

RESEARCH ARTICLE

NuSeT: A deep learning tool for reliably separating and analyzing crowded cells

Linfeng Yang^{1,2,3,4}, Rajarshi P. Ghosh^{1,2,3,4✉*}, J. Matthew Franklin^{1,2,3,4,5}, Simon Chen^{1,6}, Chenyu You^{1,7}, Raja R. Narayan^{1,8}, Marc L. Melcher^{1,8}, Jan T. Liphardt^{1,2,3,4*}

1 Bioengineering, Stanford University, Stanford, CA, United States of America, **2** BioX Institute, Stanford University, Stanford, CA, United States of America, **3** ChEM-H, Stanford University, Stanford, CA, United States of America, **4** Cell Biology Division, Stanford Cancer Institute, Stanford, CA, United States of America, **5** Chemical Engineering, Stanford University, Stanford, CA, United States of America, **6** Department of Pathology, Stanford University School of Medicine, Stanford, CA, United States of America, **7** Electrical Engineering, Stanford University, Stanford, CA, United States of America, **8** Department of Surgery, Stanford University School of Medicine, Stanford, CA, United States of America

✉ Current address: Howard Hughes Medical Institute and Department of Molecular and Cell Biology, University of California, Berkeley, CA, United States of America

* ghosh.rpg@gmail.com (RPG); jan.liphardt@stanford.edu (JTL)



OPEN ACCESS

Citation: Yang L, Ghosh RP, Franklin JM, Chen S, You C, Narayan RR, et al. (2020) NuSeT: A deep learning tool for reliably separating and analyzing crowded cells. PLoS Comput Biol 16(9): e1008193. <https://doi.org/10.1371/journal.pcbi.1008193>

Editor: Anand R. Asthagiri, Northeastern University, UNITED STATES

Received: January 29, 2020

Accepted: July 25, 2020

Published: September 14, 2020

Copyright: © 2020 Yang et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: All the code and pretrained models have been released on GitHub with MIT license. Detailed instructions have been included in [S1 Text](#) as well in GitHub so that researchers can easily apply them in practice. Model weights and training dataset used in this work are provided at <https://zenodo.org/record/3996370#.X0aXi9MzbsI>. Please download the NuSeT repository at: <https://github.com/yanglf1121/NuSeT>.

Funding: LY, RPG, JMF, and JTL received funding from the National Institutes of Health (NIH)

Abstract

Segmenting cell nuclei within microscopy images is a ubiquitous task in biological research and clinical applications. Unfortunately, segmenting low-contrast overlapping objects that may be tightly packed is a major bottleneck in standard deep learning-based models. We report a Nuclear Segmentation Tool (NuSeT) based on deep learning that accurately segments nuclei across multiple types of fluorescence imaging data. Using a hybrid network consisting of U-Net and Region Proposal Networks (RPN), followed by a watershed step, we have achieved superior performance in detecting and delineating nuclear boundaries in 2D and 3D images of varying complexities. By using foreground normalization and additional training on synthetic images containing non-cellular artifacts, NuSeT improves nuclear detection and reduces false positives. NuSeT addresses common challenges in nuclear segmentation such as variability in nuclear signal and shape, limited training sample size, and sample preparation artifacts. Compared to other segmentation models, NuSeT consistently fares better in generating accurate segmentation masks and assigning boundaries for touching nuclei.

Author summary

Nuclear size and shape are essential indicators of cell cycle stage and cellular pathology. Efficient segmentation of nuclei in complex environments, especially for high-value yet low-quality samples is critical for detecting pathological states. In the majority of cases, biological features are still segmented using traditional segmentation methods requiring manual curation of segmentations, which is hugely time-consuming and does not achieve optimal performance. While a recent surge in deep learning tools has helped tremendously with the automation of segmentation tasks, existing platforms inefficiently segment

National Institute of General Medical Sciences (NIGMS)/National Cancer Institute (NCI) Grant GM77856, NCI Physical Sciences Oncology Center Grant U54CA143836, and National Institute of Biomedical Imaging and Bioengineering (NIBIB)/4D Nucleome Roadmap Initiative 1U01EB021237. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

nuclei in crowded cells with overlapping nuclear boundaries. NuSeT, assimilates the advantages of semantic segmentation (U-Net) and instance segmentation (Mask R-CNN), and consistently outperforms other start-of-the-art deep learning segmentation models in analyzing complex three-dimensional cell clusters and in tracking nuclei in crowded, dynamic environments. NuSeT can work with both fluorescent and histopathology image samples. We have also developed a graphic user interface for customized training and segmentation, that will aid considerably in the ease and accuracy of image segmentation in a wide range of image types.

This is a *PLOS Computational Biology* Methods paper

Introduction

Quantitative single-cell analysis can reveal novel molecular details of cellular processes relevant to basic research, drug discovery, and clinical diagnostics. For example, cell morphology and shape are reliable proxies for cellular health and cell-cycle stage, as well as indicating the state of disease-relevant cellular behaviors such as adhesion, contractility, and mobility.[1–5] However, accurate segmentation of cellular features such as the size and shape of the nucleus remains challenging due to large variability in signal intensity and shape, and artifacts introduced during sample preparation.[6,7] These challenges are exacerbated by cellular crowding, which juxtaposes cells and obscures their boundaries. Additionally, in many traditional segmentation methods[8], parameters need to be iteratively adjusted for images varying in quality.[9]

Convolutional neural networks (CNN) have emerged as a robust alternative to traditional segmentation methods for segmenting cell nuclei.[10–16] CNNs achieve their superior performance through new deep-learning models.[10,17–19] CNNs' applicability for high precision image segmentation was first demonstrated by a Fully Convolutional Network (FCN) for pixel-level segmentation.[10] Additional FCN cell segmentation models have since been developed.[14,20,21] These pioneering approaches established a basic pipeline for CNN-based nuclear segmentation and achieved significant improvements in segmenting different types of cells including bacteria and mammalian cells.[14,21] However, in their original form, FCNs typically required large training datasets to achieve high levels of accuracy.[10] This bottleneck was overcome in U-Net by introducing a U-shaped network that incorporates pooling layers and up-sampling layers.[15] Additionally in U-Net, the network was guided to segment overlapping objects by introducing weight matrices at cell-boundaries. Several state-of-the-art nuclear segmentation models have since been developed using this architecture.[11,13,22,23] Several online cell segmentation interfaces allow users to predict and train on their own image data, facilitating front end use by researchers.[23,24] However, U-Net and FCN based models are curated and evaluated on pixel-level accuracy, where each pixel is segmented directly without the object detection step. In cell biology, the main goal is to make reliable statements about *cells* as a whole (e.g. the number of cells, their average size and shape, detection of rare/unusual cells) rather than focusing on image pixels. For such problems, the idea of instance segmentation provides a more effective solution, as the loss function incorporates a sense of the whole object and not just individual pixels. One such approach, the Deep watershed transform[25], incorporates the object by learning a distance transform computed from the original training masks. The distance transform is further fed into a watershed layer to have the final segmentation results. A recent improvement is to incorporate a Faster R-CNN detection module. In this

approach, the algorithm computes object locations and uses them as markers for the watershed layer, improving the segmentation.[26] Another approach, Mask R-CNN[19], applies FCN-based segmentation to regions proposed by Region Proposal Networks (RPN) and achieves good segmentation results in real-world image datasets. A more recent implementation of this approach replaces the RPN with a single shot detection module[27], achieving superior performance in segmenting and tracking cells and nuclei.[28,29] However, the performance of Mask R-CNN based approaches remains to be validated for images with high cell density. Mask R-CNN also employs fixed anchor scales for bounding boxes across all images, which is a limitation for samples with variable-sized nuclei.[18,19] Additionally, at the pixel-level, the segmentation task of Mask R-CNN is performed by FCN, which is less accurate with small training datasets compared with U-Net.[15,30]

To address these issues, we have developed a Nuclei Segmentation Toolset (NuSeT), which integrates U-Net[15] and a modified RPN (based on the implementation of previous works [31,32]) to accurately segment fluorescently labeled nuclei. In this integrated model, U-Net performs pixel-segmentation, while the modified RPN predicts unique bounding boxes for each image based on U-Net segmentations. The resulting output provides seeds for a watershed algorithm to segment touching nuclei. To minimize segmentation errors stemming from fluorescence signal variability and cell density variability in samples, we employed a novel normalization method that uses only foreground pixel intensities for image normalization. To increase the robustness and applicability of the model, we used training sets including samples with wide variations in imaging conditions, image dimensions, and non-cellular artifacts. Extensive qualitative and quantitative evaluation suggest that our segmentation pipeline significantly improves nuclei segmentation, especially in distinguishing overlapping boundaries, and is generalizable to both fluorescent and histopathological images.

Results

NuSeT is a robust nuclear segmentation tool

Tools for segmenting fluorescent nuclei need to address multiple features and limitations of biological images.[6,33] Typical issues and limitations include:

1. Boundary assignment ambiguity: biological samples frequently have very high cell density with significant overlap between objects.
2. Signal intensity variation: Within one image, the signal can vary within each nucleus (e.g. due to different compaction states of the DNA in heterochromatin vs. euchromatin) and across nuclei (e.g. due to cell to cell differences in nuclear protein expression levels and differences in staining efficiency).
3. Non-cellular artifacts and contaminants: Fluorescence microscopy samples are often contaminated with auto-fluorescent cell debris as well as non-cellular artifacts.
4. Low signal to noise ratios (SNRs): Low SNRs typically result from lower expression levels of fluorescent targets and/or high background signal, such as sample autofluorescence. ([S1 Fig](#)).

We used an end-to-end training approach that incorporates both U-Net and Region Proposal Network (RPN)[15,18] to address these issues (Methods). In our approach, the training and inference step consists of running an input image in parallel in both U-Net and RPN. The final output of U-Net consists of two feature maps of the same shape as the input image, representing background and foreground pixel-assignment scores.[10] The final foreground prediction is then computed from the maximum class score of each pixel. Although U-Net alone performs well on some microscopy datasets[30,34], we incorporated RPN since it was

originally designed to detect objects in images with high information content.[18] We reasoned that the accurate performance of RPN in detecting objects can be leveraged to improve nuclear segmentation performance. To achieve robust separation of touching nuclei, we used RPN bounding boxes to determine nuclear centroids, which were then supplied as seeds to a watershed algorithm.[35,36] To improve segmentation accuracy in images with large nuclear size variations, we modified the original RPN architecture to use bounding box dimensions based on average nuclear size for each image (S2 Fig). Instead of training U-Net and RPN separately, we merged the feature-extraction part of RPN with the down-sampling part of U-Net to avoid longer training time and more memory cost (Fig 1A).[10,15,18,19] In this way, the instance detection insights of RPN are extracted from the model structure. To evaluate the segmentation performance of the different algorithms, we computed the mean intersection over union for foreground and background (mean IoU), Root Mean Square Error (RMSE), and pixel accuracy (to benchmark pixel-level performance). Since in biological image processing the primary focus is on cell-level segmentation rather than pixel-level accuracy, we also included object-level segmentation metrics, including the rate of correctly separating overlapping nuclei, correct and incorrect detections, splits, merges, catastrophes, and both the false-positive and false-negative detection rates (Methods).[29,30] Two separate datasets, ‘MCF10A’ and ‘Kaggle’, were used to compare the performance of the algorithms.[33] The MCF10A dataset consists of images of relatively uniformly fluorescent nuclei of a non-tumorigenic breast epithelial cell line[37], grown to different levels of confluence. The Kaggle dataset was adapted from a public dataset[33] representing cells from different organisms (including humans, mice, and flies) and containing images with a wide range of brightness, cell densities, and nuclear sizes. The overall comparison in S1 Table and S2 Table suggests that NuSeT achieves similar pixel-level segmentation accuracy compared with a current state-of-the-art pixel-level cell segmentation approach (U-Net) but has higher separation rates for overlapping nuclei and fewer merge errors. With the Kaggle dataset, NuSeT improved the separation of touching nuclei by more than 75% compared with U-Net. Compared with another state-of-the-art instance segmentation approach, Mask R-CNN, NuSeT achieved much lower false-negative detection rates in Kaggle dataset, leading to significantly better pixel-level segmentation accuracy. To make NuSeT more user-friendly, we have prepared a cross-platform graphic user interface (GUI) for the scientific community. Our GUI comes with the pretrained model which we used to benchmark NuSeT performance for various nuclei segmentation tasks. The GUI also allows the use of training and predicting modules (Fig 1B), allowing the users to perform custom segmentation tasks with NuSeT.

Foreground normalization improves segmentation performance

Normalizing training data to alleviate image intensity differences is central to accelerating learning and improving network performance. Historically, imaging data have been normalized by subtracting the mean intensity calculated from all pixels in a dataset.[38,39] However, this leads to discrepancies in normalization, particularly for images with markedly different brightness levels. Normalizing data at whole-image level addresses the issue of illumination differences[40], but introduces brightness differences in images with sub-regions of varying cell densities (Fig 2A). Additionally, whole-image normalization fares poorly in images strewn with auto-fluorescent artifacts (S3 Fig). We incorporated a foreground normalization step in our data preprocessing. In this approach, only the pixels that belong to cell nuclei (foreground) are selected to calculate mean and standard deviation of pixel intensities. Since no label is provided during inference, foreground normalization requires two passes. In step one, the test data are normalized on a per image level to generate a coarse prediction of the foreground

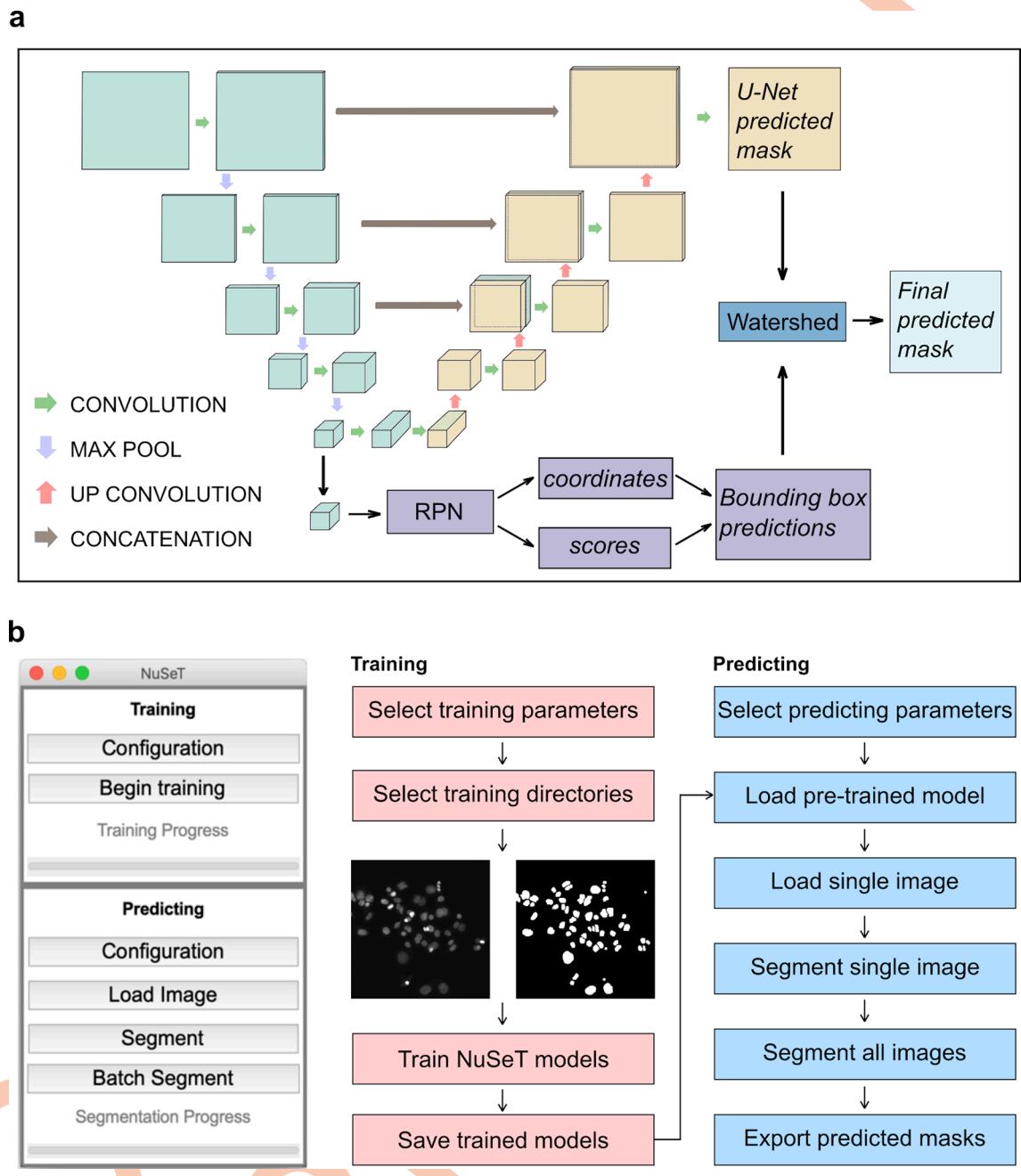


Fig 1. Pipeline for segmenting nuclei with NuSeT. (A) Deep-learning model structure of NuSeT. The inputs of the model are gray scale images with different sizes. The outputs are binary masks with the same size as inputs, with predicted foreground regions as Ones and background regions as Zeroes. The model combines U-Net (gray and orange) and Region Proposal Network (purple), which performs nuclei segmentation and detection separately. The results are then merged and processed by watershed (dark blue) to generate final predictions. (B) Outlook of NuSeT Graphic User Interface(GUI), and example training and predicting pipelines using NuSeT GUI.

<https://doi.org/10.1371/journal.pcbi.1008193.g001>

with our RPN-U-Net fusion. In step two, this coarse prediction is used to perform foreground normalization on test images before they are fed into the model for a second pass (Fig 2B). Compared with whole-image normalization, the two-step foreground normalization approach is relatively robust to illumination differences, cell-density variations, and image artifacts and

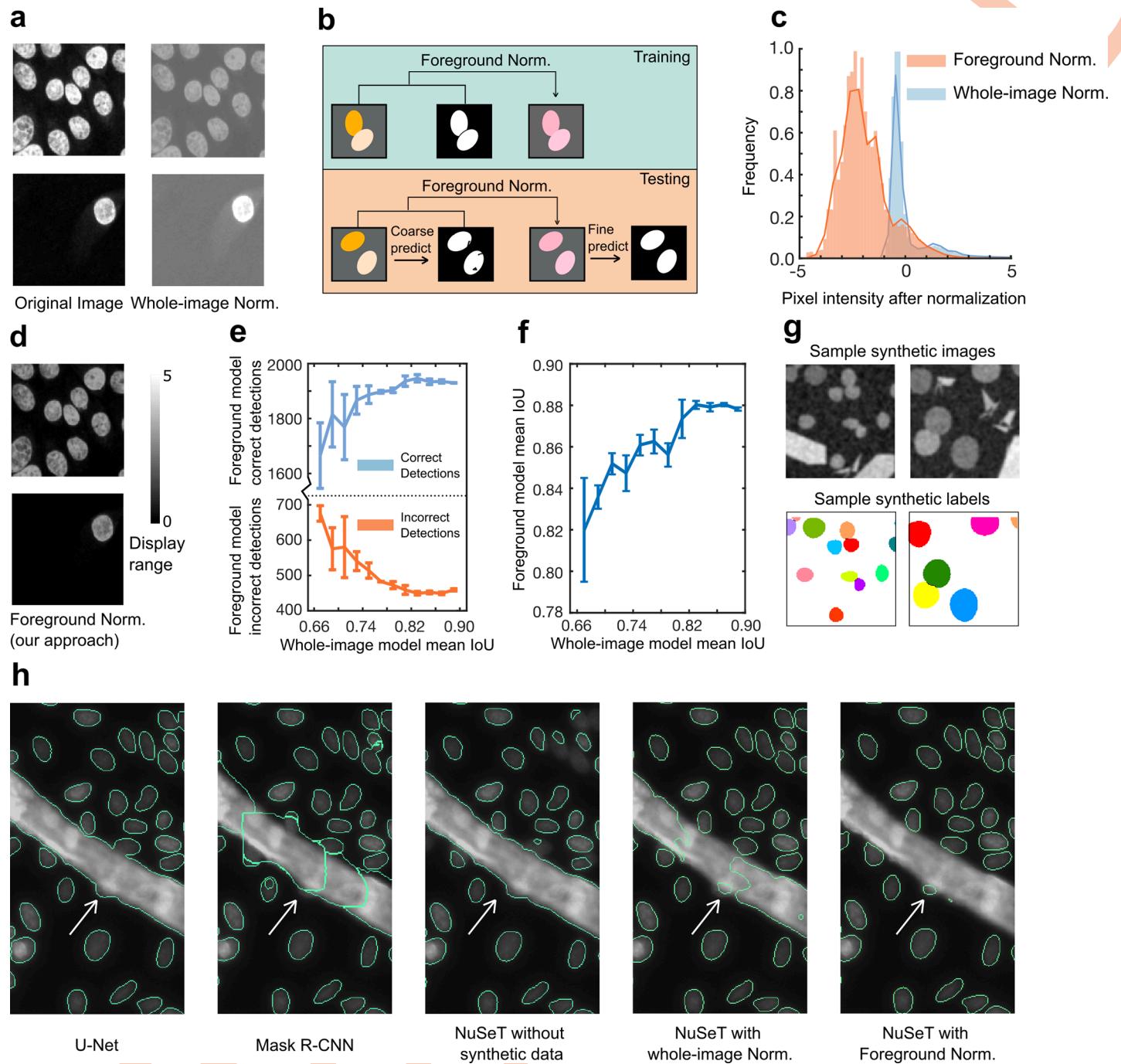


Fig 2. Improved normalization performance by foreground normalization and synthetic training. (A) The visual effects of normalizing sparse/dense samples using whole-image normalization showing images having inconsistent nuclear signals after normalization. (B) Foreground normalization during training and testing. During training, only pixels belonging to cell nuclei are used to normalize the image. During testing, a coarse segmentation prediction is generated by the model, and pixels belonging to the predicted nuclei are used to perform foreground normalization. The model then makes final predictions based on the normalized input images. (C) Distribution of pixel intensities over an entire training dataset after different normalizations, showing foreground normalization has wider dynamic range. (D) The visual effects of normalizing sparse/dense samples using foreground normalization showing images have a higher dynamic range and more consistent nuclear signals. (E, F) Line charts showing that the object-level performance (E) and the pixel-level performance (F) of the foreground normalization model depend on the pixel-level performance of the whole-image normalization model. Error bars represent three individual experiments. (G) Examples of synthetic images with labels used during training. Our algorithm can generate synthetic nuclei-shaped blobs with different sizes, as well as different types of artifacts to increase the robustness of the model. Overlapping nuclei were introduced to enhance NuSeT performance in touching nuclei separation. (H) Representative examples comparing the performances of different segmentation approaches. Training without synthetic data mis-identified artifacts (stripes) as foreground. The addition of synthetic data improved artifact detection. Switching to foreground normalization led the best performance including robust identification of imaging artifact, detected of more nuclei, and better separation of touching nuclei compared to Mask R-CNN and U-Net.

<https://doi.org/10.1371/journal.pcbi.1008193.g002>

performs better in normalizing images with a broader dynamic range of pixel intensities ([Fig 2C and 2D](#)). As a result, model training with foreground normalization increased nuclei detection accuracy and boundary assignment for both Kaggle and MCF10A datasets, with more correct detections and less merge errors ([S1 Table](#)).

To further analyze how whole-image normalization models affect the performance of the foreground normalization model, we trained the whole-image normalization models to different mean IoU levels. This step was essential as the pixel-level accuracy of the whole-image normalization model was critical for selecting pixels to perform the following foreground normalization. By connecting the different whole-image normalization models with the final foreground normalization model, we found that when the mean IoU of the whole-image normalization models were less than 0.82, the performance of the foreground normalization model heavily depended on the whole-image normalization models ([Fig 2E and 2F, S3 Table](#)). This suggests that the performance of the foreground normalization model relies on the accuracy level of the whole-image normalization model. However, when the mean IoU of the whole-image normalization models were higher than 0.82, the foreground normalization model was less affected ([Fig 2E and 2F, S3 Table](#)).

Given the modularity of the foreground normalization approach, we next asked whether foreground normalization could be integrated into other deep learning models, such as U-Net and Mask R-CNN, to enhance their performance. Consistent with our expectation, training U-Net with foreground normalization improved the overlapping nuclei separation performance by 6% to 35% (in MCF10A and Kaggle datasets). Foreground normalization also improved nuclei-detection accuracy of U-Net, and reduced merge errors ([S4 Table](#)). However, the segmentation performance of Mask R-CNN was not significantly improved by foreground normalization. The segmentation performance was almost identical to the model trained with whole-image normalization ([S4 Table](#)). Given that the performance of Mask R-CNN is highly dependent on the detection accuracy of RPN, whereas both NuSeT and U-Net rely heavily on pixel classification to perform segmentation, we concluded that foreground normalization improved the segmentation performance by rescaling the image pixels more consistently, aiding in better classification of foreground and background pixels.

Synthetic datasets in model-training improve detection and segmentation accuracy

Common sample contaminants have irregular shapes, significantly different overall brightness levels and aspect ratios compared to real cells, and uneven pixel intensities. To improve model performance and minimize false-positive detection rates, we computationally generated synthetic images containing irregular shapes with varying intensities, as well as nuclei-like blobs (Methods). We also added Gaussian blur and noise to the synthetic images to better represent real-world images. Additionally, overlapping blobs were included to mimic touching nuclei. Example synthetic images and training labels are shown in [Fig 2E](#). Including synthetic data in the training process notably improved the model's performance in distinguishing real nuclei from imaging artifacts ([Fig 2F](#)) and enhanced the separation of touching nuclei ([S1 Table](#)). The addition of foreground normalization on top of the synthetic images during model-training further reduced false positive detections ([Fig 2F](#)). Aided by these improvements, NuSeT outperformed both U-Net and Mask R-CNN in artifact detection/rejection ([Fig 2F](#)).

RPN-aided Watershed improves boundary-resolution of highly overlapping objects

Having improved nuclear segmentation performance, we revisited the problem of separating overlapping nuclei. Previous studies have used algorithms such as intervening and concave

contour-based normalized cut[41,42] on binary segmentation masks extracted using traditional segmentation methods such as Otsu's method[8] to delineate overlapping nuclear boundaries. However, nuclear segmentation using traditional thresholding approaches failed to detect half of the nuclei in the Kaggle dataset ([S2 Table](#)), indicating that this approach is only effective for images with clean backgrounds and uniform signal. Recent studies have trained deep neural networks to learn the Euclidean distance transform (EDT) of the original mask corresponding to the input images[25,26], and apply a watershed transform on the model-predicted distance map to perform the final segmentation. This method has been further improved by adding the cell location information to the watershed transform to achieve better segmentation results.[26] These methods successfully address the challenges of separating overlapping objects, as EDT provides the neural networks with more morphological information.

Instead of training the model on EDT space, we trained the U-Net module directly with the binary masks. We also employed our modified RPN approach to detect nuclei. The nuclear centroids estimated from the RPN derived bounding box coordinates were passed as seeds for the watershed algorithm to generate cuts at touching nuclei boundaries on the U-Net produced binary masks ([Fig 3A](#)).[35,36]

Our results suggest that a modified RPN can detect most nuclei in overlapping regions, and a RPN-aided watershed separates 72%/94% of overlapping nuclei for Kaggle/ MCF10A dataset ([Fig 3B](#), [S2 Table](#)). Compared with the modified RPN model without watershed, RPN-aided-watershed improved the overlapping nuclei separation performance and lowered the number of merge errors ([S2 Table](#)).

Through the integration of synthetic images, foreground normalization, and RPN-aided watershed, NuSeT consistently outperforms other state-of-the-art segmentation methods including U-Net and Mask R-CNN in nuclear boundary demarcation, particularly for blurry, low SNR nuclei ([Fig 3C](#), [S4 Fig](#), [S2 Table](#)). Mask R-CNN and NuSeT perform comparably in relatively sparse and homogenous samples ([S2 Table](#)). However, NuSeT approximates ground-truth boundaries more closely than U-Net and Mask R-CNN in samples with high cell densities ([Fig 3D](#) and [3E](#)).

Three-dimensional spatio-temporal tracking of individual nuclei in mammary acini

To investigate the performance of our algorithm in segmenting densely packed nuclei, we used NuSeT to segment and track nuclei in 3D reconstituted mammary acini grown from a Ras transformed MCF10A (MCF10AT) cell line. MCF10AT was chosen since upon continued growth in matrigel, this cell line produces mammary acini with very high cell density. Three-dimensional segmentation was performed by processing individual 2D slices from a Confocal Microscope Z-stack followed by three-dimensional reconstruction. NuSeT successfully segmented most of the nuclei in an acinus ([Fig 4A](#)), which facilitated seamless tracking of nuclei in mammary acini disorganizing on a 3D collagen matrix ([Fig 4B and 4C](#)). Both NuSeT and Mask R-CNN performed similarly on early-stage mammary acini (cell count = ~34 cells/acinus) ([S5 Fig](#)). To further evaluate the performance of different algorithms (NuSeT, U-Net, Mask R-CNN and Otsu's method) on segmenting nuclei in mammary acini, we carried out nuclear segmentation on 2D projections of dense mammary acini.

NuSeT accurately segmented most of the nuclei in dense mammary acini ([Fig 4D–4G](#)). We were also able to track single nuclei through the entire process of acinar disorganization (*time* = 17.5 h, [S5 Fig](#)). Compared with the other widely used segmentation models, NuSeT performed consistently better at matching the number of nuclei detected manually for multiple

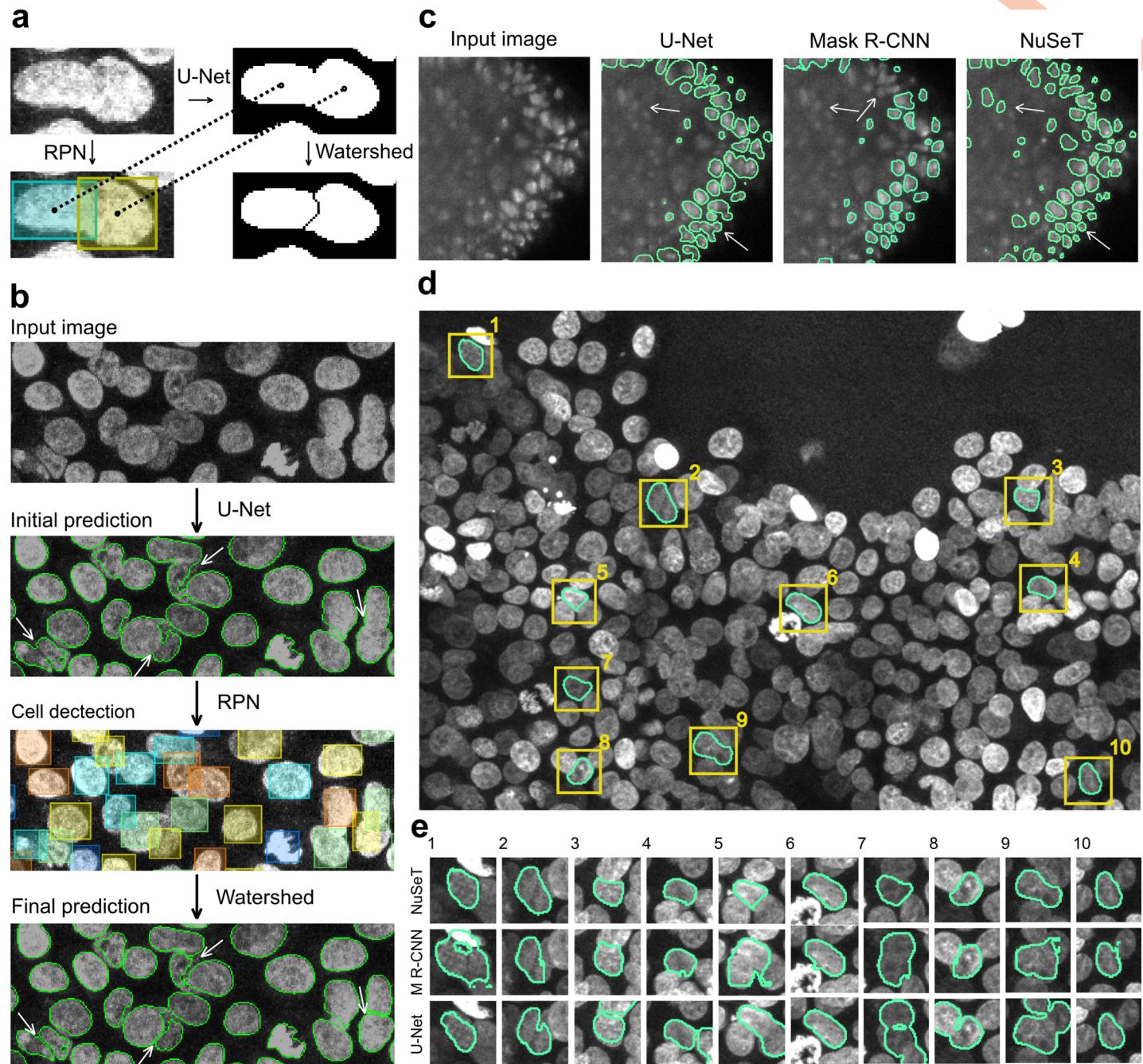


Fig 3. NuSeT efficiently addresses common segmentation challenges. (A) Implementing RPN-aided watershed algorithm improves touching cell separation. Bounding boxes and segmentation masks are computed by RPN and U-Net. Then the estimated centroid of each cell is computed from the coordinates of the bounding box. The watershed line is then estimated based on the binary mask and centroids. (B) Sample results showing that RPN successfully detects most of the cells, and watershed lines further separate touching cells. (C) Representative examples showing NuSeT detected more nuclei and better separated touching nuclei compared to Mask R-CNN and U-Net. (D,E) Examples nuclear masks generated using NuSeT for an image with high nuclei density (D). Comparison with the corresponding masks generated by Mask R-CNN and U-Net show subtle as well as prominent irregularities in boundary delineation that are circumvented by NuSeT (E).

<https://doi.org/10.1371/journal.pcbi.1008193.g003>

acini ($n = 8$) at different stages of disorganization. U-Net, Otsu's Method and Mask R-CNN all detected only a fraction of all the nuclei (Fig 4D) in a dense acinus. The distribution of areas of segmented nuclei ($n = 1365$ nuclei) across multiple acini ($n = 8$) at first 5 time points shows

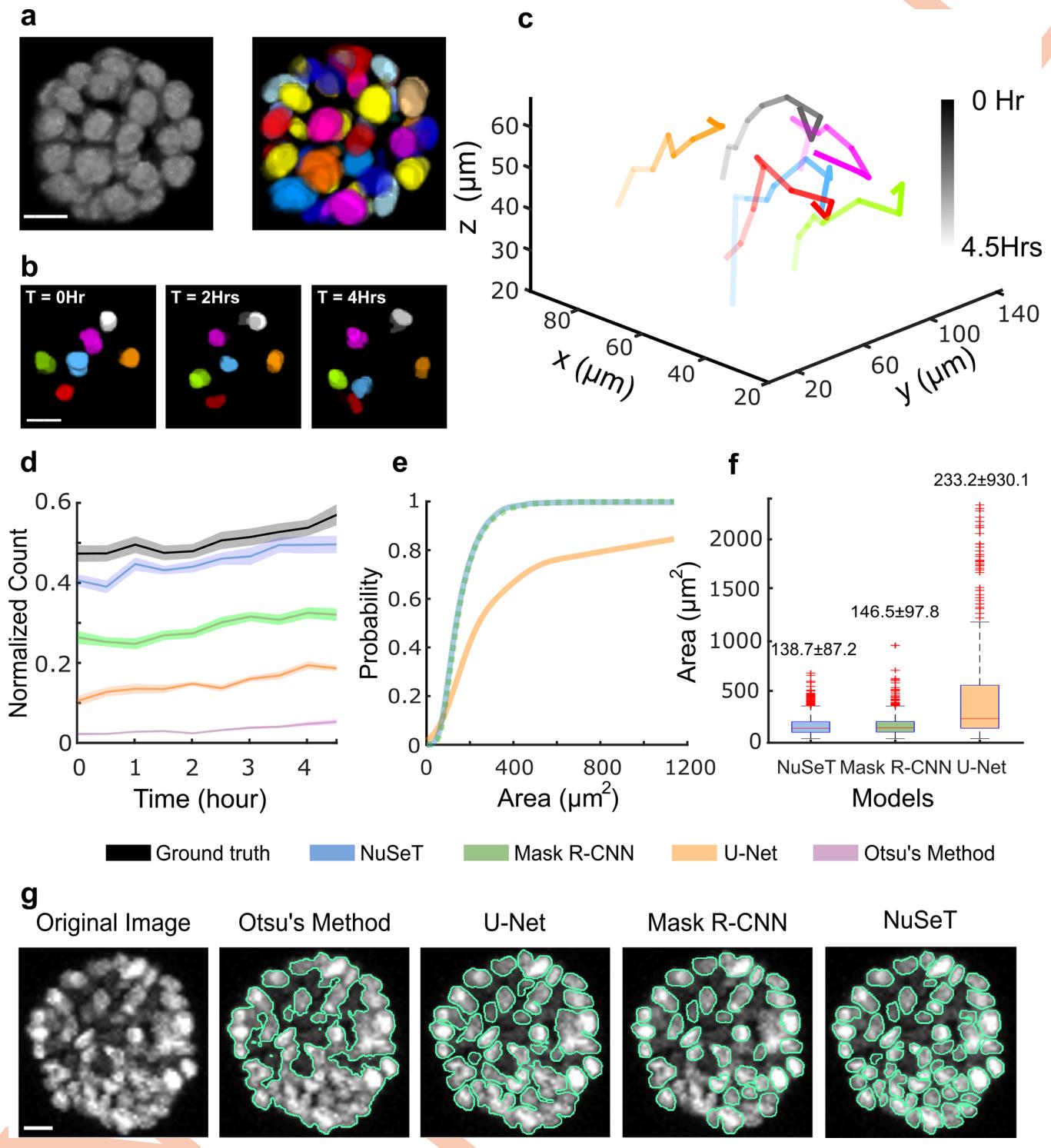


Fig 4. NuSeT effectively segments single nuclei in disorganized dense mammary acini. (A) Representative 3D MCF10AT acinus segmentation using NuSeT. (B) Nuclei tracking. For ease of visualization, only a few of the segmented nuclei are shown at different time points. (C) 3D tracks of the nuclei shown in (B) over time, from 0 h (dark) to 4.5 h (light). (D) Number of nuclei detected in disorganized acini at different time points using different segmentation methods. Data were collected from 8 representative acini and were normalized by the total number of nuclei at the last time point. Data from the first 5 hours are shown. (E) Cumulative distribution function plots of area of nuclei segmented using different methods. (F) Box plots of nuclear area distribution. The median area for each method is indicated on the top. The area box plot for Otsu's method (median area: $2816.6 \pm 2845 \mu\text{m}^2$) is shown in S6 Fig. (G) Representative examples comparing nuclei segmentation in dense mammary acini using different methods. Scale bars are 20 μm .

<https://doi.org/10.1371/journal.pcbi.1008193.g004>

that while Mask R-CNN and NuSeT achieved comparable accuracy in nuclear boundary determination (median area of detected nuclei: $147 \mu\text{m}^2$ vs. $139 \mu\text{m}^2$, [Fig 4E and 4F](#)), Mask R-CNN only detected a subset of all nuclei ([Fig 4D and 4G](#)). Nuclear segmentation with U-Net on the other hand resulted in much larger nuclear area (median area of detected nuclei = $233 \mu\text{m}^2$, [Fig 4E and 4F](#)), indicating that U-Net often failed to separate touching nuclei ([Fig 4G](#)). All deep learning approaches outperformed the ‘traditional’ algorithm (Otsu’s Method, nuclei area = $2816.6 \mu\text{m}^2$, [S5 Fig](#)), as it rarely segmented single nuclei in dense settings ([Fig 4G](#)). Together, our results suggest that NuSeT outperforms both Mask R-CNN and U-Net in detecting nuclei and assigning boundaries for overlapping nuclei.

Segmentation of histopathology samples and dividing cells

To further validate the performance and assess the generalizability of our algorithm, we extended NuSeT based segmentation to histopathology samples and rare-event detections as in the case of dividing cells.

As a test case for segmentation of histological samples, we re-trained NuSeT to segment fat globules in H & E stained sections of liver tissue. Evaluation of liver steatosis is a key step in both fatty liver disease diagnosis as well as pre and post-liver transplantation evaluation. The key challenges of segmenting fat globules from liver sections include detecting multi-scale globules and distinguishing them from tissue tearing artifacts. NuSeT successfully segmented both micro and macro-globules and avoided false detection of tissue tearing artifacts ([Fig 5A and 5B](#)), with mean IoU = 0.73 on a validation dataset.

Detecting and segmenting rare events in images are more challenging, as the majority area of the image is denoted as background, and back-propagation of gradients will overwhelm the model in classifying simple background pixels. Mitotic events, especially in images populated densely with non-dividing cells, is an example of such rare-event detection. To address this challenge, we designed an approach to highlight the regions close to mitotic events and give them more weights to ‘catch’ the attention of the model during training ([Fig 5C](#)). Using this strategy, we retrained the NuSeT model to detect and segment mitotic nuclei in human breast cancer histopathology samples [[43,44](#)], as the total number of mitotic events detected is a crucial indicator of the degree of malignancy for breast cancer diagnosis. Our results indicate that NuSeT can detect and segment the majority of mitosis events in breast cancer histopathology slides ([Fig 5D and 5E](#)), and was able to provide confidence scores for all the detected mitotic events (detection precision = 56.22%, recall = 58.85% on validation dataset). When we inspected the data, we found several detection errors stemming from mis-classification of other objects such as dense nuclei and lymphocytes, which are very similar in appearance to mitotic nuclei. When trained with fluorescently labeled nuclei (MCF10A cells stably expressing the nuclear marker histone H2B-eGFP), NuSeT captured the mitotic progression from prophase to telophase ([Fig 5D and 5E, S7 Fig](#)) (detection precision = 73.90%, recall = 90.20% on validation dataset). Together our results indicate that NuSeT is highly generalizable and can be applied to histopathology segmentation tasks as well as detection of rare events in samples of high clinical value.

Discussion

Here we present a deep learning model for nuclear segmentation that is robust to a wide range of image variabilities. Compared with previous models that need to be trained separately for specific cell types, NuSeT provides a more generalized approach for segmenting fluorescent nuclei varying in size, brightness and density. We have also developed novel training and pre/post-processing approaches to address common problems in biological image processing. Our

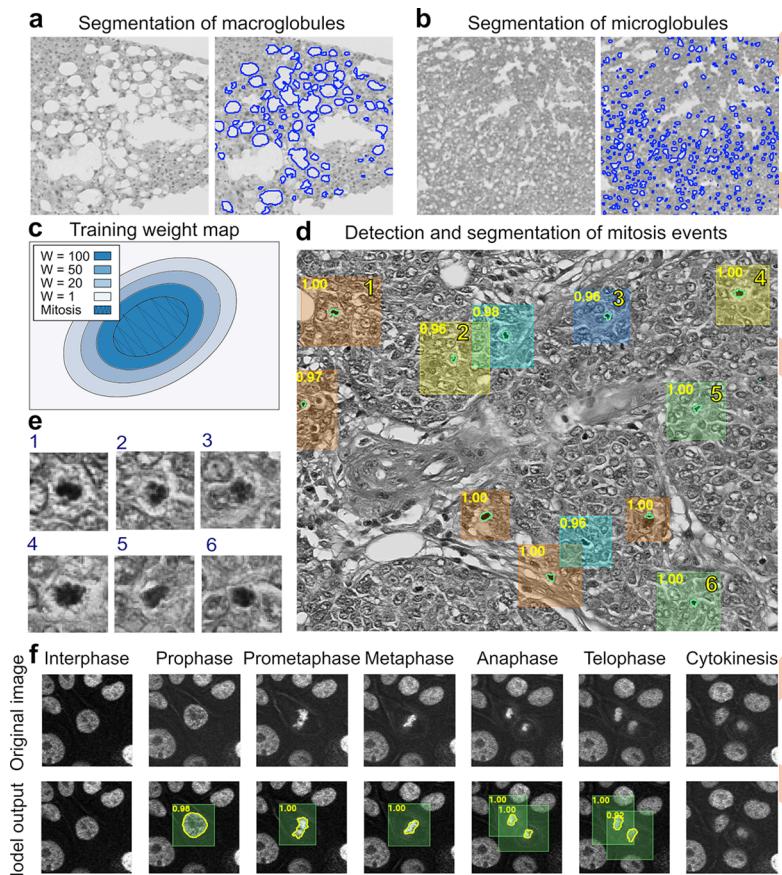


Fig 5. NuSeT can be applied to segment histopathology samples and detect mitotic events. (A, B) Representative example of liver fat globule segmentation using NuSeT. Notice that NuSeT performs well on both macroglobule (A) and microglobule (B) segmentation. (C-E): Representative example of segmentation and detection of mitosis in breast cancer samples from ICPR 2012. (C) Weight map used for training the mitosis model. An ‘attention’ strategy has been used to focus more on the mitotic events and the environment surrounding them. The shaded region denotes the label for the mitotic event, and colors denote the weights applied during training process. (D) Representative example of mitosis detection and segmentation results with breast cancer sample. Scores on the top-left corner of the bounding boxes denote the possibility of a mitosis event evaluated from the model. Zoom-in of some detected mitotic events are shown in (E). (F) Representative example of mitosis detection and segmentation results with fluorescent nuclei in a time lapse movie of MCF10A epithelial cells stably expressing histone H2B-eGFP. NuSeT can detect mitotic progression from prophase to telophase (mitotic events were identified by NuSeT and then manually classified into different phases).

<https://doi.org/10.1371/journal.pcbi.1008193.g005>

results indicate that every stage in deep learning, from data collection to post-processing, is crucial to training an accurate and robust nuclear segmentation model. When compared with the state-of-the-art cell segmentation models, NuSeT separates touching nuclei better than U-Net and detects more nuclei than Mask R-CNN. Thus, it assimilates the advantages of both semantic segmentation (U-Net) and instance segmentation (Mask R-CNN) and circumvents their limitations. This combination enables NuSeT to analyze complex three-dimensional cell clusters such as mammary acini and track single nuclei in dynamic crowded environments. When retrained on histopathology images, NuSeT is able to segment cells and rare events in H&E Stained samples using new training data. Therefore, we expect NuSeT to find wide applicability, particularly in the areas of cell lineage tracing and clinical diagnosis.

Although we have modified the original RPN architecture to adjust detection scales based on the median nucleus size for each image, NuSeT assumes similar nuclear sizes in the same

image. This may account for the occasional errors in nuclei segmentation when using RPN-aided watershed. If markedly irregular (such as dim/deformed/blurry) nuclei are encountered in the same image, RPN may over- or under-detect the nuclei and produce incorrect numbers of bounding boxes. This would lead to marker misplacement and erroneous segmentation lines. While we expect NuSeT to perform well for nuclei of most mammalian cell types, its performance for mixed populations remains to be validated. Recent studies have extracted image features from multi-scale and ‘pyramidal hierarchy’ neural networks to improve detection accuracy for objects with large size variations.[45,46] Subsequent work has improved object detection in dense samples using weighted loss functions.[27] By incorporating these advances into our current model, we expect to further improve NuSeT in multi-scale nuclei detection.

Our approach has cross-platform support and comparatively low hardware requirements ([S5 Table](#)). With a medium-level Nvidia GPU (Quadro P4000), training an accurate model only takes five hours, and the inference proceeds at 1.98 seconds/Mega pixel. From a user standpoint, the NuSeT GUI enables researchers to easily segment their images without needing to understand all the details of machine-learning, which connects state-of-the-art computer vision algorithms to a suite of cell biology problems. While in the present work we provide an effective and efficient pipeline for cell nuclei segmentation, this approach should be easily adaptable to a wide variety of image segmentation tasks involving densely packed and overlapping objects, such as jumbled piles of boxes or people in crowds.

Methods

Kaggle dataset preprocessing

The Kaggle dataset was downloaded from the Broad Bioimage Benchmark Collection (Accession number BBBC038v1).[33] This dataset was sampled from a wide range of organisms include human, mice and flies, and the nuclei were recorded under different imaging conditions. Stage-1 training and test datasets were used for training and validation process. All the images were manually censored and training data with low segmentation accuracies were discarded. Only fluorescent images were used for training and validation process. We converted the run-length encoded labels to binary masks for both training and validation labels in MATLAB. The final Kaggle dataset used for our model contains 543 images for training and 53 images for validation. Segmentation errors, including mask misalignment and touching cells, were manually corrected image-by-image for training and validating data.

Mammary acini, MCF10A monolayer growth, and mitosis data collection

The MCF10A data and fluorescent mitosis data were collected on an Olympus FV10i confocal microscope with a 60X objective on MCF10A human breast epithelial cell line. The cell nuclei were stained with 1uM Sir-DNA for 1 hour before imaging. The test set consists of 25 experiments with the corresponding ground-truth binary labels. MCF10AT acini were grown and the acini disorganization assays were performed as described in Shi *et al.*[4] The fluorescent mitosis data were collected on an Olympus FV10i confocal microscope with a 60X objective on MCF10A human breast epithelial cell line stably expressing H2B-eGFP with 10 minutes time interval over 3.5 days.[4] The final fluorescent mitosis training dataset used for our model contains 518 images for training and 57 images for validation.

Liver tissue slide collection

Biopsied liver tissue slides were stained with hematoxylin and eosin and scanned with Philips IntelliSite Pathology Solutions or Aperio AT2 scanners. To accelerate the training process,

each liver slide was down-sized by 8 folds and partitioned into 20–30 tiles of dimensions 256 pixels by 256 pixels. The fat globules were manually annotated by a pathologist for both training and validation datasets. The final training dataset used for our model contains 247 images for training and 10 images for validation.

Histopathology mitosis dataset preprocessing

The Mitosis dataset was downloaded from ICPR 2012[43] and ICPR 2014[44] mitosis detection contests. Breast cancer biopsy slides ranging from low-grade atypia to high-grade atypia were stained with hematoxylin and eosin and scanned by two scanners: Aperio Scanscope XT and Hamamatsu Nanozoomer 2.0-HT. Mitosis events were annotated by at least two individual pathologists. Training datasets acquired at 40X magnifications from ICPR 2012 and 2014 were used for training the model. All the images were manually censored and training data without any mitosis events were excluded. We also converted the coordinates of mitosis locations into binary masks for both training and validation labels using MATLAB scripts. The final training dataset used for our model contains 621 images for training and 69 images for validation. To accelerate the training, we down-sampled the original images by a factor of 2. The trained model was further tested with ICPR 2012 test dataset.

Data augmentation

To accelerate the training process, only simple data augmentation techniques were applied to the training images. We adopted mirror flip and small rotation (10 degrees, counterclockwise) for training data to alleviate the overfitting problem.

Synthetic data generation

Synthetic cell nuclei images were generated by utilizing nuclei-like blobs (adapted from <https://stackoverflow.com/questions/3587704/good-way-to-procedurally-generate-a-blob-graphic-in-2d>), as well as random shape polygons/lines. Signal (brightness) variations were added to both blobs and polygons/lines. The sizes of nuclei like blobs, polygons and lines were varied image-by-image to simulate different imaging conditions. The synthetic images were generated with various image sizes, with width and height ranging from 256 pixels to 640 pixels. Gaussian noise and Gaussian blur were added to these images. We applied overlapping of blobs to strengthen the model capability in separating touching nuclei. The binary masks of the synthetic images were generated separately. To correctly separate all overlapping blobs in the corresponding segmentation masks, the positions of blobs were used as markers to apply watershed transform[36] on overlapping blobs.

Training and inference details

To construct the training data, we incorporated 543 training images from the Kaggle dataset and 25 training images from MCF10A dataset as the base-training dataset. After data augmentation, the training set contained 568 (original) + 568 (flip) + 568 (rotate) = 1704 images. Then we mixed the real images with synthetic images at 1:1 ratio to generate the final training dataset. The training images were normalized by subtracting the foreground mean value and dividing by the foreground standard deviation. Since U-Net contains 4 down-sampling and up-sampling layers, to make the tensors at each layer compatible, training images were further cropped so that widths and heights of the images were adjusted to the nearest multiple of 16. To train RPN, the ground truth coordinates for bounding boxes were calculated based on the binary nuclei masks. The coordinates of the bounding box, (x_min, y_min), (x_max, y_max)

were denoted as the most upper-left and lower-right pixels of the corresponding nuclei. Weight matrices were calculated per mask with $w_0 = 10$ and $\sigma^2 = 5$ pixels. To avoid out-of-memory, one image was fed into the network at a time. During the training, the sequence order of the training data was reshuffled before each epoch to prevent overfitting. The learning rate was set to 5e-5, and Rmsprop[47] was utilized as the training optimizer, and the best performance model was chosen within the first 30 epochs. The training loss was the sum of segmentation loss and detection loss. Segmentation loss was the sum of binary cross-entropy loss[10] and Dice loss, and the detection loss was the class loss and regression loss as described in previous work.[18]

Two validation datasets were used to benchmark the model performance. The Kaggle validation dataset[33] contains 50 images that have various types of nuclei under different imaging conditions. The MCF10A dataset contains 25 images that have homogenous nuclei imaged under the same setting manner. This study was performed on Nvidia Quadro P4000. Additional segmentation performance is shown in S6 Fig.

Model evaluation

Eight models were chosen to compare their performance on both Kaggle and MCF10A validation dataset, including Otsu's Method[8], Deep Cell 1.0[14], U-Net[15], Mask R-CNN[19], NuSeT with whole-image normalization and without synthetic data, NuSeT with whole-image normalization, NuSeT with foreground normalization, and NuSeT with foreground normalization and RPN-aided watershed. The entire training dataset (with data augmentation and synthetic images) was applied to train all NuSeT models. To test Deep Cell 1.0's performance on the Kaggle and MCF10A dataset, we selected the HeLa fluorescent nuclei model from the initial set of models from (<http://www.deepcell.org/predict>, accessed on Feb 25th, 2019). Since no pre-trained two-dimensional fluorescent nuclei segmentation model was found from U-Net[15,34], we trained U-Net on our training dataset (without synthetic data) as our closest estimate for performance. The original Mask R-CNN model was trained for real-life segmentations. Therefore we trained Mask R-CNN on our training dataset (without synthetic data) starting from FPN-101 backbone.[48] We did not apply the aforementioned modified RPN to Mask R-CNN, since Mask R-CNN performs the segmentation strictly after the RPN detection, effectively blocking information transfer between the detection and the segmentation modules. We removed cells smaller than 1/5 of the average cell area in the image for prediction masks from all models prior to benchmarking.

To evaluate model performance, we adopted the following performance metrics: percentage of touching cell separated, correct detections, incorrect detections, split errors, merge errors, catastrophe errors, false negative detection rate (F.N. rate), false positive detection rate (F.P. rate), mean I.U., RMSE, F1 and pixel accuracy. The first eight metrics were evaluated on the nuclei level, and the last four metrics indicate the performance on the pixel-level. The calculation of correct and incorrect detections, as well as split, merge and catastrophe errors have been described in previous works.[29,30] Briefly, correct detections denote the number of predicted cells that can link with ground truth cells, and incorrect detections refer to the number of unlinked cells from the prediction. Split, merge and catastrophe errors are subsets of incorrect detections, where split and merge errors describe the splitting and merging of ground truth cells into prediction cells, and catastrophe errors refer to the uneven matching of ground truth and prediction cells.[29, 30] The percentage of touching nuclei separated is calculated as:

$$\%_{nuclei\ separated} = \frac{N_{nuclei\ separated}}{N_{total\ overlapping\ nuclei}}$$

$N_{nuclei\ separated}$ denotes the number of touching nuclei that have been successfully separated by the model, $N_{total\ overlapping\ nuclei}$ denotes the total number of touching nuclei in the entire dataset.

F.N. rate is the proportion of the nuclei that the model fails to detect in the entire dataset. The detection failure is defined as: given a nucleus' ground-truth binary mask, find the corresponding model-predicted mask that has the largest overlap ratio, which is measured by:

$$\text{overlap ratio} = \frac{A_{GT} \cap A_{pred}}{A_{GT}}$$

Where A_{GT} is the area of ground truth nucleus, A_{pred} is the area of model-predicted nucleus. If the overlap ratio is smaller than 0.7, it is suggested that the model fails to detect the nucleus. Hence the F.N. rate is denoted as:

$$\text{F.N. rate} = \frac{N_{missing\ nuclei}}{N_{total\ nuclei}}$$

$N_{missing\ nuclei}$ denotes the number of nuclei that the model fails to detect. $N_{total\ nuclei}$ denotes the total number of nuclei labelled by ground-truth in the dataset.

Likewise, F.P. rate is the proportion of the nuclei that the model mis-detects in the entire dataset. The mis-detection is defined as: given a nucleus' model-predicted mask, find the corresponding ground-truth mask that has the largest overlap, if the overlap ratio of the model predicted mask and the ground-truth mask is smaller than 0.7, it is suggested that the model detects an 'nucleus' that does not exist in the ground-truth. Hence the F.P. rate is denoted as:

$$\text{F.P. rate} = \frac{N_{mis-detections}}{N_{total\ nuclei}}$$

$N_{mis-detections}$ denotes the number of model-predicted nuclei that found no match in the ground-truth labels.

Pixel-level metrics mean IU, F1, RMSE and pixel accuracy were calculated as:

$$\text{mean IU} = \frac{1}{N_{cls}} \sum_n \frac{TP_n}{TP_n + FN_n + FP_n}$$

$$F1 = \frac{1}{N_{cls}} \sum_n \frac{2TP_n}{2TP_n + FN_n + FP_n}$$

$$RMSE = \sqrt{\frac{1}{N_{pix}} \sum_{i \in I} (y_{i,pred} - y_i)^2}$$

$$\text{pixel accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

Where TP , TN , FP , FN denotes the pixel-level counts of true positive, true negative, false positive and false negative for single image. N_{cls} denotes the number of classes a pixel can be predicted to, in our case $N_{cls} = 2$ (foreground and background), and TP_n denotes the true positive counts of class n. N_{pix} is the number of pixels in the image, and $y_{i,pred}$ is the binary value of pixel i in the model-predicted mask, y_i is the binary value of pixel i in the ground-truth mask. The pixel-level metrics over the entire dataset were then calculated as the average metrics of all the images in the dataset. Precision and recall were calculated as $TP/(TP+FP)$ and $TP/(TP+FN)$.

Supporting information

S1 Table. Internal performance comparison across different datasets. Step-by-step addition of synthetic data, foreground normalization, and RPN-aided watershed result in better performance at object-level. Notice that the pixel-level accuracies (mean IU, RMSE, F1, pixel accuracy) are similar, despite marked differences in object-level metrics.
(DOCX)

S2 Table. External performance comparison of published models across different datasets.
(DOCX)

S3 Table. Effects of whole-image normalization model accuracy on the performance of the foreground normalization model. Whole-image normalization models were trained to different mean IoU levels and connected to the same foreground model to benchmark the final model performance. Metrics were evaluated from three individual experiments.
(DOCX)

S4 Table. Effects of adding foreground normalization on different models. Comparison of segmentation performance for NuSeT, UNet, Mask R-CNN trained with per-image normalization and foreground normalization. Foreground normalization consistently improves the segmentation performance on most object-level metrics for both NuSeT and UNet.
(DOCX)

S5 Table. Memory footprint, training and inference speed comparison for different models.
(DOCX)

S1 Fig. Common problems encountered in nuclei segmentation. Some common factors that affect the quality of nuclei segmentation, are, touching cells (A), signal variation (B), sample preparation artifacts and contaminants (C), and low signal to noise ratio (D). Colored outlines represent the goals (ground truth) for segmentation tasks.
(TIF)

S2 Fig. Adjusting bounding box dimensions based on nuclear size. Historically RPN has used a set of rigid base sizes for all bounding boxes, which resulted in high detection error rate in the Kaggle dataset. We improved the RPN so that it applies different bounding box base sizes for different images. The base size is determined by the median of all nuclei sizes within the image. Nuclei sizes are defined by the maximum value between nuclei widths and heights.
(TIF)

S3 Fig. Foreground normalization is more robust than whole-image normalization in handling images with sample preparation artifacts. Normalizing samples with or without sample artifacts using different normalization methods show that images have more consistent nuclei signals after foreground normalization (highlighted by arrows).
(TIF)

S4 Fig. Additional segmentation performance comparisons across algorithms, including traditional thresholding approach (Otsu's method) and Deep Cell 1.0.
(TIF)

S5 Fig. Additional mammary acini segmentation and tracking results. (A) Three-dimensional acini tracking with different deep-learning models. (B) Additional time-lapse tracking of selected nuclei. (C) Comparison of nuclei area distribution for Otsu's method (median area:

$2816.6 \pm 2845.0 \mu\text{m}^2$) and NuSeT (median area: $138.7 \pm 87.2 \mu\text{m}^2$).
(TIF)

S6 Fig. Additional examples showing NuSeT's performance when handling images with signal variations, shape variations, touching nuclei and sample preparation artifacts.
(TIF)

S7 Fig. Additional fluorescent mitotic events detection and segmentation results.
(TIF)

S1 Text. Supplementary notes about the NuSeT user interface (UI).
(DOCX)

Author Contributions

Conceptualization: Linfeng Yang, Rajarshi P. Ghosh, J. Matthew Franklin, Jan T. Liphardt.

Data curation: Linfeng Yang, J. Matthew Franklin, Simon Chen, Raja R. Narayan, Marc L. Melcher.

Formal analysis: Linfeng Yang.

Investigation: Linfeng Yang, Rajarshi P. Ghosh, J. Matthew Franklin.

Methodology: Linfeng Yang.

Project administration: Linfeng Yang, Rajarshi P. Ghosh.

Resources: Linfeng Yang, Simon Chen, Marc L. Melcher.

Software: Linfeng Yang, Chenyu You.

Supervision: Rajarshi P. Ghosh.

Writing – original draft: Linfeng Yang, Rajarshi P. Ghosh, Jan T. Liphardt.

Writing – review & editing: Linfeng Yang, Rajarshi P. Ghosh, J. Matthew Franklin, Chenyu You, Jan T. Liphardt.

References

1. McBeath R, Pirone DM, Nelson CM, Bhadriraju K, Chen CS. Cell shape, cytoskeletal tension, and RhoA regulate stem cell lineage commitment. *Dev Cell*. 2004.
2. Golding I, Paulsson J, Zawilski SM, Cox EC. Real-time kinetics of gene activity in individual bacteria. *Cell*. 2005. <https://doi.org/10.1016/j.cell.2005.09.031> PMID: 16360033
3. Rosenfeld N, Young JW, Alon U, Swain PS, Elowitz MB. Gene regulation at the single-cell level. *Science (80-)*. 2005. <https://doi.org/10.1126/science.1106914> PMID: 15790856
4. Shi Q, Ghosh RP, Engelke H, Rycroft CH, Cassereau L, Sethian JA, et al. Rapid disorganization of mechanically interacting systems of mammary acini. *Proc Natl Acad Sci*. 2013. <https://doi.org/10.1073/pnas.1311312110> PMID: 24379367
5. Franklin JM, Ghosh RP, Shi Q, Liphardt JT. Spatial resets modulate YAP-dependent transcription. *bioRxiv*. 2019. <https://doi.org/10.1101/539049>
6. Ehman EC, Johnson GB, Villanueva-meyer JE, Cha S, Leynes AP, Eric P, et al. Robust Nucleus/Cell Detection and Segmentation in Digital Pathology and Microscopy Images: A Comprehensive Review. *2017; 46: 1247–1262*. <https://doi.org/10.1002/jmri.25711.PET/MRI>
7. Rogojanu R, Bises G, Smochina C, Manta V. Segmentation of cell nuclei within complex configurations in images with colon sections. *Proceedings—2010 IEEE 6th International Conference on Intelligent Computer Communication and Processing, ICCP10*. 2010. <https://doi.org/10.1109/ICCP.2010.5606433>

8. Otsu N. A Threshold Selection Method from Gray Level Histograms. *IEEE Trans Syst Man Cybern.* 1979. <https://doi.org/10.1109/tsmc.1979.4310068> PMID: 10242122
9. Ursell T, Lee TK, Shiomi D, Shi H, Tropini C, Monds RD, et al. Rapid, precise quantification of bacterial cellular dimensions across a genomic-scale knockout library. *BMC Biol.* 2017. <https://doi.org/10.1186/s12915-017-0348-8> PMID: 28222723
10. Long J, Shelhamer E, Darrell T. Fully Convolutional Networks for Semantic Segmentation. *IEEE Trans Pattern Anal Mach Intell.* 2015. <https://doi.org/10.1109/TPAMI.2016.2572683> PMID: 27244717
11. Zahangir Alom M, Yakopcic C, Taha TM, Asari VK. Nuclei Segmentation with Recurrent Residual Convolutional Neural Networks based U-Net (R2U-Net). Proceedings of the IEEE National Aerospace Electronics Conference, NAECON. 2018. <https://doi.org/10.1109/NAECON.2018.8556686>
12. Johnson JW. Adapting Mask-RCNN for Automatic Nucleus Segmentation. 2018; 1–7. Available: <http://arxiv.org/abs/1805.00500>
13. Al-Kofahi Y, Zaltsman A, Graves R, Marshall W, Rusu M. A deep learning-based algorithm for 2-D cell segmentation in microscopy images. *BMC Bioinformatics.* 2018. <https://doi.org/10.1186/s12859-018-2375-z> PMID: 30285608
14. Van Valen DA, Kudo T, Lane KM, Macklin DN, Quach NT, DeFelice MM, et al. Deep Learning Automates the Quantitative Analysis of Individual Cells in Live-Cell Imaging Experiments. *PLoS Comput Biol.* 2016; 12: 1–24. <https://doi.org/10.1371/journal.pcbi.1005177> PMID: 27814364
15. Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. *Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinformatics).* 2015; 9351: 234–241. https://doi.org/10.1007/978-3-319-24574-4_28
16. Moen E, Bannon D, Kudo T, Graf W, Covert M, Van Valen D. Deep learning for cellular image analysis. *Nature Methods.* 2019. <https://doi.org/10.1038/s41592-019-0403-1> PMID: 31133758
17. Girshick R. Fast R-CNN. Proceedings of the IEEE International Conference on Computer Vision. 2015. <https://doi.org/10.1109/ICCV.2015.169>
18. Ren S, He K, Girshick R, Sun J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *Advances in neural information processing systems.* 2015. <https://doi.org/10.1109/TPAMI.2016.2577031> PMID: 27295650
19. He K, Gkioxari G, Dollar P, Girshick R. Mask RCNN. Proceedings of the IEEE International Conference on Computer Vision. 2017. <https://doi.org/10.1109/ICCV.2017.322>
20. Zhang L, Sonka M, Lu L, Summers RM, Yao J. Combining fully convolutional networks and graph-based approach for automated segmentation of cervical cell nuclei. *Proceedings—International Symposium on Biomedical Imaging.* 2017. <https://doi.org/10.1109/ISBI.2017.7950548>
21. Sadanandan SK, Ranefall P, Le Guyader S, Wählby C. Automated Training of Deep Convolutional Neural Networks for Cell Segmentation. *Sci Rep.* 2017. <https://doi.org/10.1038/s41598-017-07599-6> PMID: 28798336
22. Zhou Z, Rahman Siddiquee MM, Tajbakhsh N, Liang J. Unet++: A nested u-net architecture for medical image segmentation. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics).* 2018. https://doi.org/10.1007/978-3-030-00889-5_1 PMID: 32613207
23. Stringer C, Michaelos M, Pachitariu M. Cellpose: a generalist algorithm for cellular segmentation. *bioRxiv.* 2020. <https://doi.org/10.1101/2020.02.02.931238>
24. Bannon D, Moen E, Borba E, Ho A, Campilisso I, Chang B, et al. DeepCell 2.0: Automated cloud deployment of deep learning models for large-scale cellular image analysis. *bioRxiv.* 2018. <https://doi.org/10.1101/505032>
25. Bai M, Urtasun R. Deep watershed transform for instance segmentation. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017.* 2017. <https://doi.org/10.1109/CVPR.2017.305>
26. Wang W, Taft DA, Chen YJ, Zhang J, Wallace CT, Xu M, et al. Learn to segment single cells with deep distance estimator and deep cell detector. *Comput Biol Med.* 2019. <https://doi.org/10.1016/j.combiomed.2019.04.006> PMID: 31005005
27. Lin TY, Goyal P, Girshick R, He K, Dollar P. Focal Loss for Dense Object Detection. *Proceedings of the IEEE International Conference on Computer Vision.* 2017. <https://doi.org/10.1109/ICCV.2017.324>
28. Fu C-Y, Shvets M, Berg AC. RetinaMask: Learning to predict masks improves state-of-the-art single-shot detection for free. *ArXiv.* 2019;190103353. Available: <http://arxiv.org/abs/1901.03353>
29. Moen E, Borba E, Miller G, Schwartz M, Bannon D, Koe N, et al. Accurate cell tracking and lineage construction in live-cell imaging experiments with deep learning. *bioRxiv.* 2019. <https://doi.org/10.1101/803205>

30. Caicedo JC, Roth J, Goodman A, Becker T, Karhohs KW, Broisin M, et al. Evaluation of Deep Learning Strategies for Nucleus Segmentation in Fluorescence Images. *Cytom Part A*. 2019. <https://doi.org/10.1002/cyto.a.23863> PMID: 31313519
31. Chen X, Gupta A. An Implementation of Faster RCNN with Study for Region Sampling. 2017; 1–3. Available: <http://arxiv.org/abs/1702.02138>
32. Rey J. Faster R-CNN: Down the rabbit hole of modern object detection. Tyrolabs. 2018.
33. Caicedo JC, Goodman A, Karhohs KW, Cimini BA, Ackerman J, Haghghi M, et al. Nucleus segmentation across imaging experiments: the 2018 Data Science Bowl. *Nat Methods*. 2019. <https://doi.org/10.1038/s41592-019-0612-7> PMID: 31636459
34. Falk T, Mai D, Bensch R, Çiçek Ö, Abdulkadir A, Marrakchi Y, et al. U-Net: deep learning for cell counting, detection, and morphometry. *Nat Methods*. 2019; 16: 67–70. <https://doi.org/10.1038/s41592-018-0261-2> PMID: 30559429
35. Malpica N, De Solórzano CO, Vaquero JJ, Santos A, Vallcorba I, García-Sagredo JM, et al. Applying watershed algorithms to the segmentation of clustered nuclei. *Cytometry*. 1997. [https://doi.org/10.1002/\(SICI\)1097-0320\(19970801\)28:4<289::AID-CYTO3>3.0.CO;2-7](https://doi.org/10.1002/(SICI)1097-0320(19970801)28:4<289::AID-CYTO3>3.0.CO;2-7)
36. Beucher S, Meyer F. The morphological approach to segmentation: the watershed transformation. *Mathematical Morphology in Image Processing*. 1993.
37. Soule HD, Maloney TM, Wolman SR, Brenz R, Russo J, Pauley RJ, et al. Isolation and Characterization of a Spontaneously Immortalized Human Breast Epithelial Cell Line, MCF-10. *Cancer Res*. 1990.
38. Krizhevsky A, Sutskever I, Hinton GE. ImageNet Classification with Deep Convolutional Neural Networks. *ImageNet Classification with Deep Convolutional Neural Networks*. 2012.
39. Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition. 2014; 1–14. Available: <http://arxiv.org/abs/1409.1556>
40. Hardt M. Deep learning requires rethinking generalization. *ICLR'17*. 2017. <https://doi.org/10.1109/TKDE.2015.2507132>
41. Abdolhoseini M, Kluge MG, Walker FR, Johnson SJ. Segmentation of Heavily Clustered Nuclei from Histopathological Images. *Sci Rep*. 2019. <https://doi.org/10.1038/s41598-019-38813-2> PMID: 30872619
42. He Y, Gong H, Xiong B, Xu X, Li A, Jiang T, et al. ICut: An integrative cut algorithm enables accurate segmentation of touching cells. *Sci Rep*. 2015. <https://doi.org/10.1038/srep12089> PMID: 26168908
43. Roux L, Racoceanu D, Loméni N, Kulikova M, Irshad H, Klossa J, et al. Mitosis detection in breast cancer histological images An ICPR 2012 contest. *J Pathol Inform*. 2013. <https://doi.org/10.4103/2153-3539.112693> PMID: 23858383
44. Roux L. Mitosis atypia 14 grand challenge. 2014.
45. Lin TY, Dollár P, Girshick R, He K, Hariharan B, Belongie S. Feature pyramid networks for object detection. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*. 2017. <https://doi.org/10.1109/CVPR.2017.106>
46. Liu W, Anguelov D, Erhan D, Szegedy C, Reed S, Fu CY, et al. SSD: Single shot multibox detector. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 2016. https://doi.org/10.1007/978-3-319-46448-0_2
47. Hinton GE, Srivastava N, Swersky K. Lecture 6a overview of mini-batch gradient descent. In: Coursera Lecture slides <https://class.coursera.org/neuralnets-2012-001/lecture>. 2012. <https://doi.org/10.1017/9781139051699.031>
48. Abdulla W. Mask R-CNN for object detection and instance segmentation on Keras and TensorFlow. GitHub Repos. 2017. Available: https://github.com/matterport/Mask_RCNN