Instance-U-Net and Watershed: Improved Segmentations for breast cancer cells

Linfeng Yang, Chenyu You
{linfeng, uniycy}@stanford.edu

Introduction

Segmenting single cells from blood/tissue sample is critical to identify diseased cells to achieve early diagnosis. On the other side, the behaviors of single cells have brought more attention in biological researches. For example, the lineage tracking of single embryo cells is vital to understand the mechanisms in early stage embryo developments; the shape analysis of mammalian cells during Epithelial-Mesenchymal transition is key to dissect the transformation of how normal cells become cancer cells. Till now, identifying cells in images (cell segmentation) has been challenging due to the high density and imbalanced illumination of the sample. Recently, machine learning and deep learning has became the instrumental tool for computer vision tasks [1] [3]. In this work, we proposed a highly efficient and effective neural network model that has accurate predictions on both cell segmentation and cell count.

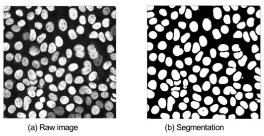


Fig.1 Training data (a) raw image (b) segmentation mask.

Contributions

- Capture each single cell in cell clusters, we integrated Region Proposal Networks (RPN) to detect cells with bounding boxes.
- Predict the cell segmentations by integrating U-Net with pre-computed weight matrices.
- Integrate Watershed algorithm with cell centroids (predicted by RPN) and cell segmentations (predicted by U-Net) as inputs to improve the boundaries predictions of densely touching cells.
- Extensive quantitative evaluations illustrated the superiority of the proposed method compared with state-of-the-art segmentation models

Dataset

ZEISS LSM 700 confocal microscope Dataset

- 40X objective on MCF10A human breast cancer cell line
- The cell nucleus was stained by DAPI and illuminated by laser with wavelength 405nm
- Training dataset includes 484 images with ground-truth binary mask, and 8 images with the corresponding masks for validation
- The size of image is of 448 × 448, and each image contains around 100 cells
- The ground truth labels are computed using thresholding and other segmentation methods, segmentation errors are manually corrected image-by-image. The ground truth coordinates for bounding boxes are computed based on the ground truth labels.

Data augumentation

- Augmented data with cropping, adding noise, rotation, mirror

Network Architecture

Feature extraction network and region proposal network (RPN) [2]

we build our own end-to-end feature extractor. Here is the training procedure:

- The input cell images with the size of 448 × 448 are fed into the convolutional neutral network (CNN) with the kernel size of 3 × 3.
- The feature maps with size of 28 × 28 × 512 are fed into RPN, then the region proposals were generated by placing k anchors on each pixel of the feature map. Bounding box and corresponding class scores are introduced to training.
- The cost function of RPN is denoted as:

$$L(p_i, t_i) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*)$$

- Where i is the index of the anchor and p_i is the predicted probability of anchor i being an object. The ground-truth label p_i^* is 1 if the anchor is positive, and 0 if negative. t_i is a vector that represents coordinates of the predicted bounding box, t_i^* is the ground-truth box associated with a positive anchor. The classification loss L_{cls} is log loss over two classes (object vs. non-object). We use L_1 loss for $L_{reg}(t_i, t_i^*)$.

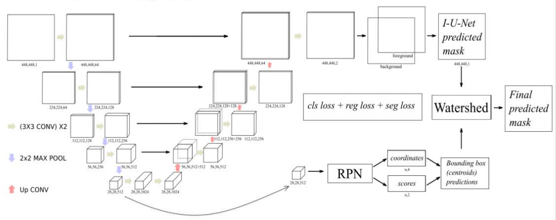


Fig.2 The proposed model

Instance U-Net

- To improve the cell segmentation performance, we bring the objectiveness insights into original U-Net.
- The abovementioned feature extractor network as encoder of U-Net is integrated with deconvolutional layers for extract optimal features.
- Two cost functions are implemented: binary cross-entropy (BCE) loss function for segmentation and dice loss.
- The BCE loss is denoted as:

$$L_{BCE} = \sum w(x) \log(p_{l(x)}(x))$$

- Where $w(x)$ is the weight map. $\log(p_{l(x)}(x))$ is the pixel-wise cross-entropy sum over all pixels.
- The weight map is computed for each ground-truth segmentation mask to urge the network to learn the small separation borders between touching cells.

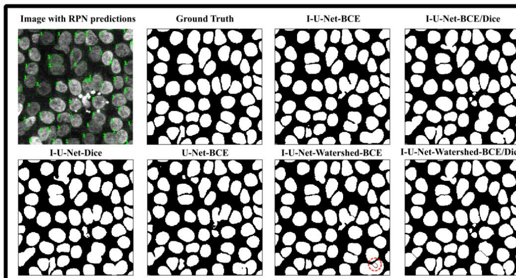
$$w(x) = w_c(x) + w_0 e^{-(\frac{(d_1(x)+d_2(x))^2}{2\sigma^2})}$$

- Where w_c is the weight map to balance the class frequencies, $d_1(x)$ and $d_2(x)$ denote the distance to the border of the nearest cell second nearest cell. In practice we set $w_0 = 10$ and $\sigma = 5$.
- The soft Dice loss is denoted as:

$$L_{Dice} = 1 - 2\frac{\sum_i p_i(x) y_i(x)}{\sum_i p_i^2(x) + \sum_i y_i^2(x) + \epsilon}$$

- Where $p_i(x)$ and $y_i(x)$ are the binary labels from prediction and ground truth labels. ϵ is a small factor to prevent division by zero, in practice we set $\epsilon = 10^{-6}$

Results



Sample results. For the inference cycle, test images(top left without bounding boxes) are fed into different models and generate: 1. the object predictions(top left bounding boxes) with confidence scores and 2. the predicted masks by different models. Among all them, Instance-U-Net with joint BCE/Dice loss obtains the visually best results, it also has the best Watershed separation boundaries without introducing too many errors. Trained with BCE or Dice loss separately only achieved good shape or touching edge segmentations.

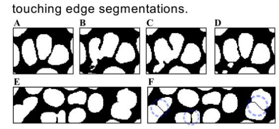


Fig.3 Zoomed-in analysis on the effect of instance training and Watershed algorithm in dense region.

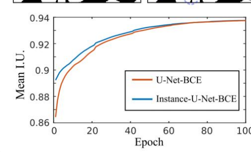


Fig.4 The effect of instance training on Mean I.U. on validation dataset. Compared with U-Net (red-line), Instance-U-Net (Blue line) achieved faster Mean I.U. convergence.

Analysis and Future Directions

- The Instance-U-Net with joint loss achieves a superior segmentation results in microscope images with densely touching cells
- The proposed model achieve fast convergence speed (~7h)
- The implementation of Watershed at the end further improves the performance of separating touching cells by 3 folds.
- In the future, we hope to incorporate the adversarial strategy and experiments on the dataset

References

[1] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask r-cnn. 2017 IEEE International Conference on Computer Vision (ICCV), Oct 2017.
[2] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: towards real-time object detection with region proposal networks. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2017.
[3]Li Liu, Wanli Ouyang, Xiaogang Wang, Paul Fieguth, Jie Chen, Xinwang Liu, and MattiPietikäinen. Deep learning for generic object detection: A survey, 2018.