Report

June 1, 2025

1 Homework 4 Report

1.1 Code Explanation

This solutions builds two fully convolutional networks that take 256×256×3 images normalized to [–0.5,0.5] via a Lambda layer, apply a few 3×3 ReLU conv layers (e.g., 32→64→128 filters), and end with a 1×1 conv: sigmoid activation for perpixel tree/notree classification (trained with Adam@4e-4, binary cross-entropy) and linear activation for perpixel height regression (trained with Adam@ and MAE loss). Each model is trained for 10 epochs with a 10% validation split, evaluated on the test set for accuracy or MAE, and we visualize input images alongside ground-truth and predicted masks/heights to verify performance.

1.2 Discussion Questions

1. Describe and analyze the performance of both models. Are you seeing underfitting or overfitting (or both)?
    - Both models show steady improvement on both training and validation sets with only small gaps between them, indicating neither extreme overfitting nor severe underfitting. The classification CNN's training accuracy rises from about 0.74 to 0.89 while validation accuracy climbs from 0.85 to 0.88, and both losses decrease steadily; the final test accuracy of ~0.896 is close to the validation performance, so there's no large generalization gap. Similarly, the regression CNN's training MAE drops from ~0.029 to 0.022 while validation MAE hovers around 0.024 to 0.022, and the test MAE of 0.0206 is in line with that trend. In both cases, the training curves and validation curves run almost parallel, so I am not seeing significant overfitting or underfitting, just a consistent, modest gap that suggests the models are capturing most of the signal but might benefit from a bit more capacity or data variety.
2. What would be your next step to try to improve the performance of your models?
    - I would first incorporate simple data-augmentation techniques to make the networks more robust to variability in canopy appearance. Next, I'd experiment with slightly deeper or wider architectures—adding another conv block or increasing filter counts—and possibly introduce batch normalization and/or dropout to help the models learn richer features without overfitting. Finally, tuning the learning rate schedule and trying alternative optimizers or loss weightings could yield incremental gains in both classification accuracy and height-prediction MAE.