Nucleus Detection and Segmentation in Digital Microscopy Images by U-Net

Guimin Dong

Abstract—Nucleus detection and segmentation are the core operations in digital pathology and microscopy image analysis for the study of cell morphology. In diagnosis process, pathological examinations make principal and critical contribution to medical protocols and prognostic assessments. The automation of nucleus detection and segmentation with advancement of computer hardware design and software released pathologist's manual labor and provided more robust and efficient analysis methods in clinical application and pathological research. The state-of-art methods in deep learning lead more promising sophistication of methodology in this area. In this paper, we implemented a U-net convolutional neural network architecture with data augmentation and drop out techniques.

Index Terms—Deep Learning, Nucleus Segmentation, CNN, U-net,

1 Introduction

Cancer, heart disease, and diabetes are fatal diseases costing people's wealth and health. These fatal diseases attract medical doctors' and pathologists' attention to dream one day people can fight against such misfortune and improve diagnosis accuracy and medicine treatment. With development of computerized technology, which makes digital pathology and microscopy images available to pathological specialists and genetic scientists, pathologists have found there exists a certain level of relationship between cancer and abnormality in nucleus and cytoplasm[1]. The significance of digital pathology and microscopy images for disease diagnosis and medicine protocol has been recognized not only in cancer research but in clinical implementation, since these digital information expands the possibility of sophistication of quantitative analysis rather than manual detection. Comparing to human assessment of medical evidence, which sometimes cause fatal error in diagnostic process because of limitation of the capability of human's brain information processing, computer science approach provides a more efficient and effective solution to improve accuracy and make reproducible analysis such that patients can gain benefit of information technology revolution[3]. Due to the complexity and mysterious nature of human genes and nucleus, computer-assisted methods embedded with robust algorithm provide rigorous measurements of critical image features, allowing scientists who commit to solve the challenge engage into more diverse and active researches.

Nucleus detection and segmentation are the necessary condition to make computer-assisted method possible to support automation of image analysis. Information of cellular morphology can be quantified in this process, which can provide several dimensions to researchers to have a comprehensive study of pathology. However, because of background clutter with noise, blurred part, and vague boundary between object and background, it is challenging to achieve robust and accurate nucleus segmentation. Besides, heterogeneity and interpersonal difference of the characteristics of nucleus still lead bias and variance nucleus detection and segmentation. Last but not the least, overlapping objects in nucleus detection and segmentation make this task even more complex. In Fig 1, which include four nucleus images, we can have a straightforward sense how difficult this task is, and where such mentioned challenges are. Although there exist several challenges for nucleus detection and segmentation, researchers achieved some robust method to solve some of the problems.

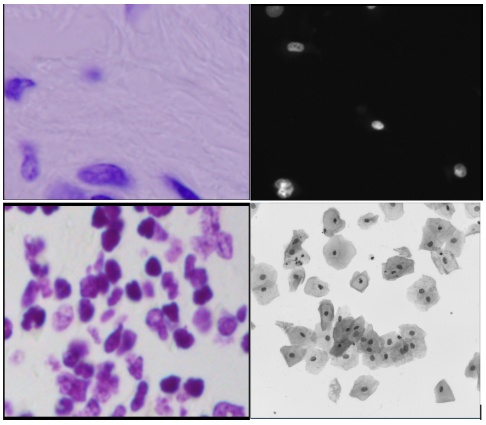

Fig. 1. Nucleus Images from Data Science Bowl 2018

All the way through the development of computer hardware, pathology and computer vision, researchers have established successful benchmark methodology to solve nucleus detection and segmentation problem. There are two main categories of methodology, one is traditional statistical image analysis, and the another is machine learning and deep learning methods in classification.Theodoros, Stephen, and Anil [4] provided a statistical modeling method for robust cell nuclei segmentation by using likelihood maximization and boundary optimization. A statistical level set approach with topology preserving constraint presented by

• *G. Dong is with the Department of Electrical Engineering and Computer Science, Vanderbilt University, Nashville, TN, 37240 .*
E-mail: guimin.dong@vanderbilt.edu .

Shaghayegh, Thomas and Tien [5] can outperform thresholding and watershed segmentation approaches in qualitative and quantitative analysis of fluorescent stained images. As deep learning has been proved having powerful capability in computer vision, more and more success in nucleus detection and segmentation has been achieved. In the review of deep learning in medical image analysis, Dinggang, Guorong, and Heung indicated successes of deep learning in image registration, anatomical structure detection and tissue segmentation [6]. Some state-of-art methods, such as deep convolutional neural network in image analysis also implies future work of research in nucleus detection and segmentation. Jun, Xiaofei, Guanhao, Hannah, and Anant [7] implemented a deep conculutional neural network for segmenting and classifying epithelial and stromal regions in histopathological images with accuracy of 88%.

In the field of biomedical image analysis, which includes nucleus detection and segmentation, advancement of methodology in this field always follows the development of computer vision techniques. And some cutting edge methods in computer vision sometimes have hysteresis impact to the development of biomedical image analysis. Thus, in this paper, we implemented a state-of-art deep neural network architecture from mathematical and computer science's point of view,and investigated the comparative performance analysis of U-net and U-net with data augmentation and drop-out.

2 Problem Statement and Data Set

In computer vision, image segmentation is the process of partitioning a digital image into multiple segments (sets of pixels). The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze [11]. Image segmentation is typically used to locate objects and boundaries in images [8]. The result of image segmentation is a set of segments that collectively cover the entire image, or a set of contours extracted from the image. Each of the pixels in a region are similar with respect to some characteristic or computed property. Adjacent regions are significantly different with respect to the same characteristics. When applied to a stack of images, typical in medical imaging, the resulting contours after image segmentation can be used to create 3D reconstructions with the help of interpolation algorithms [9].

This dataset contains a large number of segmented nuclei images. The images were acquired under a variety of conditions and vary in the cell type, magnification, and imaging modality (brightfield vs. fluorescence). The dataset is designed to challenge an algorithm's ability to generalize across these variations. Each image is represented by an associated ImageId. Files belonging to an image are contained in a folder with this ImageId. Within this folder are two subfolders: Images, shown in Fig2, contains the image file. Masks, shown in Fig 3, contains the segmented masks of each nucleus. This folder is only included in the training set. Each mask contains one nucleus. Masks are not allowed to overlap (no pixel belongs to two masks). Thus, the problem is to train a model to extract each mask object from input nuclei image under supervised learning paradigm.

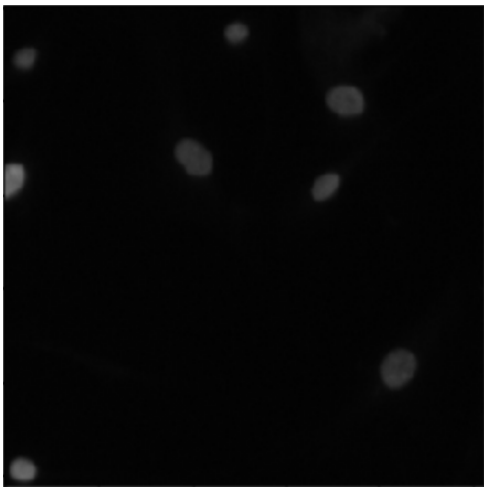

Fig. 2. Nucleus Images

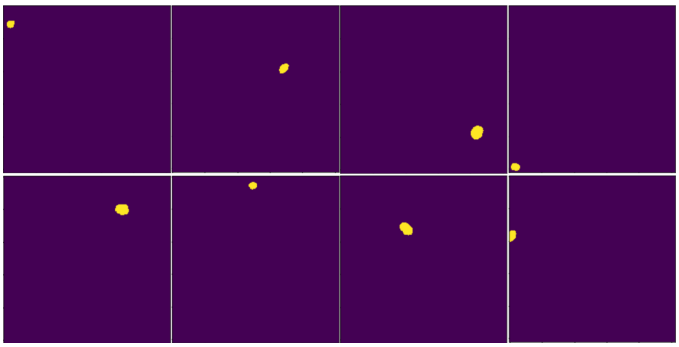

Fig. 3. Masks

3 Methodology

3.1 Exploratory Data Analysis and Statistics

Before we step into constructing deep learning architectures, it is necessary to explore the basic statistical information about the image data set, as these basic statistics, such as distribution, dimension of images, will imply what kind of approach should be applied in data prepossessing and selection of hyper-parameters when we train the deep learning models.

Firstly, we need to calculate the number of training sample and test sample. After we iterate through all images and mark them with training and testing label, we collect 670 training samples and 65 testing samples. Then we use training data set to check basic statistics of images and masks, which is shown in Fig 4 and Fig 5 respectively. We can observer that the number of masks is much larger than the number of images 29461 vs 670. The area dominated by masks only explain a very small part of images by checking mask to image ratio. Then we can have a close look at the distribution of height, width, and area of images, as shown in Fig 6. By observation, distribution of image's dimensions has a certain level of variation, and indicates that different images have different dimension, implying that we need to rescale the dimensions of image into a unified and squared size vertically or horizontally.

	height	width	area	nuclei
count	670.000000	670.000000	6.700000e+02	670.000000
mean	333.991045	378.500000	1.547583e+05	44.971642
std	149.474845	204.838693	1.908250e+05	47.962530
min	256.000000	256.000000	6.553600e+04	2.000000
25%	256.000000	256.000000	6.553600e+04	16.250000
50%	256.000000	320.000000	8.192000e+04	28.000000
75%	360.000000	360.000000	1.296000e+05	55.000000
max	1040.000000	1388.000000	1.443520e+06	376.000000

Fig. 4. Statistics of Images

	img_index	height	width	area	nucleus_area	mask_to_img_ratio
count	29461.000000	29461.000000	29461.000000	2.946100e+04	29461.000000	29461.000000
mean	346.803944	404.408642	506.068090	2.511955e+05	471.803707	0.003165
std	197.395227	187.400013	282.185678	2.516759e+05	583.837040	0.004488
min	0.000000	256.000000	256.000000	6.553600e+04	21.000000	0.000020
25%	183.000000	256.000000	256.000000	6.553600e+04	118.000000	0.000992
50%	361.000000	360.000000	360.000000	1.296000e+05	305.000000	0.001724
75%	523.000000	520.000000	696.000000	3.619200e+05	574.000000	0.003376
max	669.000000	1040.000000	1388.000000	1.443520e+06	11037.000000	0.083557

Fig. 5. Statistics of Masks

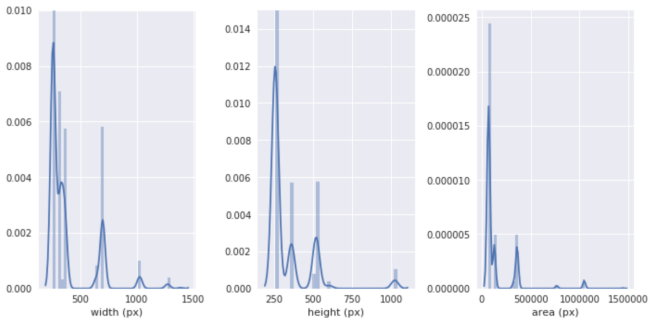

Fig. 6. Distribution of Dimension of Image

3.2 U-Net

A successful training of deep networks requires a large amount of training data. Convolutional neural networks can be specified and trained in classification tasks, where our cnn model can automatically label an object accurately. However, in many visual tasks, especially in biomedical image processing, the desired output should include localization, i.e., a class label is supposed to be assigned to each pixel [11].

Ronneberger, Fischer, and Brox proposed an elegant solution by modifying and extending a convolutional neural network architecture such that it works with very few training images and yields more precise segmentations. Upsampling is the core improvement embedded at U-net architecture by creating a large number of feature channels allow the network to propagate context information to higher resolution layers. As a consequence, the expansive path is more or less symmetric to the contracting path,

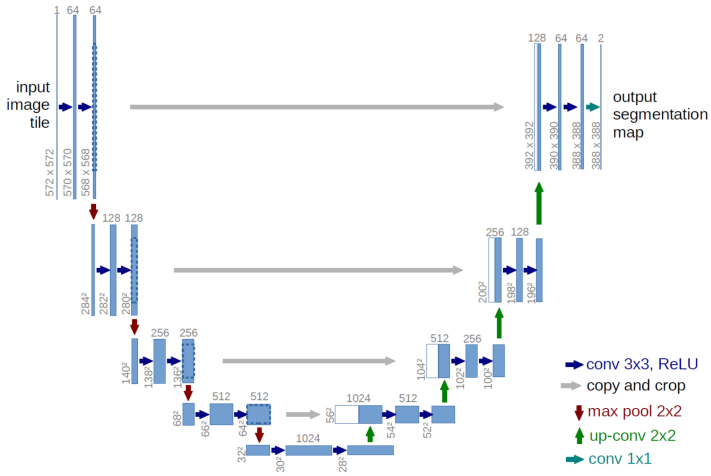

Fig. 7. U-net Architecture

and yields a u-shaped architecture. The network does not have any fully connected layers and only uses the valid part of each convolution layer [12]. This strategy allows the seamless segmentation of arbitrarily large images by an overlap-tile strategy.

The U-net architecture is shown in Fig 7. It consists of a contracting path (left side) and an expansive path (right side). The contracting path follows the typical architecture of a convolutional network. It consists of the repeated application of two 3x3 convolutions (unpadded convolutions), each followed by a rectified linear unit (ReLU) and a 2x2 max pooling operation with stride 2 for downsampling. At each downsampling step we double the number of feature channels. Every step in the expansive path consists of an upsampling of the feature map followed by a 2x2 convolution that halves the number of feature channels, a concatenation with the correspondingly cropped feature map from the contracting path, and two 3x3 convolutions, each followed by a ReLU. The cropping is necessary due to the loss of border pixels in every convolution. At the final layer a 1x1 convolution is used to map each 64- component feature vector to the desired number of classes. In total the network has 23 convolutional layers [13].

3.3 Evaluation Metrics

The results in nucleus segmentation is evaluated on the mean average precision at different intersection over union (IoU) thresholds. The IoU of a proposed set of object pixels and a set of true object pixels is calculated as:

$$IoU(A, B) = \frac{A \cap B}{A \cup B} \tag{1}$$

he metric sweeps over a range of IoU thresholds, at each point calculating an average precision value. The threshold values range from 0.5 to 0.95 with a step size of 0.05: (0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95). In other words, at a threshold of 0.5, a predicted object is considered a ”hit” if its intersection over union with a ground truth object is greater than 0.5.

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	(None, 128, 128, 3)	0	
lambda_1 (Lambda)	(None, 128, 128, 3)	0	input_1[0][0]
conv2d_1 (Conv2D)	(None, 128, 128, 8)	224	lambda_1[0][0]
conv2d_2 (Conv2D)	(None, 128, 128, 8)	584	conv2d_1[0][0]
max_pooling2d_1 (MaxPooling2D)	(None, 64, 64, 8)	0	conv2d_2[0][0]
conv2d_3 (Conv2D)	(None, 64, 64, 16)	1168	max_pooling2d_1[0][0]
conv2d_4 (Conv2D)	(None, 64, 64, 16)	2320	conv2d_3[0][0]
max_pooling2d_2 (MaxPooling2D)	(None, 32, 32, 16)	0	conv2d_4[0][0]
conv2d_5 (Conv2D)	(None, 32, 32, 32)	4640	max_pooling2d_2[0][0]
conv2d_6 (Conv2D)	(None, 32, 32, 32)	9248	conv2d_5[0][0]
max_pooling2d_3 (MaxPooling2D)	(None, 16, 16, 32)	0	conv2d_6[0][0]
conv2d_7 (Conv2D)	(None, 16, 16, 64)	18496	max_pooling2d_3[0][0]
conv2d_8 (Conv2D)	(None, 16, 16, 64)	36928	conv2d_7[0][0]
max_pooling2d_4 (MaxPooling2D)	(None, 8, 8, 64)	0	conv2d_8[0][0]
conv2d_9 (Conv2D)	(None, 8, 8, 128)	73856	max_pooling2d_4[0][0]
conv2d_10 (Conv2D)	(None, 8, 8, 128)	147584	conv2d_9[0][0]
conv2d_transpose_1 (Conv2DTrans	(None, 16, 16, 64)	32832	conv2d_10[0][0]
concatenate_1 (Concatenate)	(None, 16, 16, 128)	0	conv2d_transpose_1[0][0] conv2d_8[0][0]
conv2d_11 (Conv2D)	(None, 16, 16, 64)	73792	concatenate_1[0][0]
conv2d_12 (Conv2D)	(None, 16, 16, 64)	36928	conv2d_11[0][0]
conv2d_transpose_2 (Conv2DTrans	(None, 32, 32, 32)	8224	conv2d_12[0][0]
concatenate_2 (Concatenate)	(None, 32, 32, 64)	0	conv2d_transpose_2[0][0] conv2d_6[0][0]
conv2d_13 (Conv2D)	(None, 32, 32, 32)	18464	concatenate_2[0][0]
conv2d_transpose_3 (Conv2DTrans	(None, 64, 64, 16)	2064	conv2d_14[0][0]
concatenate_3 (Concatenate)	(None, 64, 64, 32)	0	conv2d_transpose_3[0][0] conv2d_4[0][0]
conv2d_15 (Conv2D)	(None, 64, 64, 16)	4624	concatenate_3[0][0]
conv2d_16 (Conv2D)	(None, 64, 64, 16)	2320	conv2d_15[0][0]
conv2d_transpose_4 (Conv2DTrans	(None, 128, 128, 8)	520	conv2d_16[0][0]
concatenate_4 (Concatenate)	(None, 128, 128, 16)	0	conv2d_transpose_4[0][0] conv2d_2[0][0]
conv2d_17 (Conv2D)	(None, 128, 128, 8)	1160	concatenate_4[0][0]
conv2d_18 (Conv2D)	(None, 128, 128, 8)	584	conv2d_17[0][0]
conv2d_19 (Conv2D)	(None, 128, 128, 1)	9	conv2d_18[0][0]

Total params: 485,817
Trainable params: 485,817
Non-trainable params: 0

Fig. 8. Summary of U-net Architecture for Nucleus Segmentation

At each threshold value t, a precision value is calculated based on the number of true positives (TP), false negatives (FN), and false positives (FP) resulting from comparing the predicted object to all ground truth objects:

$$\frac{TP(t)}{TP(t) + FP(t) + FN(t)} \tag{2}$$

A true positive is counted when a single predicted object matches a ground truth object with an IoU above the threshold. A false positive indicates a predicted object had no

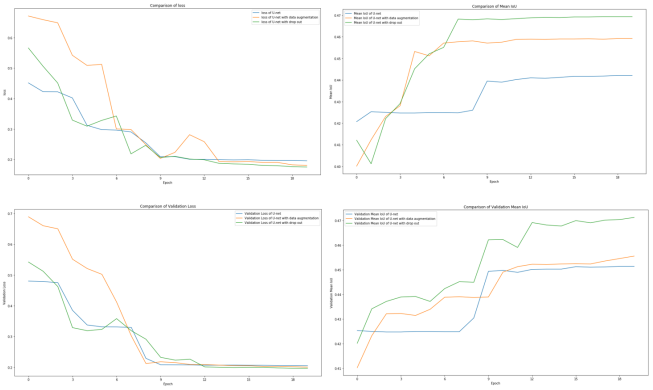

Fig. 9. Ensemble Comparison Plotting

associated ground truth object. A false negative indicates a ground truth object had no associated predicted object. The average precision of a single image is then calculated as the mean of the above precision values at each IoU threshold:

$$\frac{1}{thresholds} \sum_{t} \frac{TP(t)}{TP(t) + FP(t) + FN(t)} \tag{3}$$

Thus the final score is the mean taken over the individual average precisions of each image in the test dataset.

4 Results and Analysis

We implemented U-net in Keras. Here we define each convolutional 3×3 layers is one convolutional component of U-net, thus in U-net, we have 9 convolutional components. The dimension of input is 256×256×3. Summary of this architecture is shown in Fig 8.

After we trained this model to make prediction by using the test data set, we got 0.277 mean over the individual average previsions of each image in the test dataset evaluated by equation (3). This initial result is not every appealing, because the limited number of training images. Then we implemented data augmentation by setting up keras image generator as shear-range=0.5, rotation-range=30, zoom-range=0.3, width-shift-range=0.3, height-shift-range=0.3, fill-mode="reflect". Then the test score is 0.316, which improved the previous result. Then we applied drop out method by inserting a drop-out mask after each convolutional layer. Finally, we got the final score of 0.378. During the training process, we trained on 636 samples, validate on 34 samples. The Fig 9 is ensemble of plot of loss, IoU, which is defined at equation (1), validation loss, validation IoU respectively from data we collected when we training the U-nets. We can observe that the U-net with data augmentation and drop out performs better than the other two models.

Then we investigate why our model did not get a very good results as illustrated in the original paper proposing U-net. Firstly, we sampled the nucleus that are segmented correctly, as shown in Fig 10 . And the nucleus that our model did not segment correctly is sampled shown in Fig 11 . We can observe that our U-net can segment nucleus

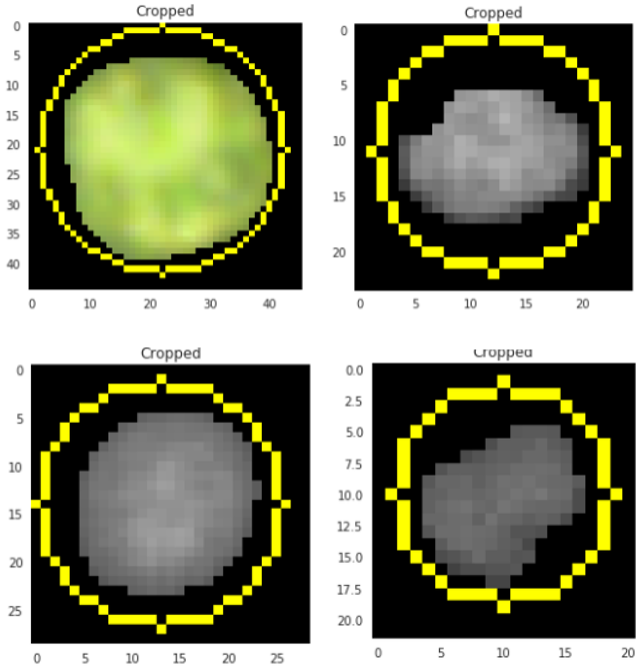

Fig. 10. Correct Nucleus Segmentation

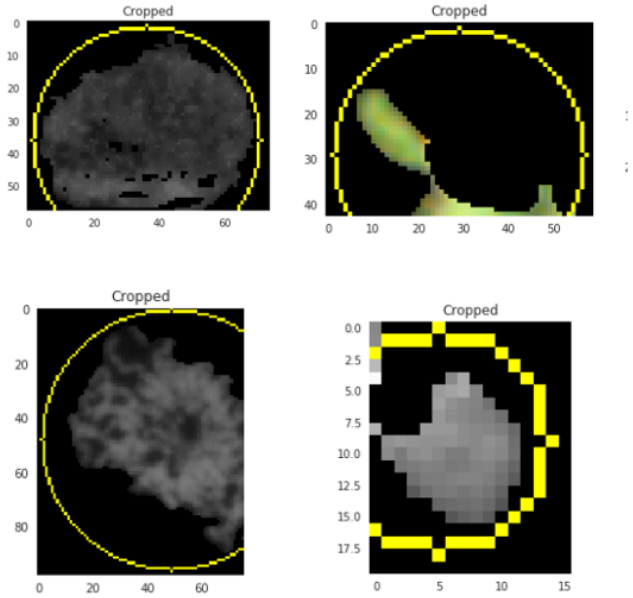

Fig. 11. Incorrect Nucleus Segmentation

with good round-up shape, and cannot perform very well if the nucleus have gaps, holes, or separated parts.

5 Conclusion

In this project, we implemented a U-net convolutional neural network for nucleus segmentation. Data augmentation and drop out technique were implemented to improve the performance of our U-net, and finally, we trained a U-net with data augmentation and drop-out with 0.378 score evaluated by mean of precision value at each IoU threshold, shown in equation (3). In the detailed investigation of analysis of the trained U-net, we can conclude that in our experiment in this data set the trained U-net can segment nucleus with high accuracy which is rounded and full. And the nucleus with holes, gaps, and deficits cannot be segmented correctly by our U-net model. Because of the complexity of U-net architecture, it is difficult to completely explain why the U-net cannot achieve the high accuracy proposed at the original paper. And it is still hard to relate the data distribution with the performance of U-net, as we barely have any assumption of the distribution of input data. Last but not the least, the size of our data is not large such that we can have biased estimation of the parameter with a certain level of variance.

In the future, we invest more time and effort to improve the result. Firstly, we can implement a hyperparameter optimizer to search for an optimized combination of hyperparamters by training our model to generate less biased estimation of parameter with the best performance. Secondly, we need to collect more data to create a better model. Last but not the least, more detailed investigation into the failure of nucleus segmentation by our U-net can help to better understand how U-net works and how to make a more robust convolutional neural network.

References

[1] M. Garca Rojo, V. Punys, J. Slodkowska, T. Schrader, C. Daniel, and B. Blobel, "Digital pathology in europe: coordinating patient care and research efforts", Stud. Health Technol. Inform., vol. 150, pp. 997-1001, 2009.

[2] M. Garca Rojo, "State of the art and trends for digital pathology", Stud. Health Technol. Inform., vol. 179, pp. 15-28, 2012

[3] A. Katouzian, E. D. Angelini, S. G. Carlier, J. S. Suri, N. Navab, and A. F. Laine, "A state-of-the-art review on segmentation algorithms in intravascular ultrasound (IVUS) images", IEEE Trans. Inf. Technol. Biomed., vol. 16, no. 5, pp. 823-834, September 2012.

[4] Theodoros Mouroutis, Stephen J Roberts and Anil A Bharath, "Robust cell nuclei segmentation using statistical modelling", Bioimaging 6 (1998) 7991.UK.

[5] Shaghayegh Taheri; Thomas Fevens; Tien D. Bui, "Robust nuclei segmentation in cyto-histopathological images using statistical level set approach with topology preserving constraint", Proceedings Volume 10133, Medical Imaging 2017: Image Processing; 1013318 (2017); doi: 10.1117/12.2254658 Event: SPIE Medical Imaging, 2017, Orlando, Florida, United States

[6] Dinggang Shen,Guorong Wu,and Heung-Il Suk3, "Deep Learning in Medical Image Analysis", Annu Rev Biomed Eng. 2017 Jun 21; 19: 221248, doi: 10.1146/annurev-bioeng-071516-044442

[7] Jun Xua, Xiaofei Luo , Guanhao Wang, Hannah, Gilmoreb and Anant Madabhushic, "A Deep Convolutional Neural Network for segmenting and classifying epithelial and stromal regions in histopathological images", Neurocomputing Volume 191, 26 May 2016, Pages 214-223

[8] Linda G. Shapiro and George C. Stockman (2001): "Computer Vision", pp 279-325, New Jersey, Prentice-Hall, ISBN 0-13-030796-3

[9] Pham, Dzung L.; Xu, Chenyang; Prince, Jerry L. (2000). "Current Methods in Medical Image Segmentation". Annual Review of Biomedical Engineering. 2: 315337.

[10] Ronneberger, O., Fischer, P., and Brox, T. (2015, October). "U-net: Convolutional networks for biomedical image segmentation". In International Conference on Medical image computing and computer-assisted intervention (pp. 234241). Springer, Cham.

[11] Roohollah Aslanzadeh, Kazem Qazanfari and Mohammad Rahmati. "An Efficient Evolutionary Based Method For Image Segmentation". arXiv preprint arXiv:1709.04393, 2017.

[12] Olaf Ronneberger , Philipp Fischer and and Thomas Brox,. "Dental X-ray Image Segmenation using a U-shaped Deep Convolutional Network". 2015 IEEE International Symposium on Biomedical Imaging.

[13] Nassir NavabJoachim, HorneggerWilliam M., WellsAlejandro Frangi. " Medical Image Computing and Computer-Assisted Intervention". MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part 3, p237-238.