmentation but not automatic 3D reconstruction to generate 3D segmentation, and (2) the performance on segmenting clustered cells is not satisfying (Figure S1).

Here we report a new analysis method, Seg-3D, for the segmentation of bacterial cells in three dimensions that can also be flexibly applied to 2D images. Seg-3D is based on local thresholding, 2D shape and concavity analysis, concavity-based cluster splitting in two dimensions, and morphology-based 3D reconstruction (Figure 1). First, we use a two-step local thresholding technique to minimize background noise and the additive background characteristic of bacterial clustering. Second, the likelihood of identified 2D objects being single cells is calculated using 2D-shape and concavity analysis. Third, we split multicell clusters at concave points along the object borders, which denote cell boundary intersections.[11−13] The single-cell identification step and concavity-based splitting iterate until all objects can be either identified as single cells or discarded. Finally, 2D objects are combined with their most likely partners in adjacent slices to form 3D cells. The parameters used to reconstruct 3D cells can be determined using low-density, single-cell images, which improves the flexibility and accuracy of automatic 3D segmentation.

IMAGE SMOOTHING AND THRESHOLDING

3D image stacks are loaded as 3D matrices and converted to grayscale. The grayscale image stacks are minimally smoothed using an anisotropic diffusion algorithm,[14] with previously published MATLAB code.[15] We compared the effect of the number of smoothing iterations using experimental images containing densely packed cells (Figure S2). Without

Special Issue: Future of Biochemistry

Received: August 27, 2017
Revised: October 25, 2017
Published: October 30, 2017

DOI: 10.1021/acs.biochem.7b00839
Biochemistry XXXX, XXX, XXX−XXX

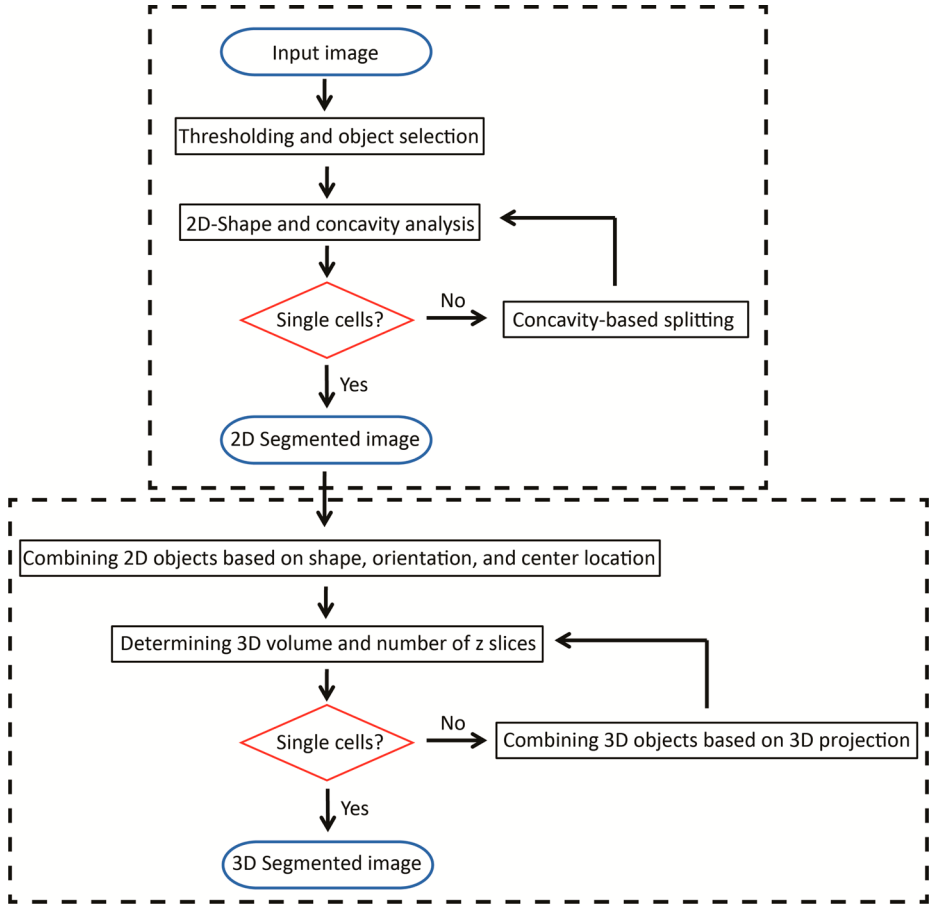


Figure 1. Workflow of the image analysis program. Details are explained in the text.

smoothing, the object outlines remain jagged to the point of creating many identifiable concave points. After three smoothing iterations, the object outlines become smooth and, mostly, significant concave points, representing actual intersections between cells, remain. After 10 smoothing iterations, objects become indistinguishable and meaningful concave points are obscured. Depending on the actual properties of the input images, especially the signal-to-noise ratio, the users may need to optimize the number of smoothing iterations empirically. However, we generally recommend one to three iterations, because, on one hand, without any smoothing, signal variations within the same cell can cause more roughness on the edges that leads to oversplitting of cells into multiple small fragments (Figure S3); on the other hand, oversmoothing can lead to the merging of multiple cells into one.

After image smoothing, the first stage in cell identification is implemented through a two-step local thresholding technique. First, the original image is sharpened by subtracting its corresponding low-frequency image, in which only the low-frequency signals of the Fourier-transformed image are kept, and then rescaling.[16] This helps correct for the slow-sweeping changes in background illumination characteristic of bacteria clustering around distinct puncta in images. Second, objects pass through a Bradley adaptive threshold,[17,18] wherein the intensity of pixels of candidate objects must be a defined percentage higher than the mean intensity of the pixels in an $n \times n$ neighborhood (we set n equal to 10, 11, or 12, depending on the size of the image). We chose a local, adaptive threshold with a fixed percentage, rather than a single, global threshold, to avoid false positives and false negatives due to uneven illumination, autofluorescence from the host cells, and/or additive fluorescence from neighboring cells. We tested the threshold parameter sensitivity of the 2D object selection on experimental input images with or without smoothing. The initial thresholding results are robust across a range from 25 to 400% of the default intensity thresholding value (0.00001), leading to no discernible differences in the initial characterization of the objects (Figure S4).

SINGLE-CELL IDENTIFICATION BY SHAPE AND CONCAVITY ANALYSIS

For each identified object in a 2D slice, we perform shape and concavity analysis to distinguish single cells from clustered cells. For shape analysis, we fit each 2D object to an ellipse using a least-squares criterion.[19,20] We then compare the actual outline of the thresholded object to the ellipse and calculate the deviation of the outline from the best-fit ellipse, using an analogue of the Hausdorff distance[21] (Figure 2a). The assumption is that bacterial cells are roughly elliptical, and if a 2D slice of an object deviates too far from its best-fit ellipse, it is likely not a single cell (Figure 2c). The deviation of the object from its best-fit ellipse is calculated by the sum of the distance between each object edge pixel and the nearest edge pixel in the best-fit ellipse. The sum is then normalized by the total number of pixels on the periphery of the object to give an error value ($Err_{2D\text{-}shape}$) (Figure 2a).

To calculate the concavity of each edge pixel (Figure 2b), the edge coordinates are first arranged into a two-column array and

DOI: 10.1021/acs.biochem.7b00839
Biochemistry XXXX, XXX, XXX−XXX

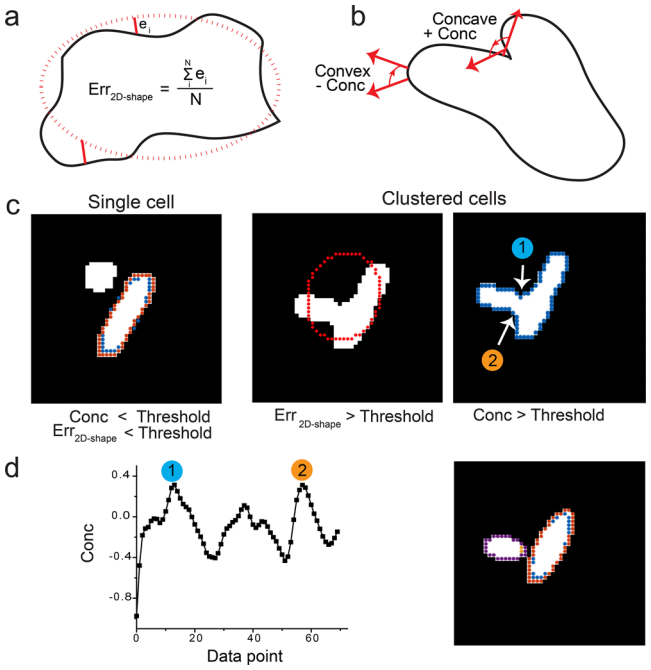


Figure 2. Shape and concavity analysis for single-cell identification in two dimensions. (a) Illustration of the calculation of $Err_{2D\text{-}shape}$. (b) Illustration of the calculation of Conc. (c) Example of single cell vs clustered cells, showing a difference in $Err_{2D\text{-}shape}$ and Conc. The outline of the isolated, single cell closely resembles its best-fit ellipse, resulting in a subthreshold $Err_{2D\text{-}shape}$ value. The outline of the clustered cells, in contrast, deviates significantly from their best-fit ellipse, resulting in a high $Err_{2D\text{-}shape}$ value and leading to this object being labeled as a nonsingle cell. Similarly, the isolated single cell is a mostly convex object, whereas the clustered cells can be easily identified by the presence of one or more concave points. (d) Concavity values for every border pixel in the clustered cell object are shown in the plot (left). The two most concave points are marked and are made candidates for splitting location. These splitting coordinates result in a successful segmentation.

then smoothed with a Savitzky−Golay filter.[22] After the smoothing, the tangent line of a given edge pixel is approximated by fitting a straight line to the edge pixel and two pixels before and after the edge pixel. The angle of the approximate normal line, perpendicular to the tangent line, is then calculated. Finally, the difference between the angles of the normal lines for $pixel_i$ and $pixel_{i-1}$ is recorded, which we define as a parameter for concavity (Conc) (Figure 2b). With our definition, a positive Conc value marks a concave point (Figure 2b). In the case in which multiple cells are clustered into the same binary object, the point at which the cells intersect is a concave point in the binary map (Figure 2c).

Therefore, we could distinguish single cells from clustered cells using their $Err_{2D\text{-}shape}$ and Conc values. If $Err_{2D\text{-}shape}$ and Conc are below the user-defined thresholds, the object is classified as a single cell; otherwise, the object is deemed a nonsingle cell and is subjected to concavity-based splitting (Figure 2c).

CONCAVITY-BASED SPLITTING

Objects that do not pass the single-cell identification described above are passed through a concavity-based splitting algorithm, adapted from previously published methods.[11−13] After calculating and storing the concavity of every edge pixel for nonsingle cells, we arrange them from highest to lowest Conc value (x_1 to x_n, respectively), disregarding convex points (Conc < 0), and select the largest Conc values along the object borders as candidate locations for splitting (Figure 2d). Starting from the pair of x_1 and x_2, we draw a straight line between the points, splitting the cell. After the cell is split, we characterize the fragments using the single-cell identification procedure described above. If one or both of the fragments now pass the single-cell identification, we move on to the next nonsingle cell object. If neither of the fragments passes the threshold, we draw a new line between points x_1 and x_3, x_1 and x_4, ..., and x_1 and x_n, and then between points x_2 and x_3, etc. In the interest of time, if the nonsingle cell is not split into objects that pass the single-cell identification in k attempts (k can be adjusted by the user), the algorithm passes over this object onto the next one, and the object remains classified as a nonsingle cell and discarded in the end.

MORPHOLOGY-BASED 3D RECONSTRUCTION

After each slice of the 3D image has been fully segmented and characterized, the 2D slices of the cells are combined into 3D volumes. We introduce three parameters for 3D reconstruction: D_{Center}, the distance between the geometric centers of the candidate objects from two consecutive z slices (Figure 3a);

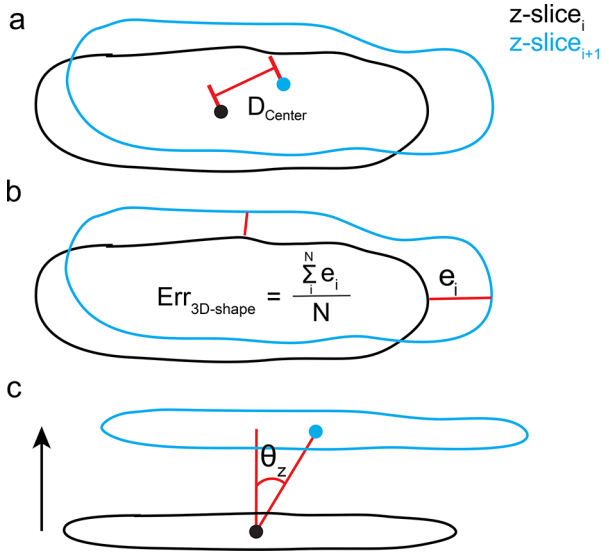


Figure 3. Parameters for 3D reconstruction. (a) Illustration of the calculation of D_{Center} between potential partner objects in adjacent z slices. (b) Illustration of the calculation of $Err_{3D\text{-}shape}$ for potential partners in adjacent z slices. (c) Illustration of the calculation of θ_z for a 3D cell.

$Err_{3D\text{-}shape}$, the shape deviation between two candidate objects, a parameter equivalent to $Err_{2D\text{-}shape}$, substituting the best-fit ellipse with a potential partner 2D object (Figure 3b); and θ_z, the angle between the z axis and the line connecting the centers of the candidate objects from two consecutive z slices (Figure 3c). The use of these parameters is based on the assumption that each projection on the $x-y$ plane from a single cell should have similar localization and shape, and the orientation of the cell relative to the z axis should be a constant. Therefore, we can define thresholds for these parameters to determine whether 2D objects belong to the same cell. If multiple objects in $slice_{i+1}$ meet the criteria for 3D recombination with a 2D cell in $slice_i$, the object in $slice_{i+1}$ that minimizes these parameters is chosen as the partner.

DOI: 10.1021/acs.biochem.7b00839
Biochemistry XXXX, XXX, XXX−XXX

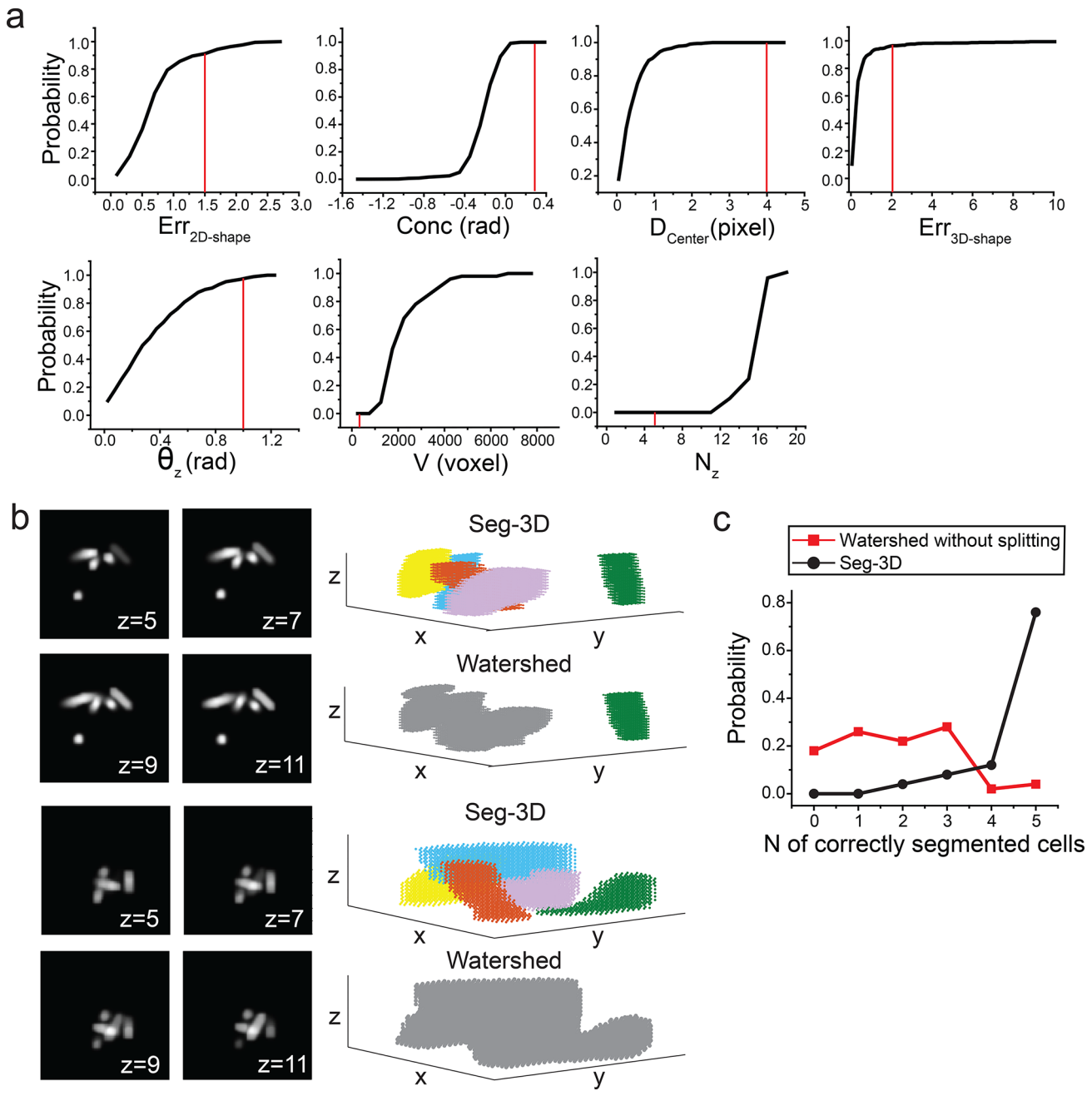


Figure 4. Examination of Seg-3D on synthetic images. (a) Distribution of all parameters from synthetic images containing single cells. Red lines mark the thresholds used in analyzing synthetic images. (b) Two examples of 3D segmentation on synthetic images with Seg-3D and the watershed-based method without object splitting or manual fixation. Each group of successfully segmented cells is color-coded, while cells that failed to segment correctly are colored gray. (c) Success probability of Seg-3D (black) and the watershed-based method (red). Each synthetic image contains five randomly oriented but closely neighboring cells.

The reconstructed 3D objects are checked for two additional criteria: V, the total voxel number occupied by the cell, and N_z, the number of z slices occupied by the cell. Considering all possible orientations, a single cell should still occupy a minimum number of z slices and 3D volume. Therefore, by applying these two criteria, we can eliminate incompletely reconstructed cell fragments due to mis-segmentation in two dimensions. The identified 3D fragments based on V and N_z can then be combined into a complete single-cell candidate if they have matching θ_z and small D_{Center} values. Fragments that remain after attempted 3D recombination will be disregarded if they do not exceed the V and N_z thresholds.

METHOD VALIDATION WITH SYNTHETIC DATA

To validate Seg-3D, we first tested it on synthetic data (Figure 4). To generate the synthetic data, we modeled bacterial cells as 3D rod-shaped objects comprised of individual voxels with an xy pixel size and z step interval of 130 nm × 130 nm × 130 nm, the parameters of our microscope and camera setup.[23] The 3D objects were then convolved with the point-spread function to better represent the 3D image of the bacteria. Several synthetic bacteria were randomly placed in 3D space with certain interbacterial space to make synthetic images with certain crowdedness such that cells can touch but cannot intersect (Figure 4b).

DOI: 10.1021/acs.biochem.7b00839
Biochemistry XXXX, XXX, XXX−XXX

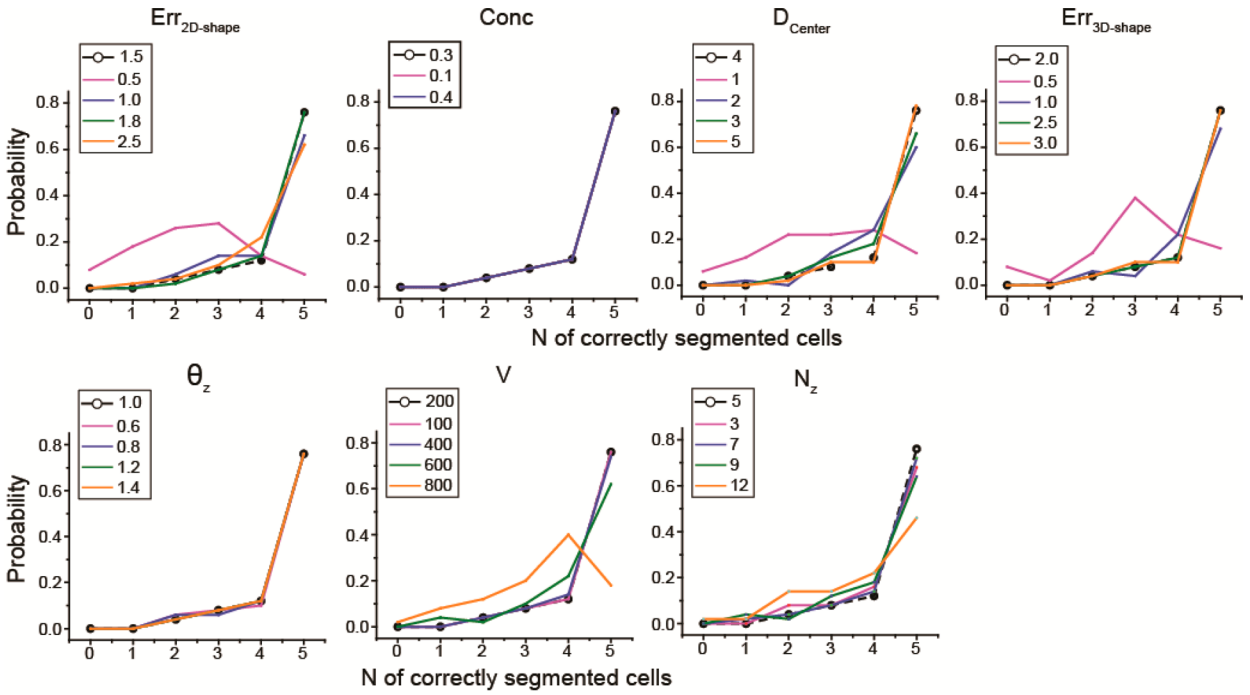


Figure 5. Test of parameter sensitivity on synthetic images. Parameters are varied relative to the default values marked in Figure 4. The success probability using default parameters is shown as dashed lines.

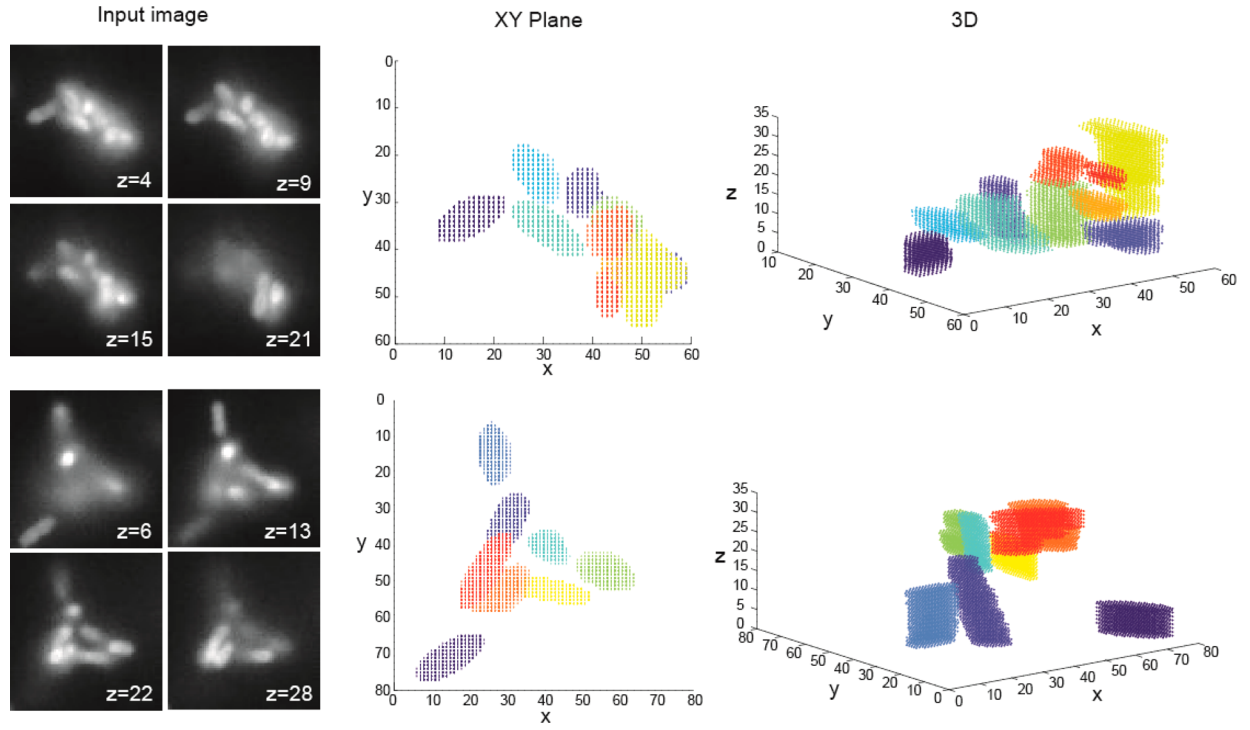


Figure 6. Examination of Seg-3D on experimental images. Segmented cells are color-coded.

Seg-3D requires several user-input parameters for single-cell identification and 3D reconstruction as described above, including $Err_{2D\text{-}shape}$, Conc, D_{Center}, $Err_{3D\text{-}shape}$, θ_z, V, and N_z. To correctly decide the thresholds for these parameters, we generated synthetic single cells with a random orientation and extracted all the parameters from these single cells (Figure 4a). A histogram of each parameter represents the expected range from single cells. Thresholds were then set on the basis of the histograms to include at least 90% of the single-cell population. Specifically, we picked values marked by the red lines in Figure 4a as parameters for analyzing all of our images.

Seg-3D was then applied to the synthetic data (Figure 4b). We compared Seg-3D with previously published methods used to study bacterial biofilm based on the watershed algorithm without splitting or user correction.[7,24] Seg-3D showed improved accuracy in segmentation of the clustering bacteria.

DOI: 10.1021/acs.biochem.7b00839
Biochemistry XXXX, XXX, XXX−XXX

With 50 randomly generated synthetic images each containing five clustered bacteria, Seg-3D correctly segmented all five cells in ~76% of the synthetic images. Of 250 cells in the synthetic data, Seg-3D correctly segmented ~92% of them, while the watershed algorithm correctly segmented only 36% of the cells (Figure 4c). Several cases contributed to the incorrect segmentation (~8% of the total), including failure in splitting two significantly merging cells in two dimensions (missing splitting), failure to identify objects in two dimensions (missing objects), and failure to combine two 3D fragments belonging to one cell (oversplitting). Very rarely (two of 250 cells from 50 synthetic images), we observed misidentification of an object from the background, which was characterized as a cell. We found tuning parameters such as $Err_{2D\text{-}shape}$ could eliminate this error.

TEST OF PARAMETER SENSITIVITY

To evaluate the parameter sensitivity of the segmentation accuracy, we picked four other values smaller than (minimum of 25% of) and larger than (maximum of 400% of) the standard values for each parameter and tested them on the same synthetic data sets. Our results show that the Seg-3D values are robust to the changes in these user-input parameters as long as they exclude the expected ranges for single cells. Specifically, the results are not sensitive to the change in Conc (0.1−0.4), as single cells should generate only negative Conc values, or sensitive to the change in θ_z (0.6−1.4). Mis-segmentation happens frequently when the choice of the parameters gets close the single-cell parameter range. For example, when $Err_{2D\text{-}shape}$ is set to 0.5 (too stringent), 2D objects more often fail to be considered as single cells after a few trials of concavity-based splitting and are therefore discarded. When $Err_{3D\text{-}shape}$ or D_{Center} is set to be too small (too stringent), 2D objects more often fail to be considered to come from the same 3D cell, causing frequent oversplitting in three dimensions. The choice of a large value for V or N_z causes rejection of small cells and therefore undercounts the cell number (Figure 5).

APPLICATION OF THE METHOD TO INTRAHOST PATHOGENIC BACTERIA

We then applied our 3D segmentation method to sample images of bacteria invading macrophages (Figure 6). *Salmonella* cells expressing GFP from a constitutive promoter[25] were used to infect murine macrophages (RAW 264.7), and macrophages were then fixed and imaged under the fluorescence microscope. Similar to the synthetic data analysis, we first analyzed the distributions of all critical parameters corresponding to single cells from low-cell density images. Relative to the histograms derived from synthetic data, we found that ranges expected from single cells were very similar for all seven parameters, suggesting the parameters would be robust for applications to bacterial species with similar 3D shape and size. Therefore, we directly applied the same thresholds to analyze the real data. Figure 6 shows two examples of clustered intramacrophage *Salmonella* cells, and Seg-3D effectively segmented individual *Salmonella* in three dimensions.

ADDITIONAL FEATURES IN THE USER INTERFACE

While Seg-3D has greatly improved the efficiency and accuracy of 3D segmentation, mis-segmentation still occurs. For example, when two cells happen to be completely touching each other with the exact same orientation, it is very likely that they can pass the single-cell criteria in 2D segmentation, or the concavity-based automatic splitting fails to split them. Therefore, we add in the manual proofread and correction feature after automatic segmentation. Possible corrections include switching positive selections to negative, switching negative selections to positive, manually drawing split lines for remaining clustered objects, manually drawing borders for missed objects, and deleting objects.

Moreover, the algorithm allows analysis for multichannel images. 3D segmentation will be performed on a user-defined channel with a uniformly stained fluorescent signal that can represent the full cell volume well. The fluorescence signal from other channels with staining on biomolecules of interest will be allocated into each segmented cell, allowing further single-cell quantification of biomolecules. Finally, the 3D segmentation code can be very flexibly adapted to analyze surface-attached cells in two dimensions. Seg-3D will benefit single-cell imaging and analysis under complex conditions such as bacterial pathogen infection and biofilm formation. The full package of Seg-3D, coded in MATLAB, with a user manual can be found in the Supporting Information or downloaded as open source code (https://github.com/JingyiFeiLab/Cell_Seg).

ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acs.biochem.7b00839.

Supplementary Figures S1−S4 (PDF)

Seg-3D analysis package (ZIP)

AUTHOR INFORMATION

Corresponding Author

*GCIS W142, 929 E. 57th St., Chicago, IL 60615. E-mail: jingyifei@uchicago.edu.

ORCID

Jingyi Fei: 0000-0002-9775-3820

Funding

J.F. acknowledges support from the Searle Scholars Program and the National Institutes of Health (R01 GM092830-06A1). E.L.M. is supported by a Molecular and Cellular Biology Training Grant (2T32GM007183-42). S.C. is supported by the Katen Scholars Program at The University of Chicago.

Notes

The authors declare no competing financial interest.

ACKNOWLEDGMENTS

The authors acknowledge Dr. Yong Wang, Dr. Seongjin Park, and Soo Ji Kim for testing Seg-3D.

REFERENCES

(1) Vera, M., Biswas, J., Senecal, A., Singer, R. H., and Park, H. Y. (2016) *Annu. Rev. Genet. 50*, 267−291.

(2) Skylaki, S., Hilsenbeck, O., and Schroeder, T. (2016) *Nat. Biotechnol. 34*, 1137−1144.

(3) Elowitz, M. B., and Leibler, S. (2000) *Nature 403*, 335−338.

(4) So, L.-H., Ghosh, A., Zong, C., Sepúlveda, L. A., Segev, R., and Golding, I. (2011) *Nat. Genet. 43*, 554−560.

(5) Sadanandan, S. K., Baltekin, O., Magnusson, K. E. G., Boucharin, A., Ranefall, P., Jalden, J., Elf, J., and Wahlby, C. (2016) *IEEE J. Sel Top Signal Process 10*, 174−184.

DOI: 10.1021/acs.biochem.7b00839
Biochemistry XXXX, XXX, XXX−XXX

(6) Weiner, A., Mellouk, N., Lopez-Montero, N., Chang, Y.-Y., Souque, C., Schmitt, C., and Enninga, J. (2016) *PLoS Pathog. 12*, e1005602.
(7) Yan, J., Sharo, A. G., Stone, H. A., Wingreen, N. S., and Bassler, B. L. (2016) *Proc. Natl. Acad. Sci. U. S. A. 113*, E5337−43.
(8) Abràmoff, D. M. D. (2004) *Biophotonics International 11*, 36−42.
(9) de Chaumont, F., Dallongeville, S., Chenouard, N., Hervé, N., Pop, S., Provoost, T., Meas-Yedid, V., Pankajakshan, P., Lecomte, T., Le Montagner, Y., Lagache, T., Dufour, A., and Olivo-Marin, J.-C. (2012) *Nat. Methods 9*, 690−696.
(10) Carpenter, A. E., Jones, T. R., Lamprecht, M. R., Clarke, C., Kang, I. H., Friman, O., Guertin, D. A., Chang, J. H., Lindquist, R. A., Moffat, J., Golland, P., and Sabatini, D. M. (2006) *Genome Biol. 7*, R100.
(11) Song, H., Zhao, Q., and Liu, Y. (2014) *Frontiers of Computer Science 8*, 156−162.
(12) Bai, X., Sun, C., and Zhou, F. (2008) *2008 Digital Image Computing: Techniques and Applications*, pp 271−278, IEEE, New York.
(13) Indhumathi, C., Cai, Y. Y., Guan, Y. Q., and Opas, M. (2011) *J. Microsc. 243*, 60−76.
(14) Perona, P., and Malik, J. (1990) *IEEE Trans Pattern Anal Mach Intell 12*, 629−639.
(15) Lopes, D. S. (2007) *Anisotropic Diffusion (Perona & Malik)*, MATLAB, MATLAB Central File Exchange.
(16) Arce, S. H., Wu, P.-H., and Tseng, Y. (2013) *Sci. Rep. 3*, 2266.
(17) Bradley, D., and Roth, G. (2007) *Journal of Graphics Tools 12*, 13−21.
(18) Motl, J. (2015) *Bradley Local Image Thresholding*, MATLAB, MATLAB Central File Exchange.
(19) Fitzgibbon, A. W., Pilu, M., and Fisher, R. B. (1996) *Proceedings of the 13th International Conference on Pattern Recognition*, Vol. *1*, pp 253−257, IEEE, New York.
(20) Gal, O. (2003) *fit_ellipse*, MATLAB, MATLAB Central File Exchange.
(21) Huttenlocher, D. P., Klanderman, G. A., and Rucklidge, W. J. (1993) *IEEE Trans Pattern Anal Mach Intell 15*, 850−863.
(22) Savitzky, A., and Golay, M. J. E. (1964) *Anal. Chem. 36*, 1627−1639.
(23) Portal, R., Dias, J., and de Sousa, L. (2010) *Archive of Mechanical Engineering LVII*.
(24) Meijering, E. (2012) *IEEE Signal Process Mag 29*, 140−145.
(25) Westermann, A. J., Förstner, K. U., Amman, F., Barquist, L., Chao, Y., Schulte, L. N., Müller, L., Reinhardt, R., Stadler, P. F., and Vogel, J. (2016) *Nature 529*, 496−501.

DOI: 10.1021/acs.biochem.7b00839
Biochemistry XXXX, XXX, XXX−XXX