Segmenting Functional Tissue Units in Human Organs

Yiqing (Melody) Wang
Computational Biology Department
Carnegie Mellon University
yiqingwa@andrew.cmu.edu

Parker Simpson
Computational Biology Department
Carnegie Mellon University
psimpson@andrew.cmu.edu

Kevin Elaba
Computational Biology Department
Carnegie Mellon University
kelaba@andrew.cmu.edu

1 Introduction

Functional tissue units (FTUs) can inform doctors of pathological changes, indicate disease progression, and the density of FTUs can be correlated with age, sex, body mass index (BMI), and other clinically relevant metrics [1]. An FTU is defined as a three-dimensional block of cells centered around a capillary that are within diffusion distance (about $100\mu m$) from each other [2]. The cell type compositions of FTUs are distinct for different tissues [2]; for example, they are the glomeruli in kidneys and crypts in the large intestine [1]. Manual annotation of FTUs is quite labor intensive. The kidney alone has around 600,000 glomeruli [3]. Therefore, automated methods for accurate annotation of FTUs in various tissues can save tremendous amount of manual labor and aid medical diagnosis. Prior research has shown that deep learning methods can effectively segment FTUs in a robust and generalizeable fashion [4]. This project aims to create deep learning models to identify FTUs in five different tissues: prostate, lung, spleen, kidney, and large intestine.

In the following report, we compare and contrast three different deep learning architectures for segmentation of FTUs in histology images: a fully convolutional neural network, U-Net, and reverse attention network (PraNet).

1.1 Dataset

Our dataset is sourced from the Human BioMolecular Atlas Program (HuBMAP). It contains 351 stained histology images from spleen, lung, large intestine, kidney, and prostate organs, along with their ground truth masks segmenting out the FTUs. To evaluate model performance, we split the dataset into 70% training and 30% validation for the fully convolutional neural network, U-Net, and Pranet; 80% training, 10% validation, and 10% test datasets for a pretrained U-Net. We evaluate the methods based on the mean dice coefficient. We also downsample the images to 256 pixels as we found biopsy images of varying size within the dataset (Figure 1).

2 Methods

2.1 Fully Convolutional ResNet

The fully convolution network (FCN) architecture, proposed by Long et al. [5], was a fundamental framework for deep learning based semantic segmentation methods. Unlike traditional convolutional neural networks of the time that used a multilayer perceptron as the last layer for prediction, the FCN architecture uses convolutional layers for all layers in the network, allowing the architecture to both accommodate inputs of arbitrary size and produce outputs that match the input size. After the publication of this method, many traditional CNN architectures were altered to be FCNs.

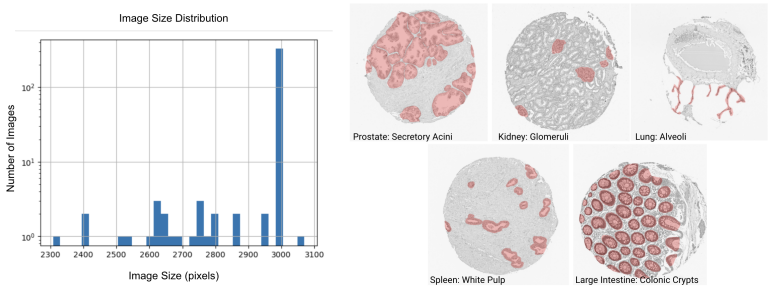


Figure 1: HuBMAP image size distribution (left). Organ biopsy images and FTU segmentation (right).

One such architecture is ResNet, proposed by He et al. [6]. This architecture introduced the idea of residual connections for neural networks. Traditionally the input to an intermediate layer of a neural network was simply the output of the previous layer. Residual connects modify the input of some intermediate layers to be the output of the previous layer summed with the input from an arbitrary layer preceding the previous layer. The use of the residual connection throughout a network proved beneficial to the performance of the model for certain tasks.

The FCN ResNet architecture will be the baseline segmentation model for this project. We will use the implementation of this model included in the PyTorch package. However, this implementation is preset to segment 21 classes, thus we have modified to it for our goal of binary segmentation.

2.2 In-house U-Net

The original U-Net is one of the state-of-the-art methods for segmenting various sets of biomedical images, including cell tracking, light microscopy cell segmentation, and electron microscopy neuronal structure segmentation [7]. It is an encoder-decoder network with the structure shown in figure 2, namely four encoder blocks, one middle layer block, and four decoder blocks. The in-house U-Net references the code from [8], which implements the original U-Net, and makes modifications upon it. Instead of supplying a gray-scale image as the input, and thus only having one channel, we input the RGB 3-channel resized image, hence modifying the input layer to be $256 \times 256 \times 3$. We also modified the output channels to be 1, and the model outputs sigmoid activation of the final layer output. As is the case in the original architecture, we doubled the number of layers moving down the encoder blocks and halve the number of layers moving up the decoder blocks. However, to make the model smaller to match our limited compute resources, we started from 16 filters instead of 64. To preserve the complexity of the model and thus let the model learn complicated structures of the masks, we decided to add one more block, making our in-house U-Net having 5 encoder blocks, one middle layer block, and 5 decoder blocks. We also added 50% dropout in the input layer, the second to the last layer, and the second middle layer.

We trained the in-house U-Net on all tissues as well as each tissue at a time using binary cross entropy (BCE) loss. Due to the small size of the dataset, especially for tissue-specific models, we augmented the data by 10 folds, creating a dataset 11 times the original size. The augmentations are random horizontal or vertical flips, random angle rotations, random shearing in the x or y direction, and combinations of the above.

We trained the tissue-specific models to maximize the soft dice loss. The soft dice loss is calculated as in equation 1, where $Pred$ is the prediction (between 0 and 1), $Mask$ is the ground truth mask, and i, j, k refers to the three dimensions of the predicted and the ground truth masks.

$$SoftDiceLoss = \frac{2 * \sum_{i,j,k} Pred \odot Mask + smoothing}{\sum_{i,j,k} Pred_{i,j,k} + \sum_{i,j,k} Mask_{i,j,k} + smoothing} \tag{1}$$

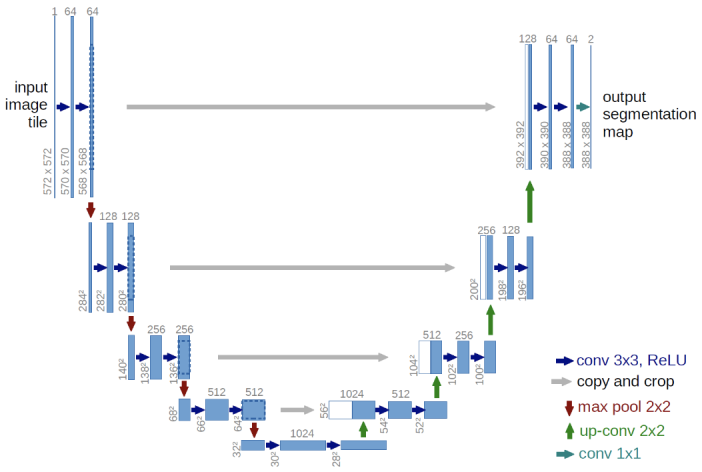


Figure 2: Original U-Net Architecture; from [7]

2.3 Pre-trained U-Net

We validate and compare the performance of our In-house U-Net with a fine-tuned pretrained U-Net, with an architecture shown in Figure 2, except that the final layer has one channel instead of two. We utilize a python package with a U-Net optimized for binary segmentation of biological microscopy images [8] and evaluate fine-tuned model performance across all organs with a combination of binary cross entropy loss and complementary soft dice loss, as well as solely the complementary soft dice loss. The combination is calculated as in equation 2, where SoftDiceLoss is calculated from equation 1. Because we use the complementary of the soft dice loss, we are able to minimize the combined BCEDice loss during training. We also compare model performance with and without data augmentations.

$$BCEDice = 0.5 * BCELoss + 0.5 * (1 - SoftDiceLoss) \tag{2}$$

2.4 Parallel Reverse Attention Network

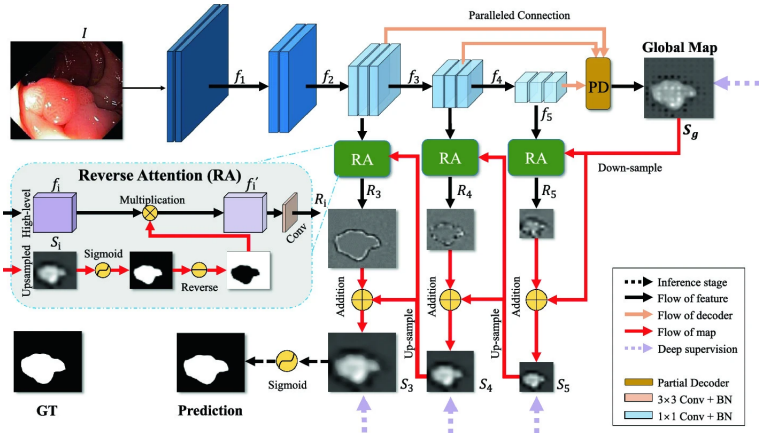


Figure 3: PraNet Architecture [9]

Reverse attention in neural networks is the idea of using part of the network to learn to area of the image corresponging to background for each class. PraNet [9] is a model developed to segment polyps in colonoscopy images that uses reverse attention in an effort to mimic human annotation of colonoscopy images. During human annotation of polyps in colonoscopy images clinicians first, roughly locate the poly, then precisely define a silhouette mask based on local features. PraNet

uses convolution to predict the approximate area and reverse attention to define the boundary of the poly in this area. We decided to apply this model to our problem because it displayed significant improvements in segmentation performance on polyps which, similarly to the FTUs in this project, have a wide range of shapes and sizes.

3 Results

3.1 Fully Convolutional ResNet

The PyTorch implementation of FCN ResNet includes weights from pre-training the model on the Common Objects in Context (COCO) 2017 dataset. We first compared the performance of FCN ResNet50 with pre-trained weights to FCN ResNet50 with randomly initialized weights on our dataset. We trained both models for 20 epochs and observed a maximum validation dice loss of 0.651 from the randomly initialized model and 0.674 for the model initialized with pre-trained weight. It was unexpected that the pre-trained model would perform so similarly to the randomly initialized weights, but given that the pre-taining dataset was so vastly different from our data, it makes sense that it would not drastically improve training results. Although the difference in performance was not large, we decided to continue our experiment using the pre-trained weights for initialization.

The PyTorch implementation of FCN ResNet architecture also has an auxiliary classifier at the last layer. This is a classifier that is used in conjunction with the primary classifier to train the network. Next, we tested if using the auxiliary network to help train the model on our data would be beneficial. To incorporate the auxiliary classifier into training we modified the training loss to be the dice loss of the primary classifier plus 0.4 multiplied by the dice loss of the auxiliary classifier. Notice, this modification moved the optimal training loss from 1.0 to 1.4 (Figure 4b). After training for 20 epochs, we found that including the auxiliary classifier yielded a validation dice score of 0.666 while excluding it yield a score of 0.674. Given the close validation loss results and similar mask predictions (Figure 4), we decided to use only the primary classifier to train FCN ResNet moving forward.

Finally, we compared a pre-trained FCN ResNet50 model we had been using with a larger FCN ResNet101 model. Extrapolating the finding of our experiments with FCN ResNet50, we decided to only test the pre-trained FCN-resNet101 model. Upon training for 20 epochs, the best validation dice loss from pre-trained FCN-ResNet101 was 0.671, just slightly less than the results of the smaller model.

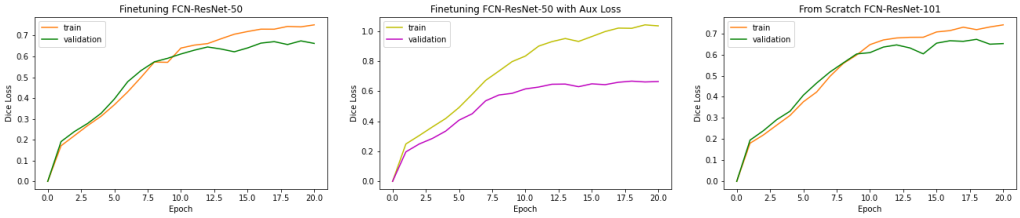
The performance of all the FCN ResNet models was very similar in terms of validation dice loss (1) and segmentation results. We noticed that for most organs, the predicted masks for all FCN models were to capture the general shape of the ground truth, but lacked the fine details within this area. We thought that a deep ResNet model may help solve this problem, but it appears that a completely different architecture is necessary for better results.

Model	Pre-trained	Auxiliary	Dice Loss
FCN-ResNet50	No	No	0.651
FCN-ResNet50	Yes	No	0.674
FCN-ResNet50	Yes	Yes	0.666
FCN-ResNet101	Yes	No	0.671

Table 1: Dice Loss for various FCN ResNet models

3.2 In-house U-Net

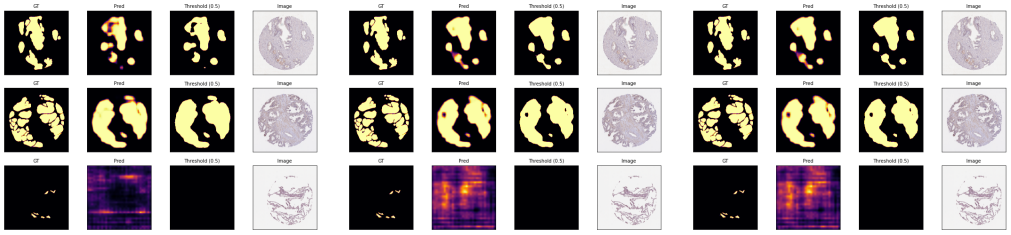
First, we investigate the performance of a tissue-nonspecific model and an example tissue-specific model. Even though the bce loss of the tissue-nonspecific model after 20 epochs (with the same magnitude of data augmentation) is lower than the prostate-specific model (figure 5a), as shown in figure 5b, after training for 20 epochs, the predictions from the tissue-nonspecific model seem to be weak, i.e. having low values, and fail to capture the general structure of the ground truth. In contrast, the prostate-specific model has strong predictions and produces masks similar to the ground truth.



(a) PT-FCN ResNet50 Dice Loss

(b) PT-FCN ResNet50 w/ Aux Dice Loss

(c) PT-FCN ResNet101 Dice Loss

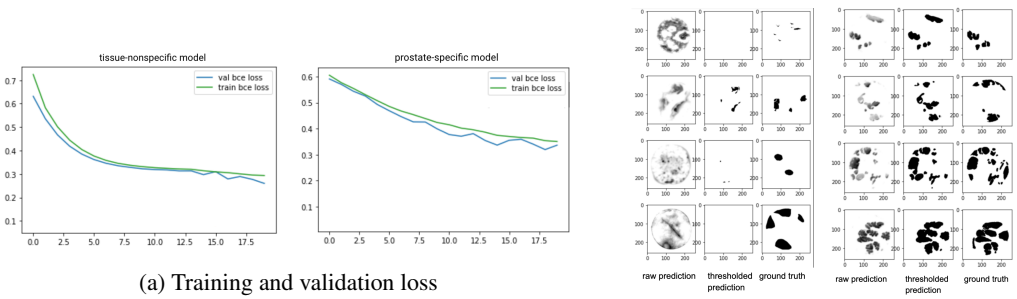


(d) PT-FCN ResNet50 Prediction

(e) PT-FCN ResNet50 w/ Aux Prediction

(f) PT-FCN ResNet101 Prediction

Figure 4: FCN ResNet50 training and predictions with and without the auxiliary classifier



(a) Training and validation loss

(b) Predictions.

Figure 5: Tissue-nonspecific U-Net (left) and prostate-specific U-Net (right). Both models trained for 20 epochs. Predicted segmentation masks sampled from validation set. Thresholding rules values above 0.3 as 1.

Through this qualitative analysis of random validation images, we speculated that the better loss curve may simply be a result of more data and decided to weigh the segmentation results more and train tissue-specific in-house U-Nets for each of the tissues.

As evident in figure 1, different tissues display vastly different functional tissue units, which makes training one model for all ambitious. In addition, as shown in figure 6, some tissues, such as the lungs, are particularly hard to train a model on, having magnitudes of lower dice loss than other models. Overall, training leads to betterment in performance. Also, all tissue-specific models show better validation dice loss than training dice loss likely due to overly harsh dropouts during training.

The low performance of the lung-specific model is also evident in the predictions. Segmentation masks for the lungs display a gross overestimate of the size of the functional tissue units (the alveoli) and incorrect estimates of the location of them. Despite missing the granularity of the ground truth, the masks from the other tissues seem to have captured the location accurately, albeit overestimating the size of it. Particularly of interest is the large intestine segmentation results. In figure 6, the large intestine model seems to behave among the best, similar to the prostate-model. However, the masks tell a different story. The dice loss seems high because the ground truth has a lot of small functional tissue units that are closely packed together, so the lack of granularity in the prediction cannot be

appropriately punished for and thus be avoided. Therefore, future effort could consider using different loss metrics and combinations of them to more accurately reflect model performance. In addition, the model complexity may need to be even higher to capture granular details in the ground truth. As a result, more compute resources need to be dedicated to this cause and the models need to be trained for more epochs with more data, driving up the cost.

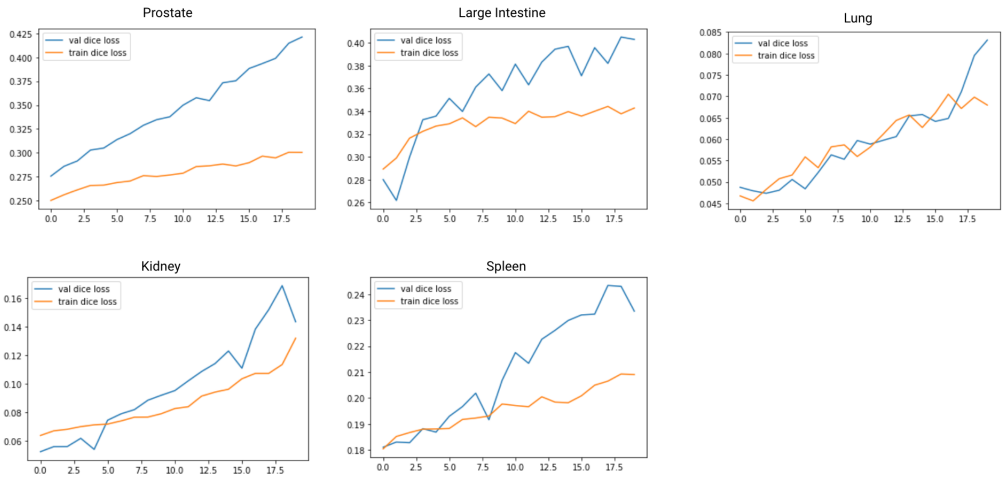


Figure 6: Tissue-specific models trained for 20 epochs, maximizing dice loss. Note that y-scales are different.

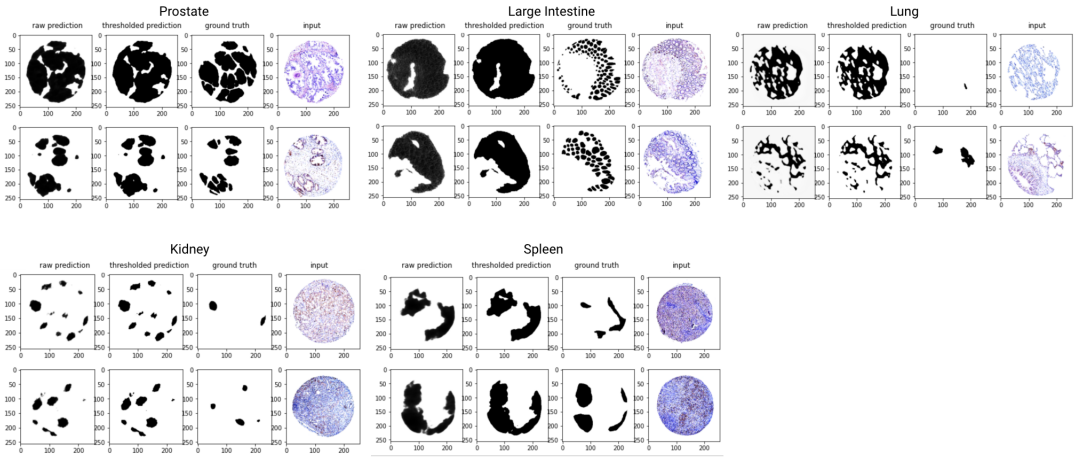


Figure 7: Predicted segmentation masks, thresholded masks (threshold = 0.3, all values above 0.3 is 1), ground truth, and input from tissue-specific models trained for 20 epochs, maximizing dice loss.

3.3 Pre-trained U-Net

We report the results of fine-tuning two pre-trained U-Net models, one with and one without data augmentation, evaluated with binary cross entropy dice loss criterion (BCEDice in equation 2) over 20 epochs. See figure 8. We see that the U-Net with data augmentations converges to a smaller loss, which is expected of larger amounts of training data. This same U-Net also earned a maximum soft dice score of 0.80054 on the validation set. The segmentation also seemed to be granular for certain organs, but we also found that when the ground truth segmentation is sparse, the model seems to ignore those images and not predict anything. As a result of this behavior, the mean dice coefficient on the test set was 0.4243, which was lower than what we expected. On one hand, compared to around 0.8 in the validation set, this indicates overfitting. To increase generalizability, we could

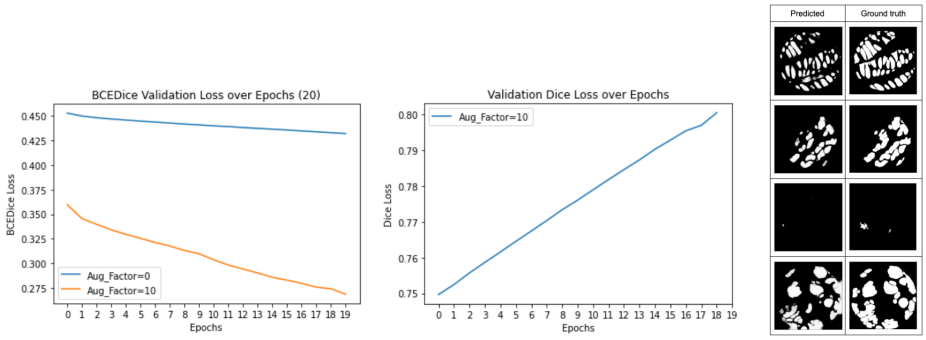


Figure 8: BCEDice Loss and Soft Dice Loss Plots (left, center). U-Net Predictions (right).

consider adding dropout to the layers and increasing augmentation size. On the other hand, we expect the lower performance to be attributed to naturally difficult segmentation from certain tissues, such as the lungs. Therefore, to improve model performance, we need to supply to model with more training data from these difficult tissues and perhaps separate the fine-tuning process for different tissues, such that the more difficult ones can be focused on. Compared to the in-house U-Net, with the same amount of data augmentation, the pre-trained U-Net performs a lot better, starting from a validation dice loss of around 0.75 at the beginning of fine-tuning and reaching the highest validation dice loss of all models after 20 epochs, achieving a soft dice loss of around 0.80. From the predictions in figure 8, it is also evident that the pre-trained and fine-tuned U-Net captures the kind of granularity that the in-house model overlooks.

3.4 PraNet

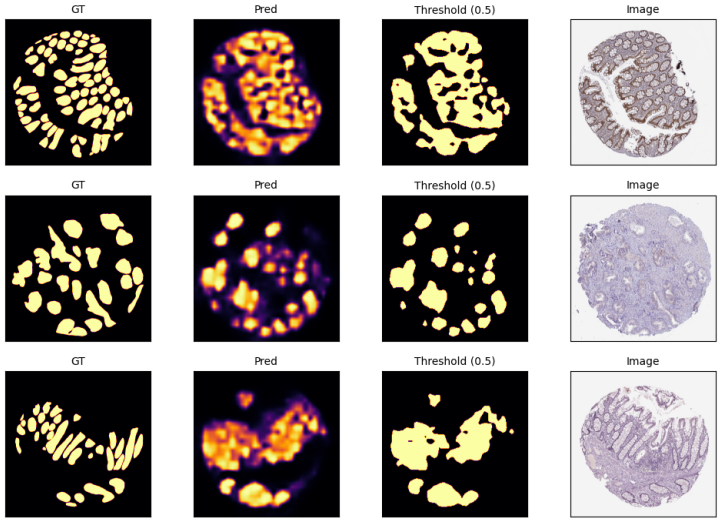


Figure 9: Pre-trained PraNet Prediction after 9 epochs

When implementing the PraNet model, there were many setbacks involving setting up the environment in Google Colab, adapting the code to our project, and troubleshooting random crashes. Due to this, we were not able to perform as extensive experimentation with this model as was done with others. Nevertheless, we trained this model for 9 epochs using a split of the full dataset. PraNet achieved and validation dice loss of 0.579. Comparing the predicted masks in Figure 9 to the FCN models, it appears that, despite training for less than half the time, the predictions from PraNet appear to begin learning some of the more granular details of the mask that the FCN models failed to capture. Unfortunately, these are the only results we have for this model, but this level of performance after 9

epochs of training leads me to believe that PraNet would benefit greatly from longer training and a data augmentation.

4 Conclusions

We have investigated various different attempts at solving the difficult segmentation problem of marking functional tissue units from histology stainings. In hindsight, the dataset is small, which makes training an in-house model challenging. Without pre-training, a fully convolution network like ResNet50 learns better than the in-house U-Net after 20 epochs. However, fine-tuning a pre-trained model, whether that was originally trained on non-biological images or specifically on biological images, leads to better results. This is an interesting observation because it suggests that there are some universal properties of images that are transferable. In addition, it highlights the fact that data is power, as a pre-trained model would have seen a large amount of images already. Another challenge is that FTU segmentation itself is a hard problem. For example, the FTUs of the lungs are the alveoli, which are very thin air sacs. In histology images, the 2D morphology looks like strings if the tissue is cut one way but chambers if cut the other way. Therefore, 3D morphological information is lost in the tissue processing step and adds intrinsic complexity to the problem.

The pre-trained U-Net and PraNet both successfully captured the granular finer details of the ground truth masks. They are the more promising candidates of all models explored in this project. The pre-trained U-Net has a misleadingly high validation dice loss but low test dice loss, so quantitatively, the PraNet is better than the pre-trained U-Net. We recommend moving on with the PraNet and train it for more epochs for better results.

References

[1] Leah L. Godwin et al. "Robust and generalizable segmentation of human functional tissue units". In: *bioRxiv* (2021). DOI: 10.1101/2021.11.09.467810.

[2] Bernard de Bono et al. "Functional tissue units and their primary tissue motifs in multi-scale physiology". In: *Journal of biomedical semantics* 4.1 (2013), pp. 1–13.

[3] Bendtsen TF. Nyengaard JR. "Glomerular number and size in relation to age, kidney weight, and body surface in normal man." In: *The Anatomical record* 232.2 (1992), pp. 194–201.

[4] Leah L. Godwin et al. "Robust and generalizable segmentation of human functional tissue units". In: *bioRxiv* (2021). DOI: 10.1101/2021.11.09.467810. eprint: https://www.biorxiv.org/content/early/2021/11/11/2021.11.09.467810.full.pdf. URL: https://www.biorxiv.org/content/early/2021/11/11/2021.11.09.467810.

[5] Jonathan Long, Evan Shelhamer, and Trevor Darrell. "Fully convolutional networks for semantic segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 3431–3440.

[6] Kaiming He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[7] 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*. Springer. 2015, pp. 234–241.

[8] *bio-image-unet*. URL: https://pypi.org/project/bio-image-unet/.

[9] Deng-Ping Fan et al. "Pranet: Parallel reverse attention network for polyp segmentation". In: *International conference on medical image computing and computer-assisted intervention*. Springer. 2020, pp. 263–273.