Jordan Becci
CAP5516
PA3
4/15/24

Introduction:

This model uses microscopic images from the NuInsSeg dataset of cells to locate and find individual nuclei. A lightweight version of SAM, MobileSAM, was implemented along with LoRA for efficient fine-tuning. The NuInsSeg dataset contains H&E-stained histological images with annotated nuclei instance masks which make it ideal data for segmentation tasks.

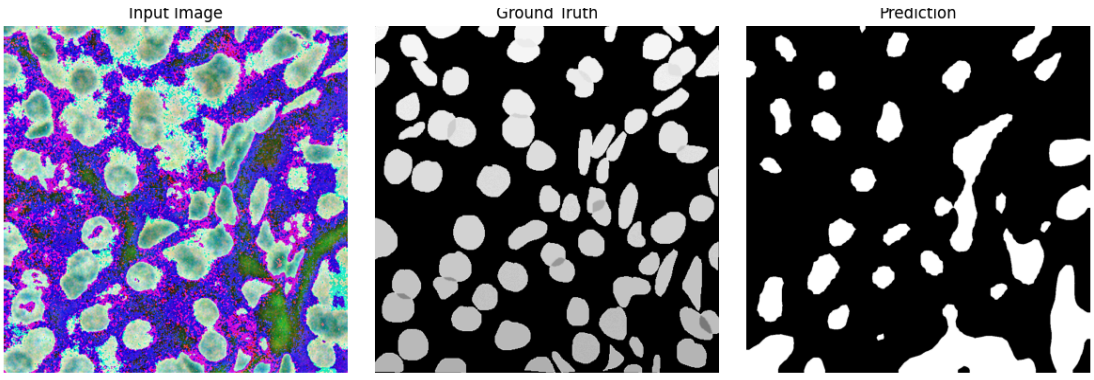
Tasks:

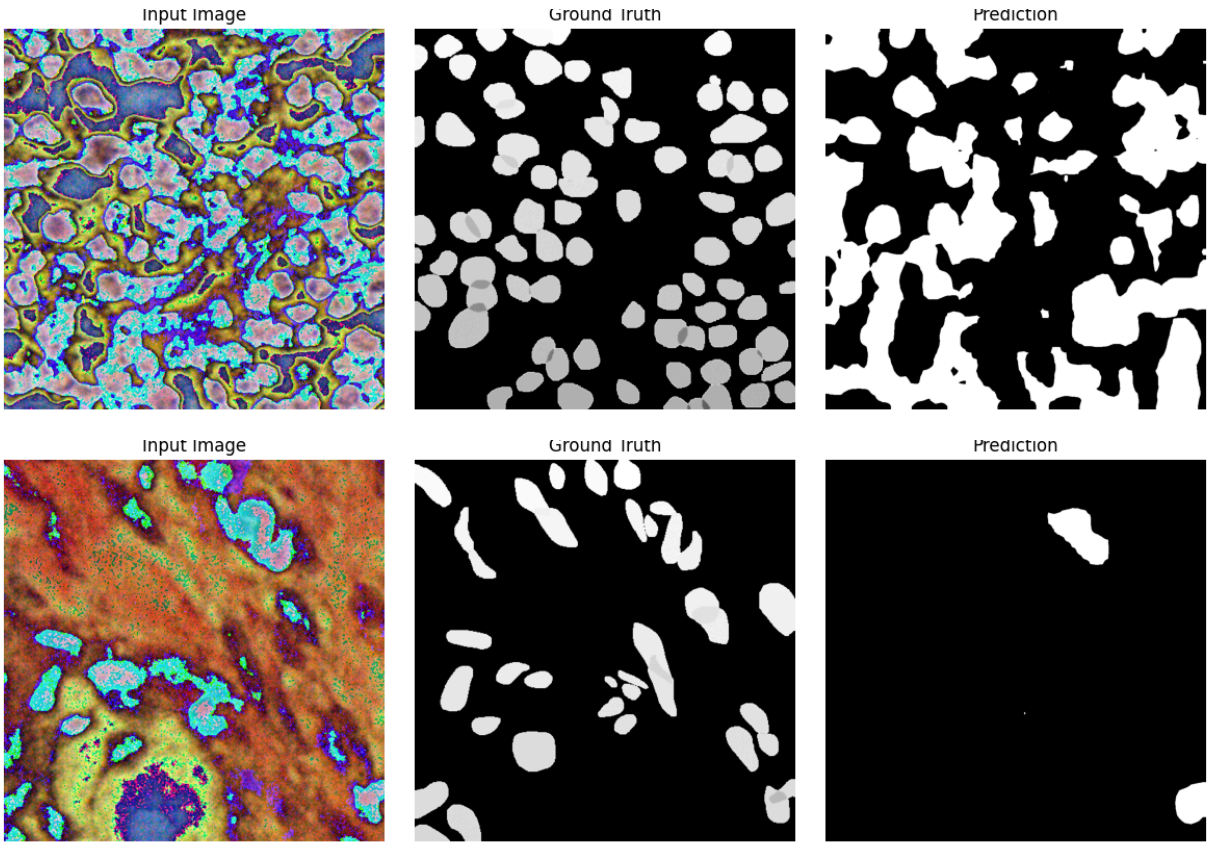
1) Implementation:
   metrics.py: Defines the AJI and PQ metrics.
   dataset.py: Images and masks were loaded, resized to 1024x1024, converted to tensors, and normalized to be used in training/evaluation. Dataset was preprocessed.
   Lora_utils.py: Introduces lightweight fine-tuning layers to MobileSAM encoder, without changing the core parameters. LoRA was applied to linear layers in the ViT where the name contains query, key, value projections in the transformer. LoRA freezes the SAM model and injects trainable adapters which reduces the trainable parameters to 89,808.
   train_eval_lora_sam.py: Training was performed using 5-fold cross-validation, with hyperparameters (batch size=4; epochs=10; adam optimizer; dice loss). A multi-layer head was fine-tuned for 5 epochs, hoping it would improve boundary detection compared to the 1x1 conv layer I tried first. Evaluation was performed on each predicted mask and metrics (Dice, AJI, and PQ) were saved to csv files along with comparison images.

2) Metrics:
3) & Tunable parameters:

Model	# Parameters	Avg. Dice (%)	Avg. AJI (%)	Avg. PG (%)
MobileSAM + LoRA	89808	25.32	5.84	4.82

4) Visualizations:





These visualizations show the preprocessed input image, since the input image comes from dataset.py, with its correct mask and the predicted mask showing that there are errors.

Conclusion

I was able to begin implementing a model for nuclei segmentation using LoRA and MobileSAM, however, the metrics I achieved were lower than baseline. There may have been issues with preprocessing and using too simple of an initial segmentation head leading to poor segmentation results. Future improvements could be made by having a stronger GPU to train more epochs. Using a stronger model could help such as full SAM or U-Net. While I focused on semantic segmentation, to get a better AJI and PQ instance-based methods may be better.