

U-turning on Deep Learning Methods for Cell Counting - A Bayesian Regression Approach

Andrew Pohl¹, Dylan Loader², and MingKuan²

¹Faculty of Kinesiology - *University of Calgary*

²Department of Mathematics and Statistics - *University of Calgary*

Introduction: blah....

Methods: Supplied training data was split randomly into a training set (80%) and validation set (20%) to allow for validation of cell count prediction models. Parameters for prediction of cell counts were estimated from the 1920 images in the training set and then performance assessed on the 480 images within the with-held validation set. To examine the effectiveness of U-Net image pre-processing a Bayesian implementation of polynomial regression was applied to predict the number of cells from either raw images or U-Net generated image mask. To account for increased likelihood of cell overlap with increasing cell count, the sum of the normalised pixels for each image x_i along with its square of were used as covariates to estimate cell count for each image (y_i). Three polynomial regression models were examined: (i) a single level model where parameter values were estimated from all images, (ii) a random slopes model where slope regression parameters (β_1, β_2) were allowed to vary for each stain/blur level group $j = 1, \dots, 6$ and (iii) a model where both slopes (β_1, β_2) and intercepts (β_0) were allowed to vary for each group. The full varying slope and intercept model is outlined in (1). Parameters of each model were estimated via Markov Chain Monte Carlo with weakly informative priors were utilised for each parameter.

$$\begin{aligned} y_{ij} &\sim N(\mu_{ij}, \sigma^2) \\ \mu_{ij} &= \beta_{0ij} + \beta_{1ij}x_{ij} + \beta_{2ij}x_{ij}^2 \end{aligned} \tag{1}$$

The U-Net image preprocessor consisted of a 5 layered encoder and subsequent 5 layer decoder neural network trained on 830 images with ground-truth segmentation masks obtained from the Broad Bioimage Benchmark Collection. To increase training set size (and improve image segmentation performance) a data-generator was constructed which rescaled images and performed random rotations along with horizontal and vertical image shifts. The network was trained using the ADAM optimiser and dice coefficient loss. Training was performed for 7 epochs with a batch size of 64 based on a early stopping condition when no improvement was observed on a random 5% validation set.

Results: Despite being effective at reducing blur and creating clear image segmentation masks, preprocessing