Image Segmentation Methods for Nucleus Detection

Hong Jing Khok
School of Computer Science and Engineering, Nanyang Technological University, Singapore

***Abstract*—Segmenting the nuclei of cells in microscopy images is critical for many biomedical applications; by measuring how cells react to various treatments, researchers can understand the underlying biological processes at work. Therefore, automating nucleus detection could help unlock cures faster. In this study, we reviewed several deep learning architectures for image segmentation. We compared the performance of various U-Net based architectures, DeepLab network, and modified U-Net with various backbone; to perform segmentation of nucleus images from the 2018 Data Science Bowl competition dataset. Our results show that these U-Net based networks outperformed ImageNet pretrained networks. These models were able to generalize to different lighting conditions and nucleus type. As a result, minimal tuning is needed for different nucleus type; this is desirable in clinical settings.**

***Index Terms*—Convolutional Neural Network, Semantic Segmentation, Nucleus Detection**

I. Introduction

Segmentation of the nuclei of cells in microscopy images is often the first step in the quantitative analysis of imaging data for biological and biomedical applications. Many bioimage analysis tools can segment the nuclei in images, but there is a need to select and configure for every experiment. Therefore, it is desirable to build a segmentation method that could be applied to any two-dimensional light microscopy image of stained nuclei across experiments without manually adjusting segmentation parameters.

A popular fully convolutional neural network designed for segmentation of the medical image is the U-Net [1]. The success of U-Net has inspired many adaptations such as the Attention U-Net [2], and UNet++ [3]. Their results have shown consistent improvement in segmentation performance over U-Net across various datasets in their studies. In this study, we will compare how well each of these methods performs on nucleus detection.

As methods like the U-Net perform segmentation, the input feature maps diminish in size while traversing through a series of convolutional and pooling layers. This causes loss of information and reduction of resolution of the images. As a result, the segmented output objects' boundaries are fuzzy, and its predictions are low resolution. DeepLab [4] network addresses these challenges by employing atrous convolutions and atrous spatial pyramid pooling (ASPP). The authors boasted excellent performance when segmenting natural images; we will evaluate this network based on cell nucleus detection.

Most networks that manage to outperform existing methods tend to apply transfer learning [5]. By utilizing features learned from a larger dataset, the network performs better than training only from a small dataset. Because certain low-level features, such as edges, shapes, corners, and intensity, can be shared across tasks. As such, in this study, we will implement a few networks that utilize on pretrained model as the encoder's backbone network. We will use pretrained VGG and ResNet; both are popular networks that have already pretrained on a huge dataset, ImageNet, with many diverse image categories.

In this study, we will review and compare the performance of U-Net, Attention U-Net, and UNet++, and DeepLab to perform segmentation of the nucleus with the 2018 Data Science Bowl dataset. We will also compare this performance against several encoder-decoder networks with various backbones networks (VGG11, VGG16, ResNet34, and ResNet101) pretrained on the ImageNet dataset. Our goal is to determine which network can generalize to well on nucleus detection and compare the trade-off between performance and computational cost.

II. Methods

A. Data

To review various U-Net architectures and our proposed method, we used a popular open dataset provided for the 2018 Data Science Bowl competition. The objective of the competition is to create an algorithm to automate nucleus detection. By automating nucleus detection, it allows researchers to identify each cell in a sample. By measuring how cells react to various treatments, the researcher can understand the underlying biological processes at work. This enables more efficient drug testing, shortening the time it takes for each new drug to come to market—from rare disorders to the common cold.

The dataset contains 2D images and associated segmented nuclei masks. These images included various cell types and were acquired under a variety of magnification and imaging modalities. Each image is paired with a set of labeled masks, where each mask denotes the segmentation of one nucleus. There are no overlap masks; thus, each pixel can only belong to one nucleus.

In this study, we use this diverse set of light microscopy images, as deep learning has shown great potential to solve difficult problems in image analysis and image segmentation. This allows us to build an algorithm that is scalable to detect different cell types and robust to different imaging modalities.

B. U-Net

Ronneberger et al. [1] introduced the U-Net architecture. It consists of a contraction path (encoder) and an expansion path (decoder). The contraction path has a typical CNN architecture where it consecutively stacks two 3x3 convolutions followed by a 2x2 max pooling. This process downsamples the input at each step and doubles the number of feature maps. The expansion path also utilizes a series of convolutional layers to output segmentation masks. 2x2 up-convolution performs upsampling from previous layers. The skip connections transfer the corresponding feature map from the contraction path and concatenate them to up-sampled decoder feature maps. This provides localization information from contraction path to expansion path, due to the loss of border pixels in every convolution.

C. Attention U-Net

Oktay et al. [2] borrowed the idea of an end-to-end-trainable attention module [6], which is commonly used in natural image analysis. These attention maps can amplify the relevant regions, demonstrating superior generalization over several benchmark datasets, resulting in a more accurate and robust image classification performance.

To improve segmentation performance, Khened et al. [7] and Roth et al. [8] relied on additional preceding object localization models to separate localization and subsequent segmentation steps. This can be achieved by integrating attention gates on top of U-Net architecture without training additional models. As a result, attention gates incorporated into U-Net can improve model sensitivity and accuracy to foreground pixels without requiring significant computation overhead. Attention gates can progressively suppress feature responses in irrelevant background regions. Thus their approach eliminates the necessity of applying an external object localization model.

The attention gates are implemented before concatenation operation to merge only relevant activations. Gradients originating from background regions are down-weighted during the backward pass. This allows the model's parameters in prior layers to be updated based on spatial regions that are relevant to a given task.

To further improve the attention mechanism, Oktay et al. proposed a grid-attention mechanism. The gating signal is not a single global vector for all image pixels by implementing grid-based gating, but a grid signal conditioned to image spatial information. The gating signal for each skip connection aggregates image features from multiple imaging scales. Using grid-based gating allows attention coefficients to be more specific to local regions as it increases the grid-resolution of the query signal. This achieved better performance compared to gating based on a global feature vector.

D. UNet++

Zhou et al. [3] aimed to improve segmentation accuracy with UNet++ by incorporating a series of nested, dense skip pathways between the encoder and decoder, and included a deep supervision method.

The redesigned skip pathways have been added to bridge the semantic gap between the encoder and decoder subpaths. The purpose of these convolution layers is to reduce the semantic gap between the feature maps of the encoder and decoder subnetworks, allowing the network to converge. U-Net's skip connections directly connect the feature maps between encoder and decoder, which results in fusing semantically dissimilar feature maps. However, with UNet++, the output from the previous convolution layer of the same dense block is fused with the corresponding up-sampled output of the lower dense block. This brings the semantic level of the encoded feature closer to that of the feature maps waiting in the decoder; thus, optimization is easier when semantically similar feature maps are received.

Dense skip connections have implemented skip pathways between the encoder and decoder. These Dense blocks are inspired by DenseNet [9] with the purpose to improve segmentation accuracy and improve gradient flow. Dense skip connections ensure that all prior feature maps are accumulated and arrive at the current node because of the dense convolution block along each skip pathway. This generates full resolution feature maps at multiple semantic levels.

Deep supervision is added so that the model can be pruned to adjust the model complexity, to balance between speed and performance. For accuracy, the output from all segmentation branches is averaged. For speed, the final segmentation map is selected from one of the segmentation branches.

E. DeepLab

There are three versions of DeepLab. DeepLabV1 introduces the use of atrous convolution and conditional random field (CRF) to control the resolution at which image features are computed. DeepLabV2 uses atrous spatial pyramid pooling (ASPP), which considers objects at different scales and improves the segmentation performance. DeepLabV3 enhances segmentation performance further with a redesigned ASPP module by including batch normalization and image-level features. DeepLabV3 also reduced the model complexity by removing the CRF used in DeepLabV1 and DeepLabV2.

DeepLabV3 extracts image features using a backbone network, pretrained from existing networks such as VGG and ResNet. The network controls the size of the feature map with atrous convolution, located in the last few blocks of the backbone. This helps extract dense feature maps from the images, increasing the receptive field exponentially without losing the spatial dimension, thus improving segmentation tasks' performance. This increases the receptive field exponentially without losing the spatial dimension and improving performance on segmentation tasks, outperforming many state-of-the-art approaches in segmenting natural images.

ASPP is used to obtain multi-scale context information, and the prediction results are obtained by performing upsampling operations. Four parallel atrous convolutions with different atrous rates are applied to handle segmenting the object at different scales. Image-level features are also applied to incorporate global context information by applying global average pooling on the backbone's last feature map. After applying all the operations parallelly, each operation's results along the channel are concatenated and 1x1 convolution is applied to get the output.

F. VGG11 and VGG16

The VGG-11 model [10] is an 11-layer network, whereas the VGG-16 model is a 16-layer network. These are fully connected convolution networks, designed for image recognition and classification on the ImageNet database. For VGG16, it consists of 13 convolution layers that use 3x3 convolution kernels along with max-pooling layers for downsampling. The last three layers are linear layers, two with 4096 units each, followed by a dense layer of 1000 units representing the number of categories in the ImageNet database.

Iglovikov et al. [11] introduced a U-Net modification by replacing the encoder with VGG11 pretrained on ImageNet; they named it TernausNet. In their work, they suggested that neural networks initialized with weights from a network pretrained on a large data set like ImageNet shows better performance than those trained from scratch on a small dataset. Their network architecture was the winning solution in the Carvana Image Masking Challenge.

In this study, we do not need the last three layers since we will link it to our own decoder for segmentation purposes. As such, we will only use the first 13 layers in the pretrained VGG16 network in our encoder so that we can leverage the VGG model as an effective feature extractor. Likewise, we will reuse eight convolution layers in the VGG11 network as the encoder's backbone network for feature extraction. Employing pretrained networks can reduce training time that also helps to prevent over-fitting. Effectively, these networks are U-Net with ImageNet pretrained encoders. We will evaluate the performance of using VGG11 and VGG16 for our U-Net encoders.

G. ResNet

Before ResNet was introduced, fully connected networks like AlexNet, and VGGNet, were standard de facto for image classification tasks. It is known that, given enough capacity to a model, it might be possibly sufficient to represent any function. Therefore, the research community has a common trend to build deeper network architecture. However, when we increase the number of layers in these networks, the problem of vanishing and exploding gradients occurs. This caused deeper networks harder to train, as the gradient is backpropagated to earlier layers, repeated multiplication may make the gradient infinitely small. As a result, as the network goes deeper, its performance gets saturated or even degrades rapidly.

In 2015, He et al. [12] introduced ResNet to solve the problem of vanishing/exploding gradients. They introduce the identity shortcut connection that skips one or more layers to fit the previous layer's input to the next layer without modifications. This modification makes ResNet possible to train hundreds or even thousands of layers and still achieve compelling performance. The authors argue that stacking layers shouldn't degrade the network performance because we could stack identity mappings upon the current network, and the resulting architecture would perform the same. This indicates that the deeper model should not produce a training error higher than its shallower counterparts.

In this study, we will utilize the first five layers of ResNet34 and ResNet101 models pretrained with the ImageNet dataset. These layers are the backbone network for the encoder, and we will link it to our own decoder for segmentation purposes. Effectively, these networks are U-Net with ImageNet pretrained encoders, relying on the pretrained ResNet backbone for feature extraction.

H. Evaluation

Intersection over union (IoU). The evaluation strategy was based on identifying object-level errors. This was accomplished by matching target object masks with predicted objects submitted by participants and then computing true positives and false positives. To match target masks and predicted objects, the IoU score was computed for all pairs of objects. This metric ranges from 0 to 1, where 0 signifies no overlap, whereas 1 signifies perfectly overlapping between predicted and ground truth.

$$\text{Intersection over Union} = \frac{|A \cap B|}{|A \cup B|} \tag{1}$$

Dice Coefficient (Dice). A metric measure of overlap between the predicted and the ground truth is often used to quantify image segmentation methods' performance. Dice measure how similar the objects are, where the size of the two segmentations' overlap is divided by the total size of the two objects. This metric ranges between 0 and 1, where a 1 denotes perfect and complete overlap.

$$\text{Dice Coefficient} = \frac{2|A \cap B|}{|A| + |B|} \tag{2}$$

I. Training Protocol

Each model was trained for 100 iterations with a minibatch size of 16. We used Adam optimizer with an initial learning rate value of 0.001 and reduced the learning rate when the loss metric has stopped improving after ten epochs by a factor of 10. L2 penalty was added to reduce overfitting by fixing 0.05 to weight decay. The loss function is a combination of the Binary Cross-Entropy loss and the Dice Coefficient loss.

We split the dataset into three sets: training, validation, and test sets. We train the model with the train set and evaluate the performance during training using the validation set. After training is completed, we evaluate each model with the test set. But due to the stochastic nature of machine learning, the performance can be affected by the randomness in data shuffle, weights initialization, and GPU. To facilitate better comparison, we will train and validate each method with 30 randomly selected seed numbers. The performance of each method will be by taking the average result.

Implementation and experiments for this study are built on the PyTorch framework and are made publicly available at a companion website[1].

[1] github.com/jinglescode/meditorch.

TABLE I
PERFORMANCE OF VARIOUS METHODS

Model	# Params	IoU
U-Net	34.5M	88.8%
Attention U-Net	34.8M	88.7%
UNet++	36.6M	88.5%
DeepLab	59.3M	83.0%
VGG11	22.9M	79.9%
VGG16	29.3M	88.1%
ResNet34	35.1M	75.4%
ResNet101	76.7M	60.9%

III. RESULTS AND DISCUSSION

The segmentation performances of 7 architectures are shown in Table I. All the above-reported results are based on ground truth segmentation, and we average the IoU of 30 runs on the test set. Overall, we can see that the U-Net based architectures outperform DeepLab, VGG, and ResNet based architectures. This is expected as U-Net specifically to perform image segmentation on biomedical images; thus, it achieves good performance and good training time. Between the U-Net based architectures, all three architectures' segmentation performance is marginally close, with U-Net best performing at 88.8%. In many deep learning research work, many researchers increase the model size to meet higher precision needs. Given that UNet++ has approximately 2 million more parameters and trains twice the time needed for the original U-Net, this is demanding in computational resources and memory. It makes one wonder whether adopting this method is worth it, comparing the gains with their costs. U-Net achieves good performance and good training time; it is the smallest model, approximately 6% smaller than UNet++. U-Net also performs training and inference faster, approximately half the time of UNet++.

For the DeepLab architecture, despite using ASPP to obtain multi-scale context information, these parallel atrous convolutions with different atrous rates that segments object at different scales did not benefit nucleus detection. We can see that by observing that DeepLab with pretrained ResNet outperformed the network that trains from scratch. Nevertheless, even with a pretrained ResNet backbone, DeepLab did not outperform any U-Net based networks. DeepLab networks are more suitable for image segmentation on natural images.

Leveraging on pretrained VGG and pretrained ResNet networks as U-Net encoder's backbone network for feature extraction did not outperform U-Net. This could be because the features trained on ImageNet did not help to detect nuclei's edges and borders.

IV. CONCLUSION

Automate the process of identifying nuclei will allow for more efficient drug testing, speeding up research for almost every disease, from lung cancer and heart disease to rare disorders, shortening the duration it takes for each new drug to come to market. Many lives would be transformed if cures for diseases came faster.

We utilized the Data Science Bowl 2018 competition dataset, which contains a large number of segmented nuclei images. We designed multiple methods to generalize across images of various cell types, magnification, and imaging modalities. These algorithms provide high precision extraction of nuclei, which are highly desired for clinicians.

This study reviews and compares the performance of U-Net, Attention U-Net, UNet++, and DeepLab. We also explore the effects of transfer learning by replacing pretrained layers in U-Net's encoder. We used intersection over union to access the predicted mask's accuracy with the ground truth. We show that these networks were robust and performed under various lighting conditions, commonly found in real-world datasets. We observed that U-Net-based architectures outperformed DeepLab, as these networks were designed to perform image segmentation on medical image datasets. We also noticed that using pretrained layers trained on the ImageNet dataset did not improve segmentation performance.

REFERENCES

[1] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[2] Ozan Oktay, Jo Schlemper, Loic Le Folgoc, Matthew Lee, Mattias Heinrich, Kazunari Misawa, Kensaku Mori, Steven McDonagh, Nils Y Hammerla, Bernhard Kainz, et al. Attention u-net: Learning where to look for the pancreas. *arXiv preprint arXiv:1804.03999*, 2018.

[3] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. Unet++: A nested u-net architecture for medical image segmentation. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*, pages 3–11. Springer, 2018.

[4] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017.

[5] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

[6] Saumya Jetley, Nicholas A Lord, Namhoon Lee, and Philip HS Torr. Learn to pay attention. *arXiv preprint arXiv:1804.02391*, 2018.

[7] Mahendra Khened, Varghese Alex Kollerathu, and Ganapathy Krishnamurthi. Fully convolutional multi-scale residual densenets for cardiac segmentation and automated cardiac diagnosis using ensemble of classifiers. *Medical image analysis*, 51:21–45, 2019.

[8] Holger R Roth, Le Lu, Nathan Lay, Adam P Harrison, Amal Farag, Andrew Sohn, and Ronald M Summers. Spatial aggregation of holistically-nested convolutional neural networks for automated pancreas localization and segmentation. *Medical image analysis*, 45:94–107, 2018.

[9] Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

[10] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[11] Vladimir Iglovikov and Alexey Shvets. Ternausnet: U-net with vgg11 encoder pre-trained on imagenet for image segmentation. *arXiv preprint arXiv:1801.05746*, 2018.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.