Fast Automated Image Segmentation of Overlapping Cervical Cells

Timothy Lee, Hari Ravichandran, Jason Wang

April 9, 2017

1 Introduction

Cervical cancer is the the fourth most common cause of cancer and fourth most common cause of cancer-specific death in women [1]. An estimated 528,000 new cases and 266,000 cervical cancer mortalities occur each year. Cervical cancer typically develops from precancerous changes in cells lining the uterine cervix over the course of 10 to 20 years [1], providing a large window for early detection. The Pap smear, which tests for the presence of precancerous or cancerous cells in the cervix, is a common early screening test for cervical cancer. Although the Pap smear does not directly test for human papillomavirus (HPV), which is involved in more than 90% of cervical cancer cases [2], it can identify cellular changes caused by the virus. Cervical cells extracted by the Pap smear are assessed by a human expert under the microscope.

In developed countries, the wide adoption of cervical screening programs has significantly reduced the prevalence of cervical cancer [3]. However, in settings with limited resources, there exists a shortage of human experts to accurately diagnosis Pap smear results; in fact, in low-income countries, cervical cancer remains the most common cause of cancer-related death in women [4]. Manual analysis of microscopic images also suffers from inter-observer variability and is difficult to scale to support large patient populations. The development of a machine-assisted diagnostic pipeline for cervical cancer from Pap smear microscopic images is thus of interest.

Key metrics for the identification of pre-cancerous changes in the uterine cervix include number of cells, cell morphology, and cytoplasm area. Clear delineation of distinct cervical cancer cells (nuclei and cytoplasm) is required. However, the task of segmenting cell masses into distinct cells proves difficult due to the heterogeneity of the cytoplasm, imaging noise, and multi-layer overlap between neighboring cells. Figure 1 shows an example image of overlapping cervical cells from a Pap smear test that encapsulates the challenges associated with segmenting cell masses.

In the context of Pap smears, a number of approaches for overlapping cell segmentation have been discussed. The large majority of algorithms split the pipeline of overlapping cell segmentation into 3 components: 1) cell mass detection, 2) nucleus detection, and 3) cytoplasm segmentation [5, 6, 7, 8]. A number of approaches sought to group raw pixels into larger, more representative clusters based on variations in intensity. Ushizima et al. introduces the concept of nuclei identification via superpixel partitioning, which seeks to reduce the full pixel space into a smaller subset of pixel clusters [5]. Ushizima et al. then proceeds to find cell segmentations by calculating the Voronoi diagram, using the previously detected nuclei as constraints to the cytoplasm boundary [5]. Recently, Lu et al. and Nosrati et al. have proposed a series of methods that impose prior assumptions on cell and nuclei shape [6, 7]. Lu et al. assessed the performance of a level sets-based method to segment nuclei and cytoplasm using an energy function that enforces an elliptical shape prior, and includes pairwise terms measuring the area overlap ratio and intensity ratio between neighboring cells. Nosrati et al. improved upon this approach using a star-shaped prior, that allowed for more degrees of variance [7].

Unfortunately, these state-of-the-art cell segmentation algorithms are computationally intensive and thus require significant computing power and runtime. There is a need for faster algorithm that can operate with fewer resources and less time, as in the case of limited resource clinical settings. In this report, we propose one such algorithm that combines components of previous work, but introduces a number of heuristics to enhance runtime.

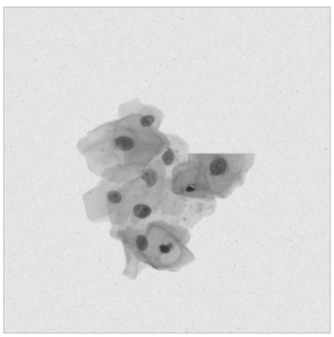
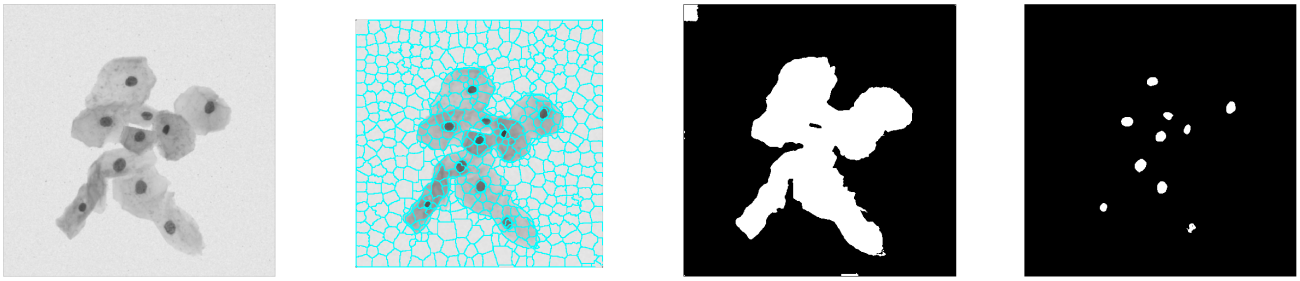
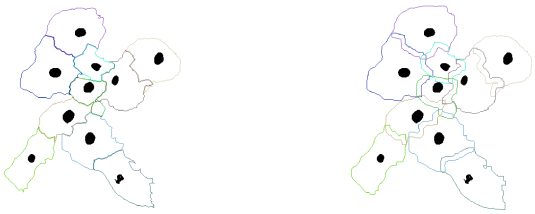


Figure 1: Example image of overlapping cervical cells from Pap smear test



(a) Raw Image (b) Segmented superpixels (c) Extracted cell mass (d) Extracted nuclei



(e) Preliminary boundaries (f) Refined final boundaries

Figure 2: Complete Pipeline of Cell Segmentation

2 Data

In 2014 and 2015, the International Symposium for Biomedical Imaging (ISBI) hosted a series of clinically-relevant grand challenges in computer vision, one of which was the creation of a robust algorithm to automatically segment overlapping cervical cells from Pap smear microscopic images [9, 10].

The challenge provided a dataset of 512 x 512 pixel grayscale images obtained from real Pap smear tests as well as computer simulation. Each image contained anywhere from 2 to 10 cells, with different degrees of contrast, overlap, and texture. In the 2014 data set, there are 16 real cervical cytology images and 945 synthetic images generated by the computer simulation. The organizers designated 151 of these images for training and hyperparameter tuning. The remaining 810 were allocated for algorithm evaluation as the test set [9].

The 2015 data set introduced only 16 additional images, which we did include in our report [10]. Each image included ground truth annotations for cytoplasm boundaries and masses. In assessing our approach, the ground truth annotations were compared against our predicted annotations.

3 Approach

3.1 Proposed Algorithm

We proceed with a model that more properly segments cell masses into individual cells by taking into account information about the location of the nuclei within each cell mass. We describe three steps of a pipeline: cell mass detection, nucleus detection, and cytoplasm segmentation. Figure 2 shows the results of major milestones along the pipeline to extract cell boundaries.

3.1.1 Cell Mass Detection

The goal of this first step is to extract shapes of the clumps of cells, which may or may not be overlapping, from the raw image. In the end, we can ignore the background extracellular matrix (ECM) around the cells. First, to remove high frequency signals from the raw image, so that the background values are more consistent, we use a median filter with a [5x5] window filter.

Next, we segment the filtered image into superpixels, clustered pixels with similar intensity values, using the Simple Linear Iterative Clustering (SLIC) algorithm. This unsupervised classification algorithm is an analog of k-means clustering. After initializing cluster centroids at the lowest gradient position, it iteratively rematches pixels to the new closest centroids based on the distance measure described in Equation (1):

$$D_s = d_g + \frac{m}{S} d_{xy} \tag{1}$$

where d_g represents the absolute difference in grayscale values between the centroid and the pixel, d_{xy} represents the Euclidean distance, m is a hyperparameter that controls the compactness of the superpixel, and S is a constant normalized by the number of initial clusters. Cluster centroids are then updated and pixels are rematched until convergence.

Within each superpixel, all pixel values are converted into the median value of original pixel values within the superpixel. This process further reduces high-frequency noise and serves to make our overall cell mass detection algorithm less computationally expensive.

Next, we reduce the image into a binary image by calculating a threshold that differentiates between areas inside a cell and outside a cell using Otsu's method. Otsu's method picks a threshold that minimizes intra-class variance; the complete optimization objective shown in Equation (2):

$$\min_t \omega_0(t)\sigma_0^2(t) + \omega_1(t)\sigma_1^2(t) \tag{2}$$

where $\omega_0(t)$ and $\omega_1(t)$ represent the probabilities of the two classes separated by threshold t, and $\sigma_0^2(t)$ and $\sigma_1^2(t)$ represent the standard deviations of the two classes.

In the initial binary image, certain pixel clusters that were classified to be inside a cell were actually too miniscule to comprise real cells. Thus, we calculate the area of each connected component inside the image. The areas that were too small (under 25 pixels) were relabeled as outside the cell. Any remaining connected components are the resulting cell masses.

3.1.2 Nucleus Detection

In this second step, we describe a process of obtaining viable nuclei from each extracted cell mass. Intuitively, nuclei within the image are the small darker spots within the cell membrane. First, we keep only portions of the image corresponding to the cell masses, extracted from the previous step. Next, we reduce the full pixel space into an image with k classes of pixels, by calculating multiple darkness thresholds using Otsu's method. Due to large variations of pixel darkness within cell masses, using only binary classification tends to induce false positives in nucleus classification. Thus we use $k > 2$. In the case of multi-class classification for Otsu's method, there is a similar minimization objective of intra-class variance, shown in Equation (3):

$$\min_{t_1,\ldots,t_k} \omega_0(t)\sigma_0^2(t) + \omega_1(t)\sigma_1^2(t) + \ldots + \omega_{k+1}\sigma_{n+1}^2(t) \tag{3}$$

In this objective, the algorithm picks k thresholds to segment the pixels of the picture into k+1 classes with ω representing the class probability and σ representing the variance for a particular class. We consider only the darkest class of pixels as nuclei.

Sometimes the nuclei extracted are too small or not round enough to be considered nuclei, thus requiring that we reject these predictions. Considering each connected nucleus component, if its number of pixel is less than a threshold number (100) then the nucleus annotation is not considered. Additionally, we use principal components analysis (PCA) as a heuristic for assessing the roundness of a nucleus. For each connected nucleus component, we consider the (x,y) coordinates of its pixels in a scatterplot. Next, we find the first and second principal components, which correspond to the vector directions that maximize and minimize intra-nucleus pixel location variance respectively. We can then obtain the variance of the scores in the first and second principal components. The reasoning is that an ideal nucleus is shaped as a near-perfect circle; thus, the variance explained by the first and second principal components should be almost equal. But when the nucleus is shaped more like a rod, much more of the variance is explained only by the first principal component. If the ratio of variance explained by the first component to variance by the second component is too high, the nucleus is rejected. Only the remaining nuclei that fit the criterion are considered.

3.1.3 Cytoplasm Segmentation

Using the locations of the nuclei obtained by the previous step, we can then outline the borders of the cell around each nucleus. First we consider the superpixels of the image again, but only those that are located inside a cell mass component. Considering each cell mass component, and the superpixels and nuclei located inside that cell mass component, we find for each superpixel, the nucleus that it is closest to. The superpixels that have the same closest nucleus are considered as one cell in the cell component. This produces a preliminary boundary where cells of a component cannot overlap each other. Since it is defined by the superpixel generation algorithm, there are likely to be rough edges for each cell. At this point, the cells may also not even have a standard convex shape. Using refinement procedures to smooth and correct the edges, we can obtain a likelier estimation of the cell boundaries.

To smooth out the edges of the cells, we use iterative morphological erosion and dilation. We employed a disc with a 3-pixel radius and 6 line-structuring elements as the morphological structuring element. In morphological erosion, pixels corresponding to sharper edges are set to be outside of the cell whereas in morphological dilation, more concave grooves become filled. This procedure enables edges to be smoothed out and not necessarily be defined only by the edges of the superpixels. Consequently, overlaps between the cells is reintroduced by the morphological erosion and dilation.

4 Results

We evaluated our cell segmentation algorithm on a test set of 810 real and synthetic Pap smear images from the ISBI 2014 challenge. We manually tuned our hyperparameters, which includeed various thresholds to determine nuclei and superpixel significance and smoothing parameters (we highlight this as an area of improvement in "Future Work").

4.1 Evaluation Metric

To evaluate our method, we used the Dice Coefficient (DC) defined by Radau et al [12], which compares the relative overlap between two binary matrices. The DC coefficient was computed for each individual cell (recall that each image contained between 2-10 overlapping cervical cells). Note that A and B refer to the ground truth and predicted cytoplasm matrices (1 = cytoplasm, 0 = not cytoplasm) for an individual cell. The average DC, computed across all cells within the 810 test set images, provides a quantitative comparison for individual cell cytoplasm segmentation between the ground truth and predicted annotations.

$$DC = \frac{2\,|A \cap B|}{|A| + |B|} \tag{4}$$

Radau et al. defines a "good" cytoplasm segmentation as having a DC > 0.7. As such, we also measured the object-based false negative rate (FNo), which corresponds to the proportion of cells having a DC $<= 0.7$. In addition, we conducted pixel-based evaluation using the true positive rate (TPR) and false positive rate (FPR) across "good" cell segmentations.

The evaluation code was provided by the ISBI challenge organizers [10].

4.2 Performance

Qualitatively, we see that our algorithm is able partition crowded cell masses, even in the case of 9-10 overlapping cells [Figure 2, Figures 3,4,5,6 in Appendix]. Quantitatively, in relation to previously published work, our results are comparable across the four metrics [5, 6, 7, 8]. Our algorithm performed poorly on the object-based false negative rate, which corresponds with our qualitative observation that cell mass substantially lighter than surrounding cytoplasm is frequently removed during thresholding. This is likely a result of a threshold that is too strict. Our algorithm succeeds primarily in the realm of computation speed and resources. Our pipeline employs a number of heuristics. But as a tradeoff for an incremental decrease in accuracy, our algorithm is 3-folds faster even when using substantially less computing power.

Table 1: Quantitative results of cytoplasm segmentation on the test set

Author	DSC	TPR	FPR	FNo	Time Per Image	Computer Specs
Ushizima [5]	0.87	0.83	0.001	0.17	12s	Cray XC30, CPU 2.4 GHz, 64 GB RAM
Nosrati [6]	0.87	0.90	0.005	0.14	16.7s	PC, CPU 3.40 GHz, 16 GB RAM
Lu [7]	0.88	0.92	0.002	0.21	1000.9s	PC, CPU 2.7GHz , 40 GB RAM
Nosrati [8]	0.88	0.93	0.005	0.11	6.6s	PC, CPU 3.40 GHz, 16 GB RAM
Our Method	**0.8442**	**0.8739**	**0.0052**	**0.2249**	**2.17s**	**MacOS, CPU 2.40 GHz, 8 GB RAM**

5 Discussion

Based on the performance results shown above, we see that our proposed algorithm enjoys nearly as good accuracy as state-of-the-art approaches. However, the key advantage of our algorithm is the significant decrease in requirement of computational power and latency. By employing a number of heuristic approaches, our algorithm is able to bypass steps that were more computationally expensive in other algorithms.

One of the challenges we faced during the implementation of this project was the cascading effect of mistakes along the course of the pipeline. For example, our algorithm may not be able to detect a certain nucleus if it is more "dilute" and larger, yet significantly lighter in color than the remaining nuclei. As a result, when the cytoplasm segmentation algorithm attempts to estimate cell boundaries, an entire cell may be missing from the estimation. Consequently, the surrounding cells are also affected, being made artificially bigger by picking up the additional mass that originally belonged to the missing nucleus. One way to deal with this issue is to loosen the threshold for Otsu's method, while creating a more precise heuristic approach of rejecting false nuclei. This would allow the more "dilute" nuclei to be passed off as nuclei and take into account the natural increase in false positives due to the looser threshold. Generally, because later steps depend on the accuracy of previous steps, it is important to maintain the cell mass and nuclei detection so that cytoplasm segmentation does not amplify cascaded mistakes.

Another one of the challenges we faced was the liberal policy of the morphological erosion algorithm during the refinement of cell boundaries. While the morphological erosion algorithm did succeed in smoothing out the edges, it failed to consider the actual boundaries of the extracted cell mass and the grayscale gradients inside the cell mass itself. As a result, the outer boundaries of the refined cell annotations do not correspond as well to the actual extracted cell mass as the preliminary, pre-refinement boundaries. By employing a morphological structuring element that is less restrictive, we can fine-tune the final boundaries so that they remain close enough to the actual cell mass boundaries while still allowing individual cells to overlap. A number of interesting edge refinement approaches have been documented.

In particular, Boykov et. al. [11] introduces a graph cut segmentation algorithm that operates on the raw image. It visualizes the image as a network of pixels with corresponding edge costs, and finds a cut that divides the pixels into two clusters that minimizes edge costs, with the final two clusters corresponding to the object and background respectively. For each initial boundary of a cell, with each cell acting independently, graph cut segmentation could be iteratively performed to obtain finer edges.

5.1 Future Work

The computational problem of overlapping cell segmentation extends beyond cervical cell partitioning in the context of Pap smear for early diagnosis of cervical cancer. For example, an automated algorithm would facilitate the diagnosis of prostate cancer from histopathology specimens [13] and the identification of epithelial nuclei, stroma, and background regions from breast microarrays [14]. A logical next step would be to assess the performance of our algorithm on different image types maintaining the same four metrics.

A number of steps in our pipeline are dependent on hyperparameters. As we tuned various thresholds and model parameters manually, we came across a moderate degree of variance in our results, particularly when tuning significance thresholds to determine whether proposed nuclei or cytoplasm boundaries were valid. To create a more robust pipeline, we need to tune hyperparameters on an explicit training set or use k-folds cross validation on a single test set. On the training or cross-validation data set, the objective would be to maximize DC score. This semisupervised approach applies to any overlapping cell problem. Given a new image type, our algorithm will be able to optimize the hyperparameters to best reflect the nuances of target cells.

Another large area for future development is decreasing the computation time of our algorithm, either by parallelizing code or making use of GPU Computing through NVIDIA CUDA. After all, many components of our pipeline involve repetitive matrix operations.

5.2 Contributions

All authors contributed equally in the development of the code and the writing of the report.

References

[1] World Cancer Report 2014. World Health Organization. 2014. pp. Chapter 5.12. ISBN 9283204298.

[2] Kumar V, Abbas AK, Fausto N, Mitchell RN (2007). Robbins Basic Pathology (8th ed.). Saunders Elsevier. pp. 718–721. ISBN 978-1-4160-2973-1.

[3] Canavan TP, Doshi NR (2000). "Cervical cancer". Am Fam Physician. 61 (5): 1369–76. PMID 10735343.

[4] "Cervical Cancer Prevention (PDQ®)". National Cancer Institute. 2014-02-27. Retrieved 25 June 2014.

[5] AGC Bianchi DM Ushizima and CM Carneiro, "Segmentation of subcellular compartiments combining superpixel representation with voronoi diagrams," in Overlapping Cervical Cytology Image Segmentation Challenge - IEEE ISBI, pp. 1–2. 2014.

[6] MS Nosrati and GHamarneh, "A variational approach for overlapping cell segmentation," in Overlapping Cervical Cytology Image Segmentation Challenge - IEEE ISBI, pp. 1–2. 2014.

[7] Z Lu, G Carneiro, and AP Bradley, "Automated nucleus and cytoplasm segmentation of overlapping cervical cells," in MICCAI, pp. 452–460. 2013.

[8] M. Nosrati and G. Hamarneh, "Segmentation of overlapping cervical cells: A variational method with star-shape prior," in IEEE ISBI, April 2015, pp. 186–189.

[9] Zhi Lu, Gustavo Carneiro, Andrew P. Bradley, Daniela Ushizima, Masoud S. Nosrati, Andrea G. C. Bianchi, Claudia M. Carneiro, and Ghassan Hamarneh. Evaluation of Three Algorithms for the Segmentation of Overlapping Cervical Cells. IEEE Journal of Biomedical and Health Informatics (J-BHI). Jan 2015 (Accepted).

[10] Zhi Lu, Gustavo Carneiro, and Andrew P. Bradley. An Improved Joint Optimization of Multiple Level Set Functions for the Segmentation of Overlapping Cervical Cells. IEEE Transactions on Image Processing. Vol.24, No.4, pp.1261-1272, April 2015.

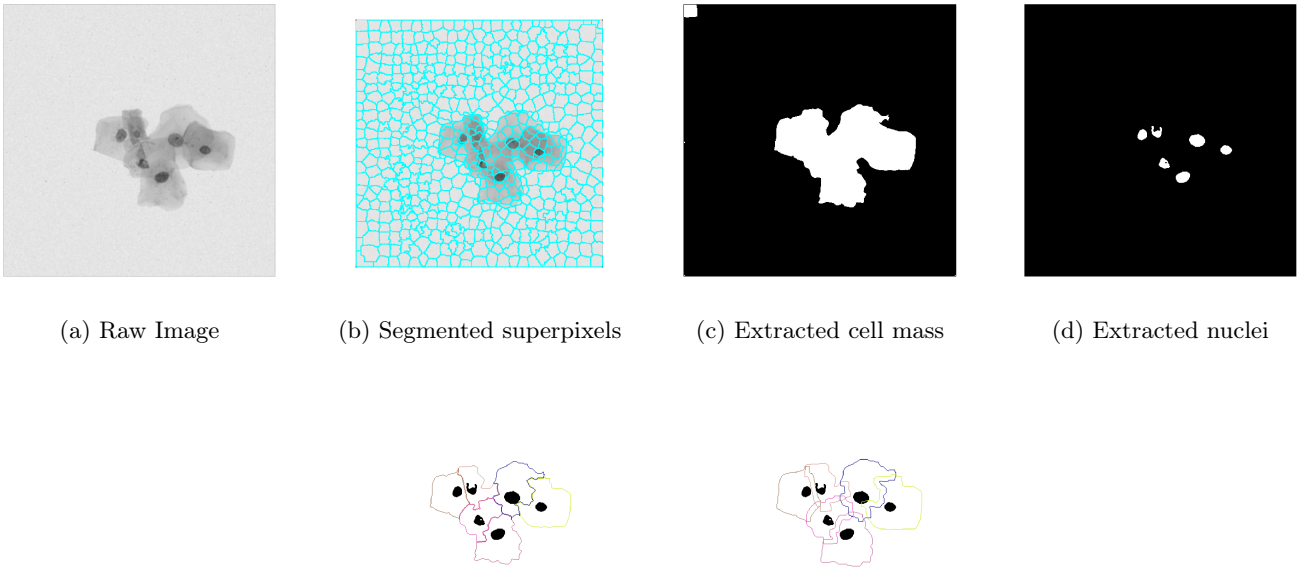
[11] Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary amp; region segmentation of objects in n-d images. In Proc. IEEE Intl. Conf. Computer Vision, volume 1, pages 105–112 vol.1, 2001.

[12] Radau, P., Lu, Y., Connelly, K., Paul, G., Dick, A., Wright, G. Evaluation framework for algorithms segmenting short axis cardiac MRI. The MIDAS Journal - Cardiac MR Left Ventricle Segmentation Challenge (2009)

[13] M. Datar, D. Padfield, and H. Cline, "Color and texture based segmentation of molecular pathology images using hsoms," IEEE International Symposium on Biomedical Imaging, pp. 292–295, 2008.

[14] T. Amaral, S. McKenna, K. Robertson, and A. Thompson, "Classification of breast-tissue microarry spots using colour and local invariants," IEEE International Symposium on Biomedical Imaging, pp. 999–1002, 2008.

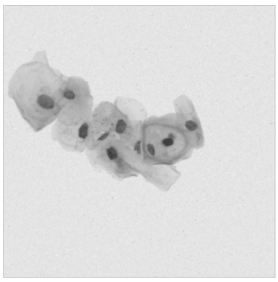
6 Appendix



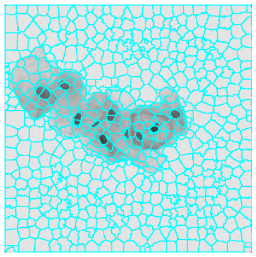
(a) Raw Image (b) Segmented superpixels (c) Extracted cell mass (d) Extracted nuclei

(e) Preliminary boundaries (f) Refined final boundaries

Figure 3: Another example of our complete pipeline for cell segmentation

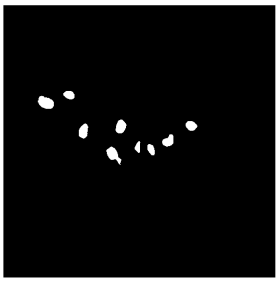


(a) Raw Image

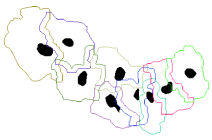


(b) Segmented superpixels

(c) Extracted cell mass

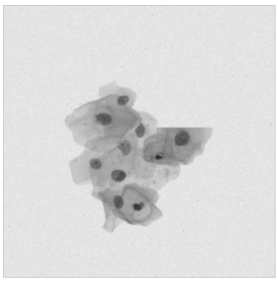


(d) Extracted nuclei

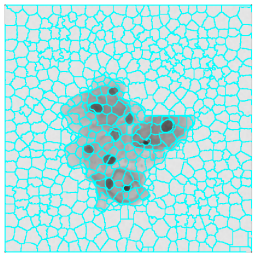


(e) Preliminary boundaries (f) Refined final boundaries

Figure 4: Another example of our complete pipeline for cell segmentation

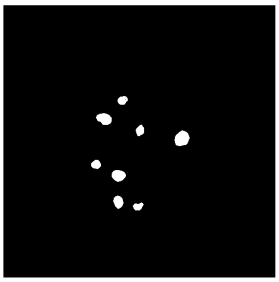


(a) Raw Image

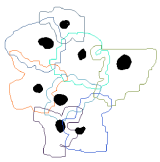


(b) Segmented superpixels

(c) Extracted cell mass

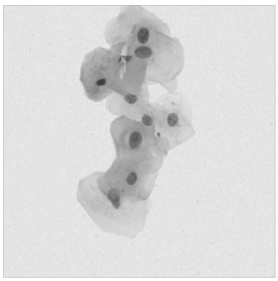


(d) Extracted nuclei

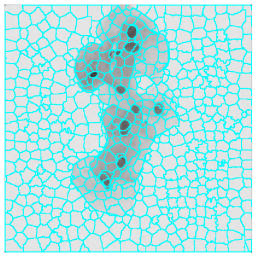


(e) Preliminary boundaries (f) Refined final boundaries

Figure 5: Another example of our complete pipeline for cell segmentation

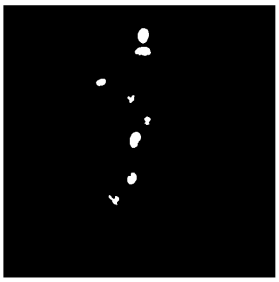


(a) Raw Image



(b) Segmented superpixels

(c) Extracted cell mass



(d) Extracted nuclei

(e) Preliminary boundaries

(f) Refined final boundaries

Figure 6: Another example of our complete pipeline for cell segmentation