NUCLEI DETECTION VIA U-NETS WITH TENSORFLOW

Charlie Carpenter and Piper Williams
BIOS 7718 Final Project
May 13, 2018

Presentation Outline

- ***Introduction***
  - Image data and annotation files
  - Problem at-hand: nuclei detection
- ***Methods***
  - Description of U-Net
  - Preparation of training data
- ***Results***
- ***Conclusions and Limitations***

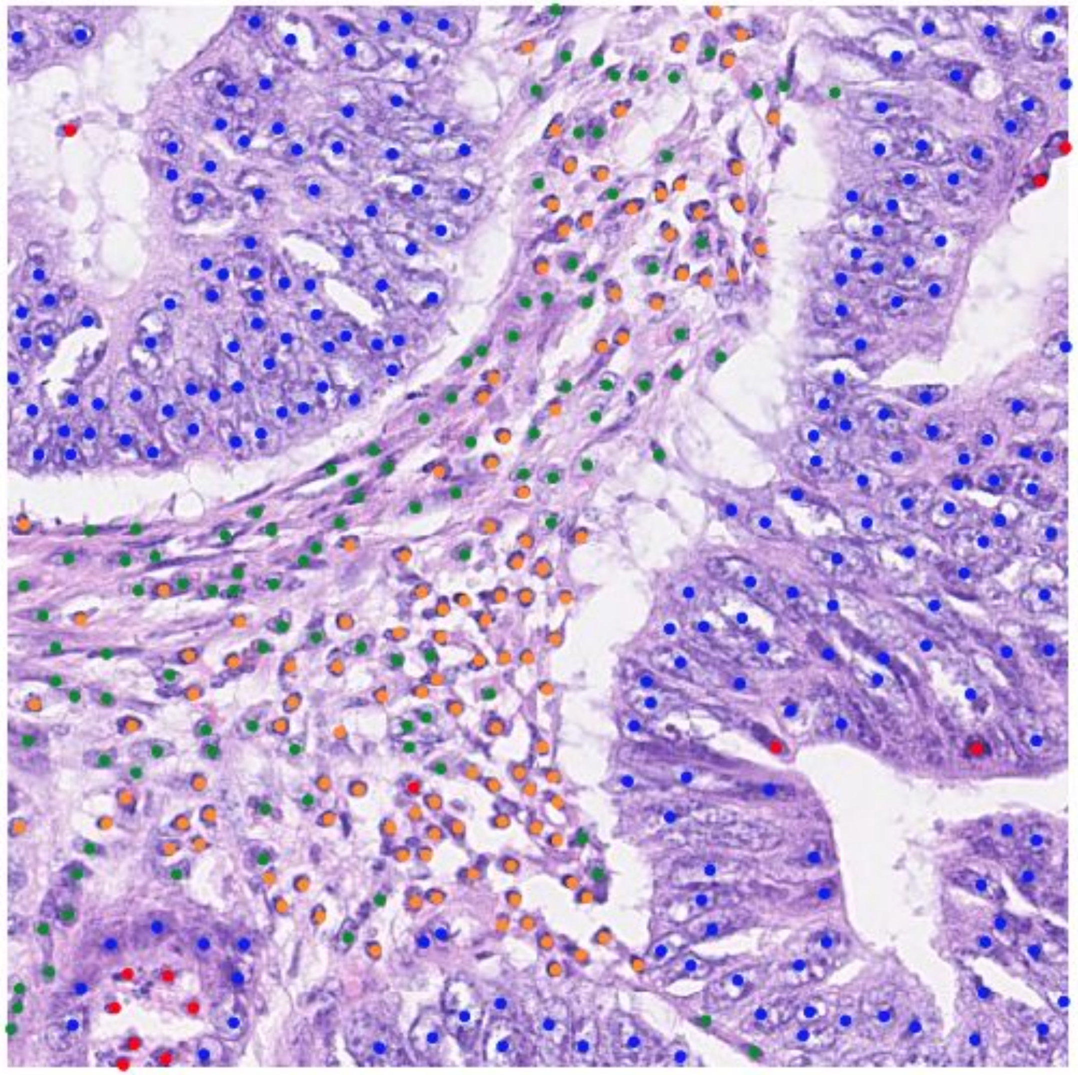
Introduction: The Data

- Subset imaging data used in the Sirinukunwattana et. al. publication (2016)[1]
- Colon cancer histology images (H&E-stained)
  - 30 of the original 100 images
  - Corresponding annotation files with…
    1. The X-Y coordinates of the center of the nuclei
    2. The nuclei labels

The Goal: Semantic Segmentation

- ***Four nuclei subtypes***
  1. epithelial,
  2. fibroblast,
  3. inflammatory, and
  4. other
- Total number of annotated nuclei: 11,004
- ***Original goal***: classify each pixel with the correct nuclei subtype

Introduction: Data Example

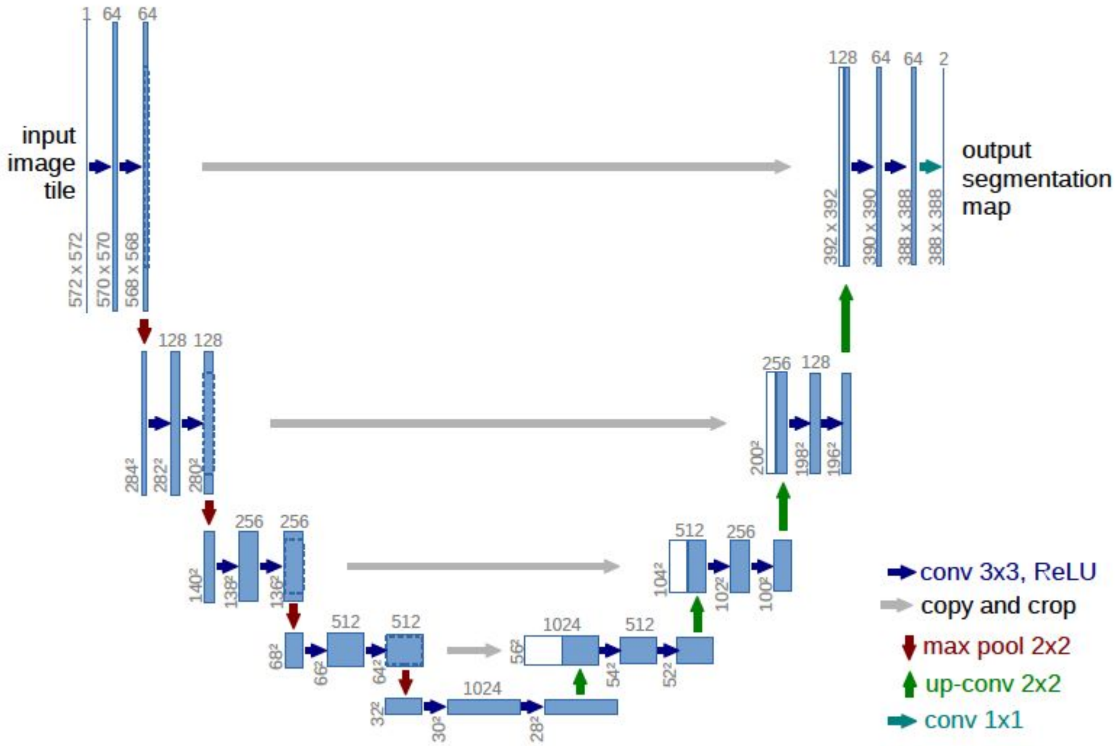


- Epithelial
- Fibroblast
- Inflammatory
- Other

Methods: TensorFlow U-Net[2]

U-Net: Convolutional Networks for Biomedical Image Segmentation

Olaf Ronneberger, Philipp Fischer, and Thomas Brox



General Outline of U-Nets

- Fully Convolutional Network
  - Consists of convolution, pooling, and up-convolution layers
  - ReLU activation function
  - Crops and copies early stages of convolution

- Designed for biomedical image segmentation[2]

Methods: ***tf_unet*** Package

- Package ***tf_unet***[3]
  - ***Primary Author***: Joël Akeret
  - Source code on GitHub (GitHub username: jakeret)
  - U-Net implementation as proposed by Ronneberger et. al.
  - Developed with TensorFlow

Tensorflow Unet

docs passing | arXiv 1609.09077 | ascl 1611.002 | launch binder

This is a generic **U-Net** implementation as proposed by Ronneberger et al. developed with **Tensorflow**. The code has been developed and used for Radio Frequency Interference mitigation using deep convolutional neural networks .

Limitations of ***tf_unet***

1. Images must have even dimensions
2. Unpadded convolutions: output smaller than input (advised ***against*** padding)
3. Possible mini-batch and validation set overlap
4. Specification of learning rate decay is only available for momentum optimizer
5. Incomplete documentation

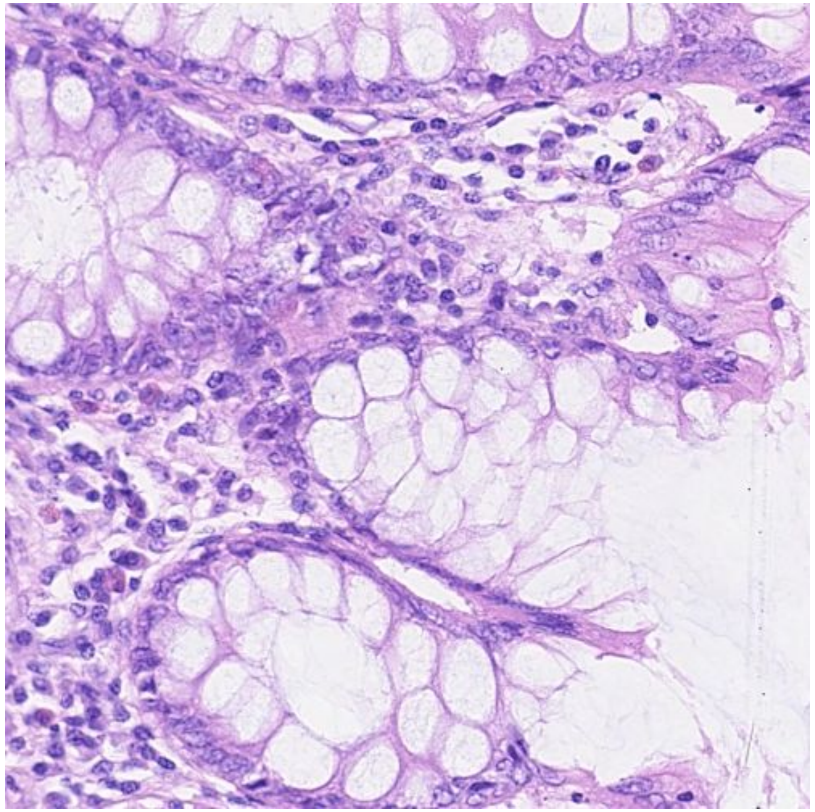
Methods: Data Augmentation

- ***15 images*** randomly selected for training / validation
  - *Each image was randomly cropped **4 times***
    - Cropped image dimensions: 350x350
    - Rotated 0, 90, 180 or 270°
- ***60 total training images*** with corresponding annotation masks

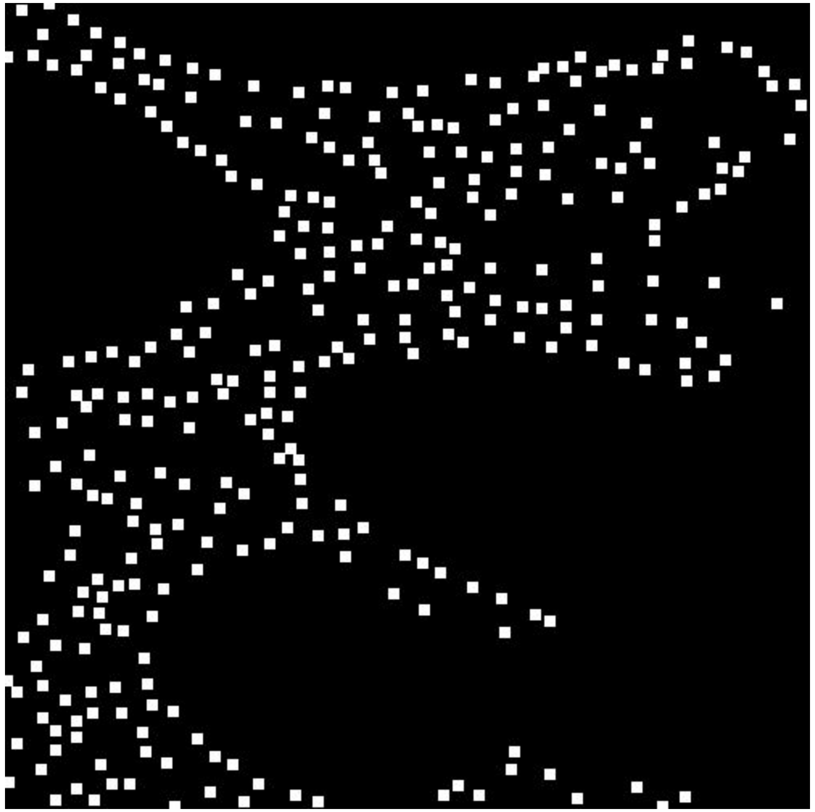
Methods: Semantic Segmentation

- U-Nets are designed for multiclass segmentation
- ~ ***6%*** of pixels represent foreground of image
  - Divided into even smaller subsets when considering the 4 nuclei subtypes
- Tried training model to segment all 4 types, but small sample size and low nuclei counts made this difficult
- Decided to train a model for pixel-wise, binary classification (foreground vs. background)

Methods: Annotation Masks



Dilation with 7x7 kernel of the original annotation masks



Methods: Loss Function

- Foreground-to-background ratio is about ***1:16***
  - Imbalanced classes are known to make classification tasks difficult
  - ***Dice coefficient*** or ***weighted cross entropy*** can be used as the loss function to account for imbalance
  - For this project, the weighted cross entropy was used as the loss function

Methods: U-Net

- Trained a variety U-nets with varying parameters
- Two choices for optimizers:
  1. ***Momentum optimizer*** (allows for learning rate decay in tf_unet package)
  2. ***Adam optimizer***

Methods: U-Net

- Momentum
  - *Performed terribly, no matter what*
    - Weighted cross entropy
      - *Predicted all background*

Methods: U-Net

- Momentum
  - *Performed terribly, no matter what*
    - Weighted cross entropy
    - Deeper architecture (3 layers to 5)
      - *Caused gradients to explode*
      - *Predicted all background*

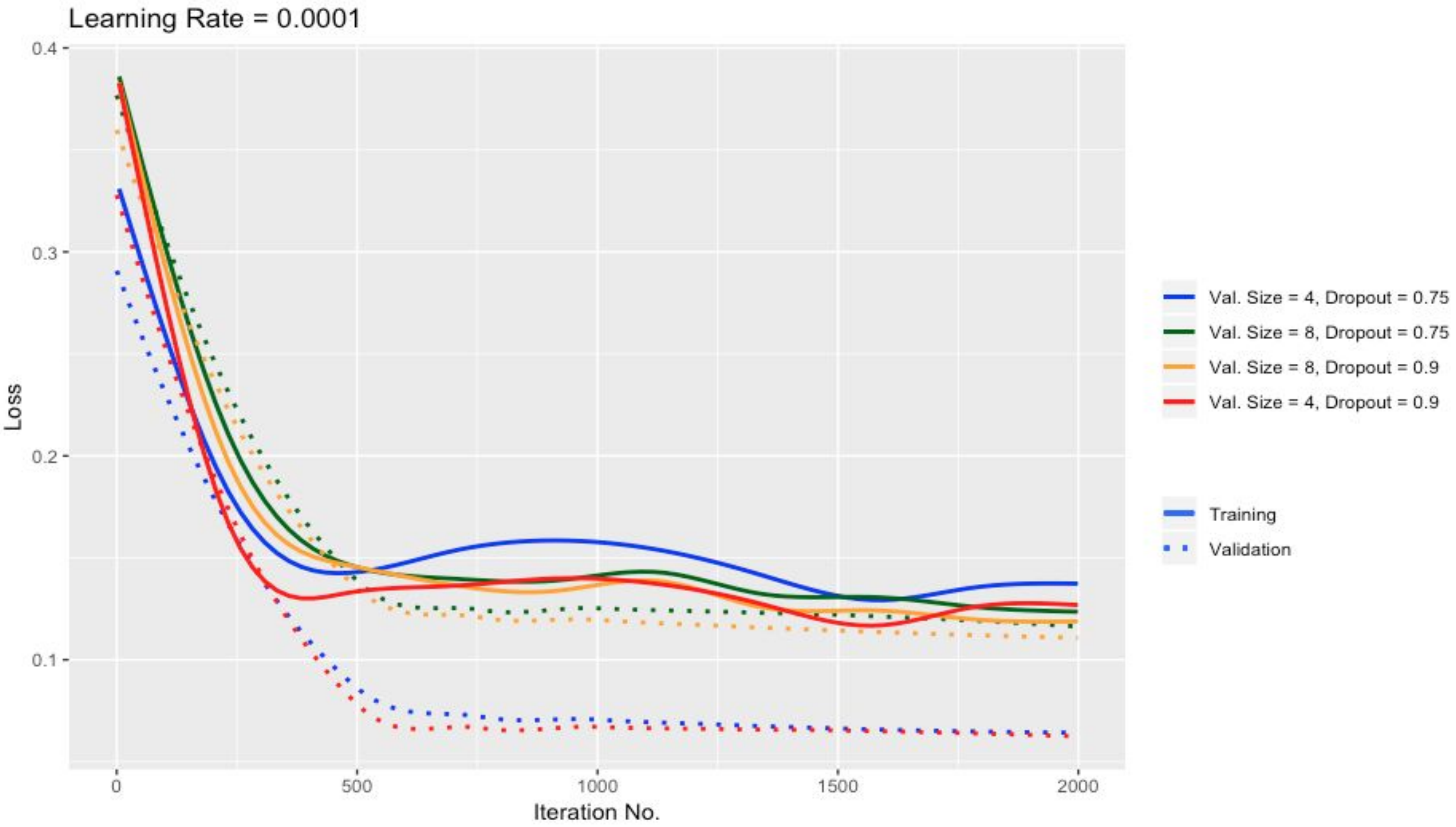
Methods: U-Net

- Momentum
  - *Performed terribly, no matter what*
    - Weighted cross entropy
    - Deeper architecture (3 layers to 5)
    - Dice coefficient
      - *Predictions were sporadic / unreliable*
        - Often ranged from $2e^{-5}$ to 0.0007

Methods: Final U-Net

- For final models, Adam optimizer was selected
- Other parameters specified:
  - Number of layers: 3
  - Number of feature roots: 32
  - Learning rate: 0.0001
  - Number of epochs: 100 epochs
  - Number of training iterations: 20
  - Loss function: weighted cross entropy
  - ***Dropout rate: 0.75 and 0.9***
  - ***Validation batch size: 4 and 8***

Results: Training & Validation Loss



Results: Comparison of Models

	Validation Size: 4 Dropout: 0.75	**Validation Size: 4 Dropout: 0.9**	**Validation Size: 8 Dropout: 0.75**	**Validation Size: 8 Dropout: 0.9**
Accuracy	0.9313	**0.9337**	0.9296	0.9280
Precision	0.6488	**0.6771**	0.6114	0.5932
Recall	0.4374	0.4171	0.4970	**0.5184**
F1	0.5226	0.5162	0.5483	**0.5533**

Conclusions

Overall…

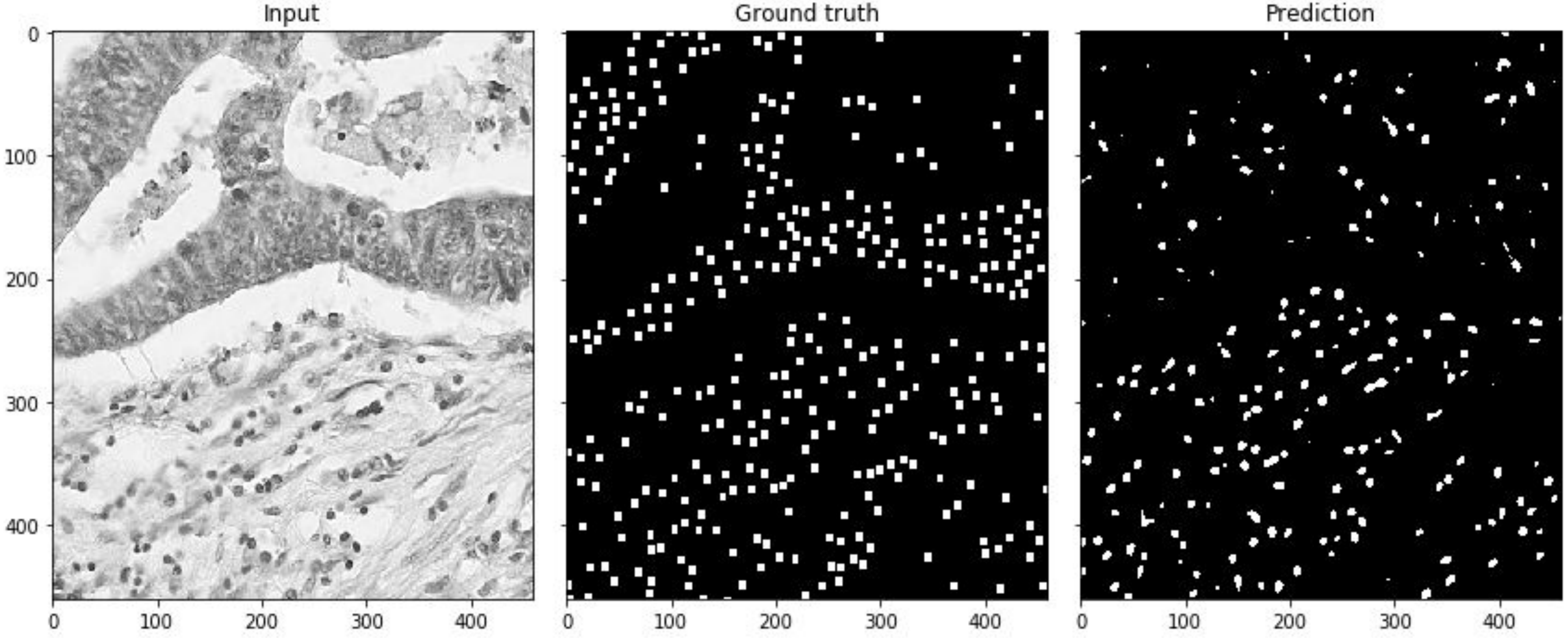
1. Momentum optimizer did not perform well in the presence of heavily imbalanced data

2. Increasing validation batch size and dropout rate resulted in higher recall and F1 scores

Limitations of the Project

1. Imbalanced classes (even in the scenario of binary classification)
2. U-Nets originally designed for single-channel images
3. Exploration of more parameters/increase training iterations to improve performance
4. Annotation masks generated via dilation, thus the ground truth indicates cells are perfect squares
5. Epithelial cells predicted poorly, potentially due to less distinct structure than other cell subtypes

Limitations of the Project



References

1. K. Sirinukunwattana, S.E.A. Raza, Y-W Tsang, D.R.J. Snead, I.A. Cree, and N.M. Rajpoot. “Locality Sensitive Deep Learning for Detection and Classification of Nuclei in Routine Colon Cancer Histology Images”. IEEE Transactions on Medical Imaging. 2016;35(5):1196-1206. doi:10.1109/tmi.2016.2525803.
2. O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” *Proc. Medical Image Computing and Computer-Assisted Intervention*, 2015, pp. 234–241.
3. J. Akeret, C. Chang, A. Lucchi, and A. Refregier, “Radio frequency interference mitigation using deep convolutional neural networks”. *Astronomy and Computing*, vol. 18, pp. 35–39, Jan. 2017. https://doi.org/10.1016/j.ascom.2017.01.002.