Deep Learning for Segmentation and Counting within Microscopy Data

Carlos Xavier Hernández and Mohammad Muneeb Sultan

CS231N: Convolutional Neural Networks for Visual Recognition Class Project

Motivation

Cell counting is a ubiquitous, yet tedious task that would greatly benefit from automation. From basic biological questions to clinical trials, cell counts provide key quantitative feedback that drive research. Unfortunately, cell counting is most commonly a manual task and can be time-intensive. The task is also made more difficult due to overlapping cells, existence of multiple focal planes, and poor imaging quality, among other things. Here, we attempt to automate both the segmentation and counting of cells for a given microscopy image. We hope to create a useful tool for biologist and ultimately expedite research.

Previous Work

At the moment there is no state-of-the-art convolutional neural network (CNN) architecture for counting objects in an image, much less counting cells. However, there is quite a bit of ongoing research in the field of image segmentation using CNNs on which we based our methodology:

- Mask R-CNN (arXiv:1703.06870)
- Feature Pyramid Networks (arXiv:1612.03144)
- Uncertainty in Bayesian Deep Learning for Computer Vision (arXiv:1703.04977)

There has also been previous work done in the task segmenting and classifying cell types using CNNs:

- DeepCell (10.1371/journal.pcbi.1005177)

Dataset

We used the BBBC005 dataset from the Broad Institute's Bioimage Benchmark Collection. This dataset is a collection of 9,600 simulated microscopy images of stained cells. These images were simulated for a given cell count with a clustering probability of 25%. Focus blur was simulated by applying Gaussian filters to the images. Each image is 696 x 520 pixels in 8-bit TIFF format (eventually converted to JPEG and scaled down to 256 x 192 pixels), with cell areas matched to the average cell areas of human U2OS cells. Of the 9,600 images, 600 images have a corresponding foreground mask.

For each experiment, we used an 80/20 split for training and validation.

Cell Segmentation

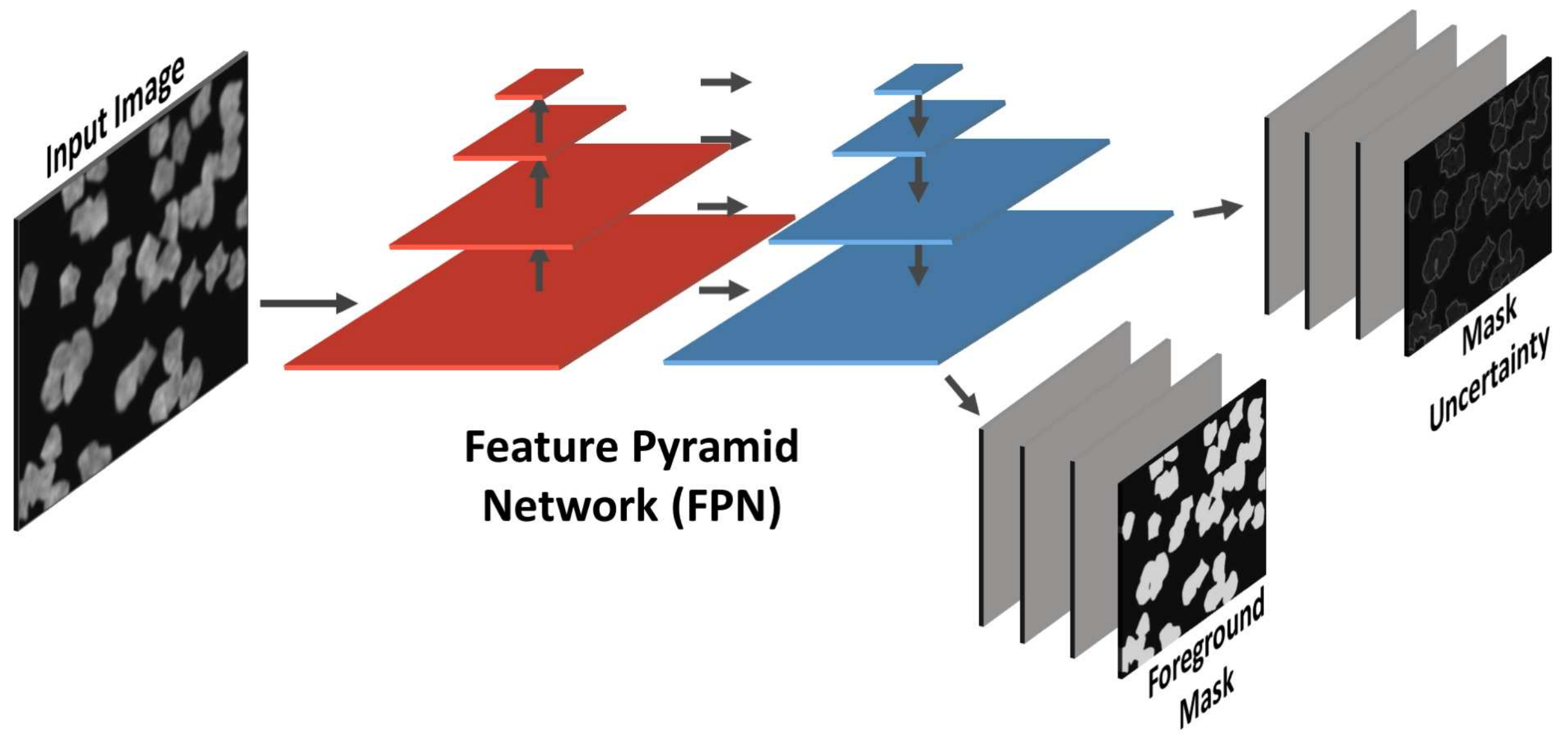

Figure 1: A schematic of our Feature Pyramid Network for generating a foreground mask.

Cell Counting

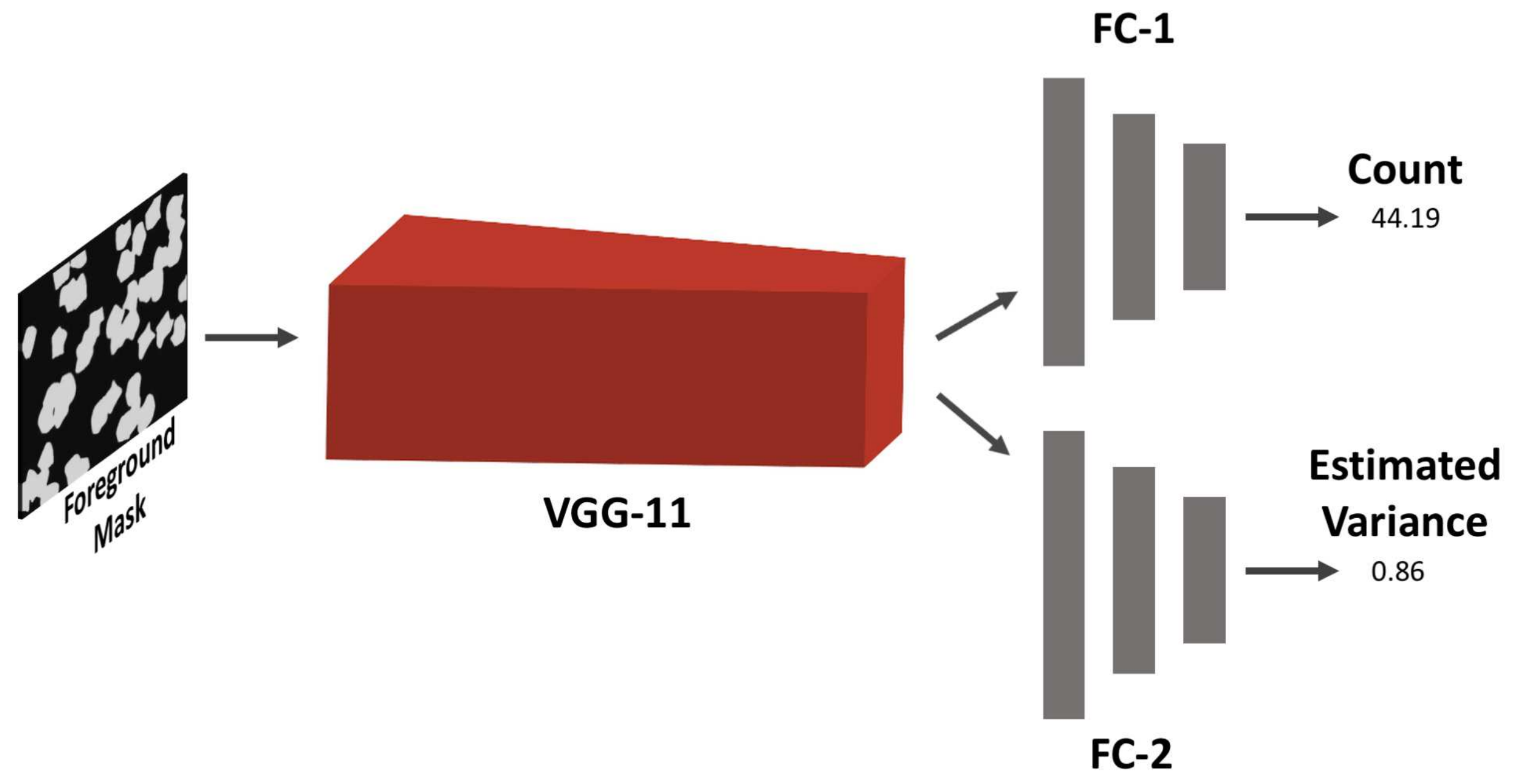

Figure 2: A schematic of our VGG-11-based network for counting cells from a foreground mask.

Experiments

Pre-trained FPN on 480 images with known foregrounds:

- Backprop on L1 (w/ aleatoric uncertainty) and TV losses for each output mask using ADAM optimizer
- Converged validation MSE on 120 validation images after 50 epochs

Trained VGG-11-based network on 7,680 FPN-generated masks:

- Backprop on L1 (w/ aleatoric uncertainty) loss with cell counts using ADAM optimizer
- Converged validation MSE between counts on 1,920 validation images after 50 epochs

Results

After 50 epochs of training, our best model is able to achieve an R^2 value of .987. 80% of the time, the ground truth falls within the predicted 95% confidence interval.

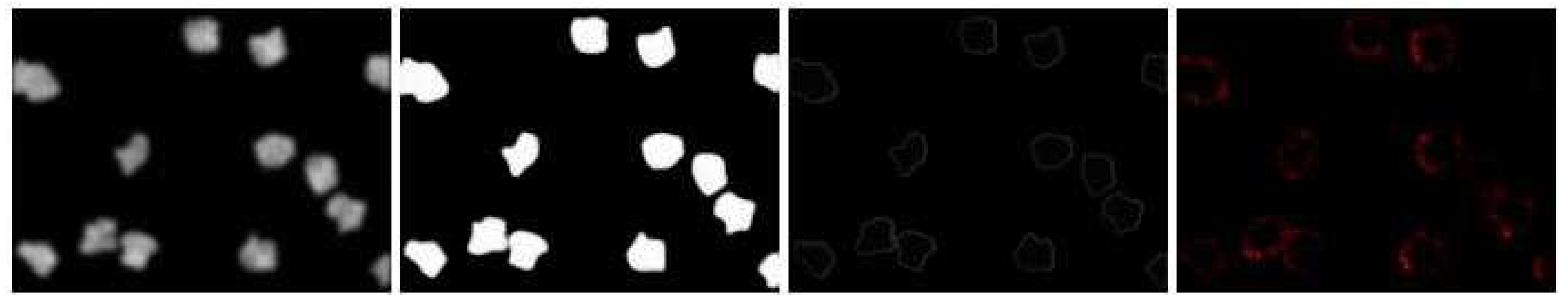

Figure 3: A validation example from our model with 14 cells. The left-most sub-figure is the input image to our model. To its right is the predicted foreground mask for the input and its associated uncertainty, respectively. The right-most sub-figure depicts the saliency map during counting. Our model predicts that this image has 14.00 ± 1.82 number of cells with 95% confidence.

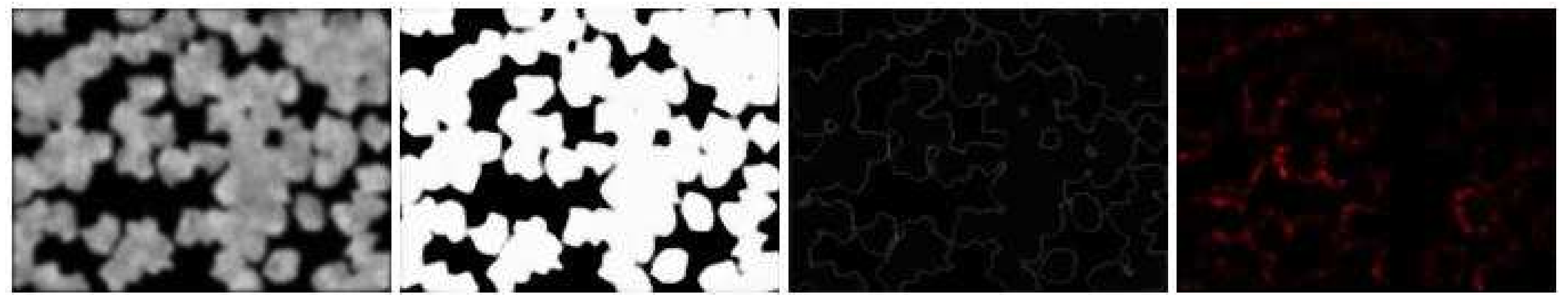

Figure 4: Model thinks this image has 96.17 ± 3.62 cells. Ground truth is 96 cells.

Failure Cases

We find that our method systematically fails in a number of cases:

- Lots of Overlapping Cells

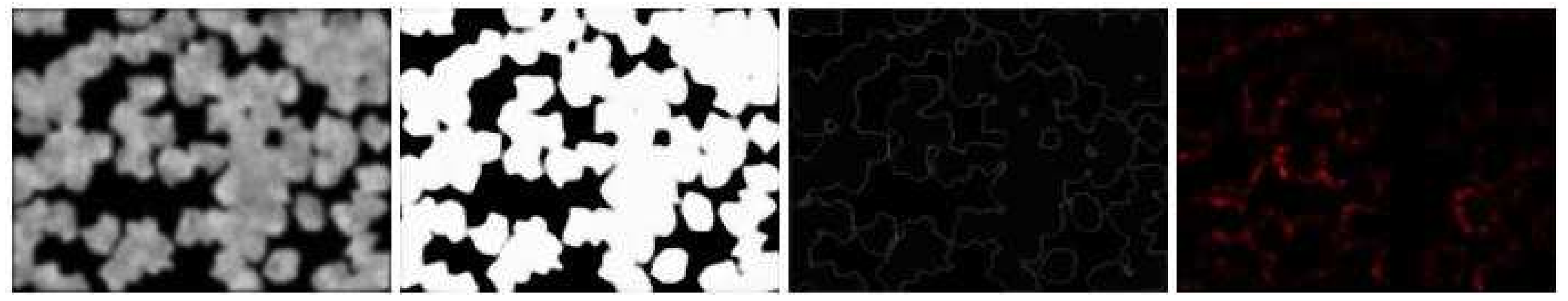

Figure 5: Model thinks this image has 93.50 ± 3.51 cells. Ground truth is 100 cells.

- Oddly Shaped Cells

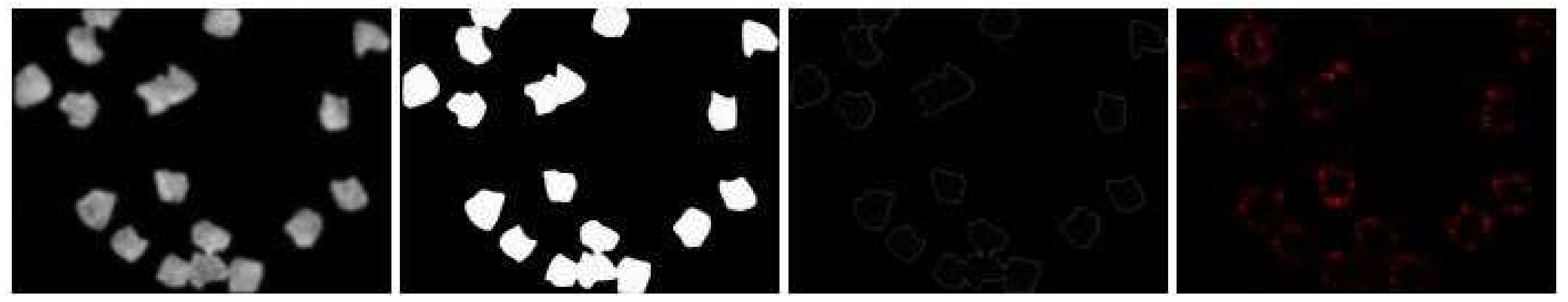

Figure 6: Model thinks this image has 22.83 ± 2.19 cells. Ground truth is 18 cells.

- Bad Focal Planes

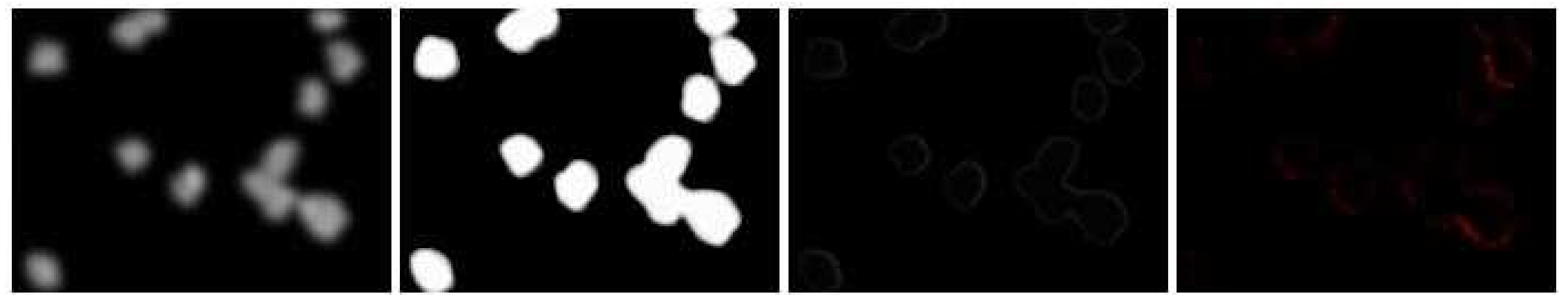

Figure 7: Model thinks this image has 11.85 ± 1.77 cells. Ground truth is 14 cells.

Conclusion

We show that it is possible to design and train a CNN architecture to count cells in microscopy images with relatively good accuracy. While there are a number of failures cases that arise, we believe that better training data might help to overcome these. Specifically, the dataset used lacked foreground masks for out of focus images, which could have helped greatly in improving performance.

We also demonstrate a good use-case for aleatoric losses in estimating uncertainty in cell counting. As the eventual goal is to creating a scientific tool, generating error bounds makes improves the statistical power of our method and the ability to form hypotheses based on its results.

Acknowledgements

We would like to thank Bharath Ramsundar and Timnit Gebru for helpful conversations and feedback.

All code (as well as this poster and a manuscript) will be made publicly available on GitHub (https://github.com/cxhernandez/cellcount).