

## **Assignment 03: Parameter Efficient Fine-Tuning Model for Nuclei Instance Segmentation**

### **Introduction:**

Often times when creating a machine learning model for a particular task, the model will not have been trained on that particular task. The model therefore has to be fine-tuned on the type of data that it will be used to interpret and process. For our application, we will be using Facebook/Meta's Segment Anything Model (SAM) to perform segmentation of medical images in the Nuclei Instance Segmentation (NuInsSeg) dataset. The SAM has not really been trained on Medical imaging data, so it has to be fine-tuned on the NuInsSeg dataset in order to work effectively at image segmentation.

There is a problem however. Training the SAM requires loading 11 billion mask images and this requires a huge amount of computing power. Fine-tuning the model by changing the weights in the model also requires a correspondingly large amount of computing power.

Another issue is that in trying to fine-tune the SAM model to work with medical data, model could end up forgetting how to do other tasks or just act in unexpectedly negative ways. When trying to change the weights of a model through retraining... There are unexpected things that can happen on such a complex system.

Compared to institutions like Meta, individuals and small research groups might not have the resources or group expertise to effectively adapt models by fine-tuning the weights directly, so it behooves individuals to leave the original training intact.

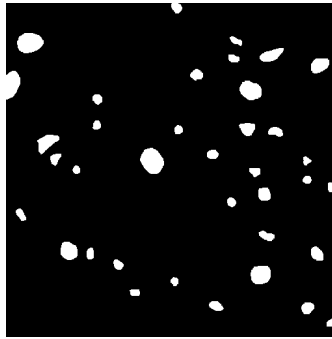
To address this issue, the concept/ technology of Low Rank Adaption (LoRA) was created. LoRA allows machine learning models to be quickly adapted to new contexts and applications. Essentially what LoRA does is to preserve the original weights and training effort from the original model but add another layer within the model that doesn't require much training, can be quickly removed, and can be customized on different datasets. You preserve the base model and add light weight layer on top of it for different situations.

A metaphor for this would be a human being with different sets of clothing and gear. To retrain and adapt a human body for different environments like say to deal with cold weather takes a long time. The human is like the base level machine learning model. However if you give them warm clothing to put on (the LoRA layer), they can quickly adapt to the cold weather. And when the human goes to hot tropical weather, the clothing can quickly be removed in order to adapt to that different environment. The human stays the same, but the clothing changes.

### **Experiment Design:**

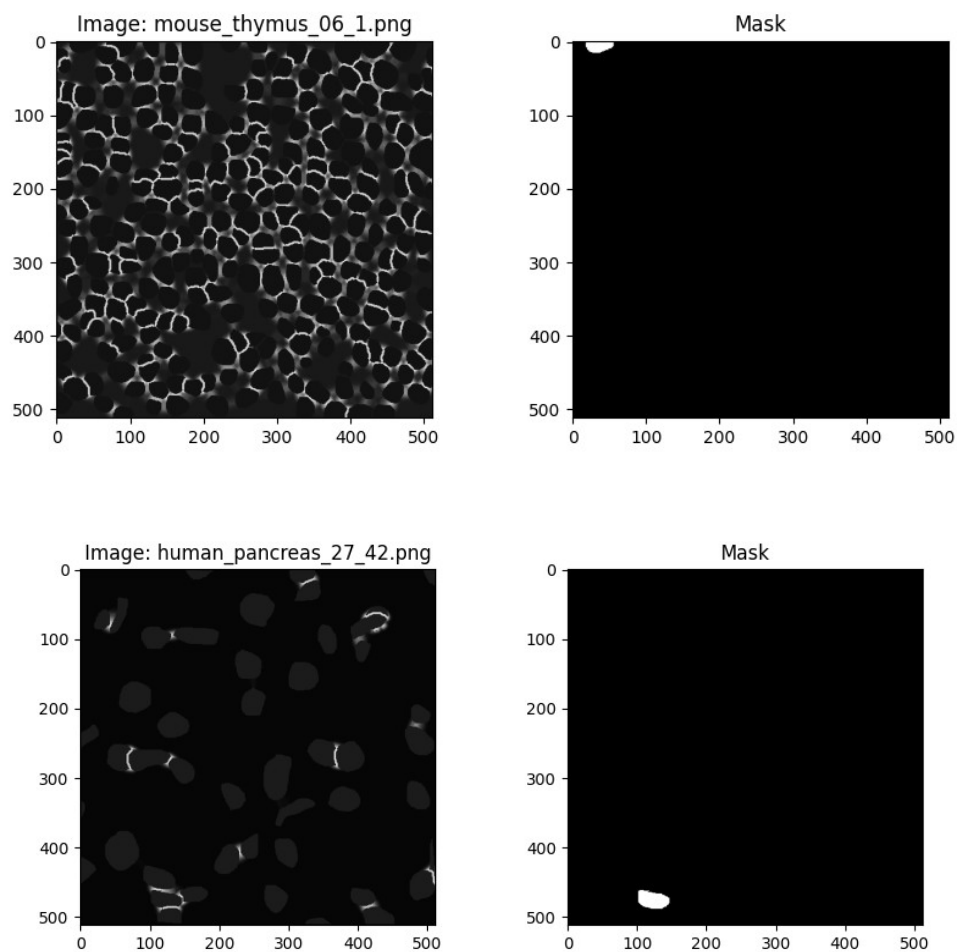
In this experiment, we will be using the Segment Anything Model (SAM) to perform segmentation on the Nuclei Instance Segmentation (NuInsSeg) dataset. The SAM will be adapted to the NuInsSeg dataset using a lightweight LoRA layer.

For the purposes of time, the LoRA layer will be modified to work only with the Binary Mask image of the dataset.



5-Fold Cross-Validation will be performed to measure the accuracy of the modified model.

Here are examples of the Image and the Ground Truth Mask:



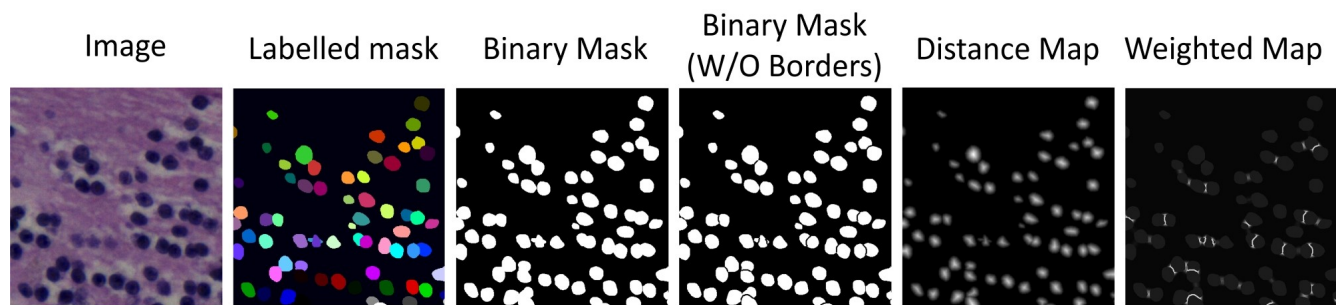
### Dataset Information:

The NuInsSeg dataset contains images patches from the tissue of 31 human and mouse organs. Here is a list of the organs present in this dataset:

Human organs: cerebellum, cerebrum (brain), colon (rectum), epiglottis, jejunum, kidney, liver, lung, melanoma, muscle, oesophagus, palatine tonsil, pancreas, peritoneum, placenta, salivary gland, spleen, stomach (cardia), stomach (pylorus), testis, tongue, umbilical cord, and urinary bladder

Mouse organs: cerebellum, cerebrum, colon, epiglottis, lung, melanoma, muscle, peritoneum, stomach (cardia), stomach (pylorus), testis, umbilical cord, and urinary bladder

Each image folder will contain a certain tissue and there will be sub folders for the different types of masks as shown in the image below.



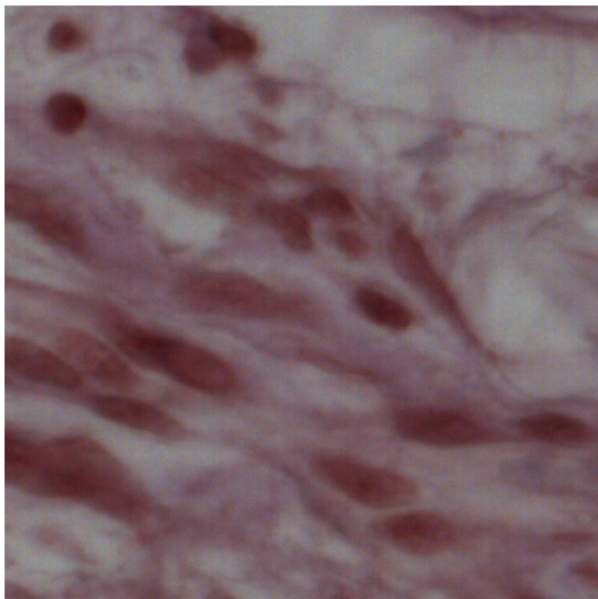
### Training Specifications:

These training specifications were used.

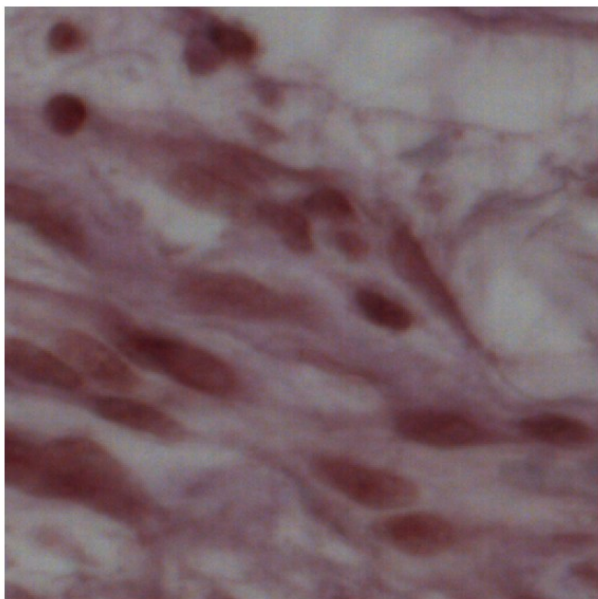
1. GPU: Laptop RTX 4090 16GB of VRAM
2. Batch Size: 4
3. Epochs: 10 per fold
4. Image Dimensions: RGB 512x512
5. Total Parameters: 6,449,918
6. Total Trainable Parameters: 384,386
7. Training Time: 23 minutes for Human Bladder Code
8. Base SAM Model: vit-T, Mobile SAM
8. Model Size in GPU: 13.523 GB

### Training Information

Segmentation using SAM without LoRA:



Segmentation using SAM with LoRA applied:



Original Mask:



### Structure of model with LoRA applied:

Layer (type:depth-idx)	Output Shape	Param #
SAMWithDecoder	[4, 2, 1024, 1024]	--
└TinyViT: 1-1	[4, 256, 64, 64]	321,640
└└PatchEmbed: 2-1	[4, 64, 256, 256]	--
└└└Sequential: 3-1	[4, 64, 256, 256]	(19,488)
└ModuleList: 2-2	--	--
└└└ConvLayer: 3-2	[4, 16384, 128]	(98,944)
└└└BasicLayer: 3-3	[4, 4096, 160]	460,520
└└└BasicLayer: 3-4	[4, 4096, 320]	2,076,600
└└└BasicLayer: 3-5	[4, 4096, 320]	2,504,660
└└Sequential: 2-3	[4, 256, 64, 64]	--
└└└Conv2d: 3-6	[4, 256, 64, 64]	(81,920)
└└└LayerNorm2d: 3-7	[4, 256, 64, 64]	(512)
└└└Conv2d: 3-8	[4, 256, 64, 64]	(589,824)
└└└LayerNorm2d: 3-9	[4, 256, 64, 64]	(512)
└Sequential: 1-2	[4, 2, 64, 64]	--
└└Conv2d: 2-4	[4, 128, 64, 64]	295,040
└└ReLU: 2-5	[4, 128, 64, 64]	--
└└Conv2d: 2-6	[4, 2, 64, 64]	258
Total params: 6,449,918		
Trainable params: 384,386		
Non-trainable params: 6,065,532		
Total mult-adds (G): 48.63		
Input size (MB): 50.33		
Forward/backward pass size (MB): 13448.77		
Params size (MB): 24.48		

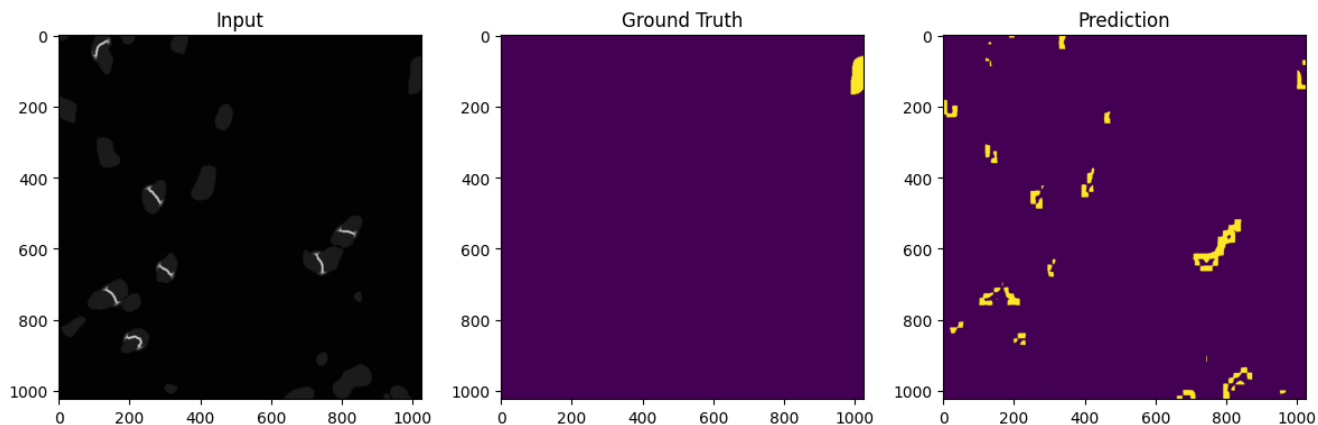
Estimated Total Size (MB): 13523.58

=====

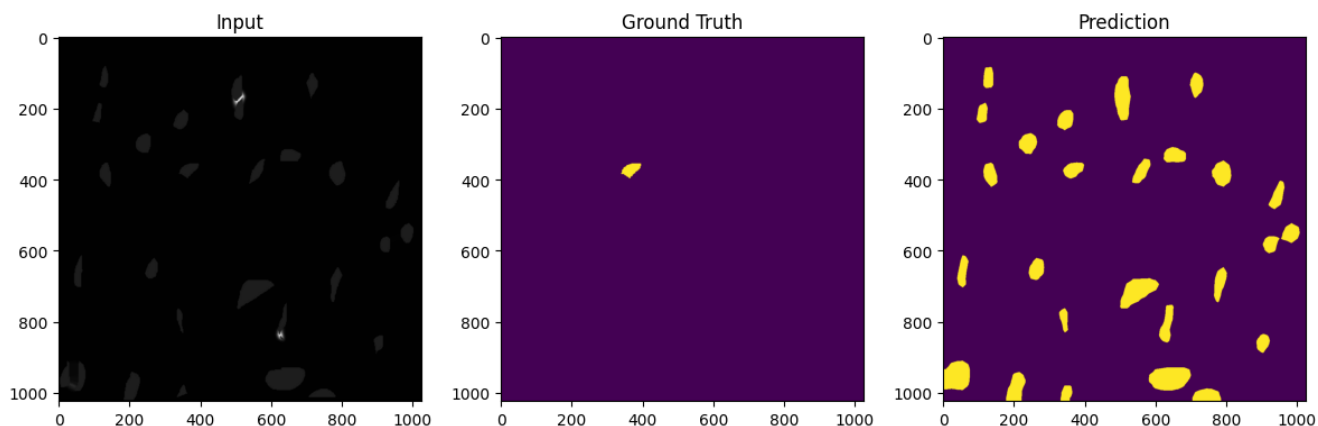
## Experiment Summary:

Due to time constraints I was not able to perform fine-tuning of the SAM model with LoRA.

### Example Predictions for Human Bladder:



### Example Predictions for Mouse Heart:



## Human Bladder Tissue Training Results:

- Dice Score: 0.0402

- **IoU Score: 0.262**
- **PQ Score: 0.304**

**Mouse Heart Tissue Training Results:**

- **Dice Score: 0.051**
- **IoU Score: 0.262**
- **PQ Score: 0.40**