

# Guide to TrainNetwork on Digital Slide Archive

## Overview:

This documentation guides you through the process of training a machine learning model to segment structures from whole-slide images using the HistomicsTK's web interface.

## Instructions:

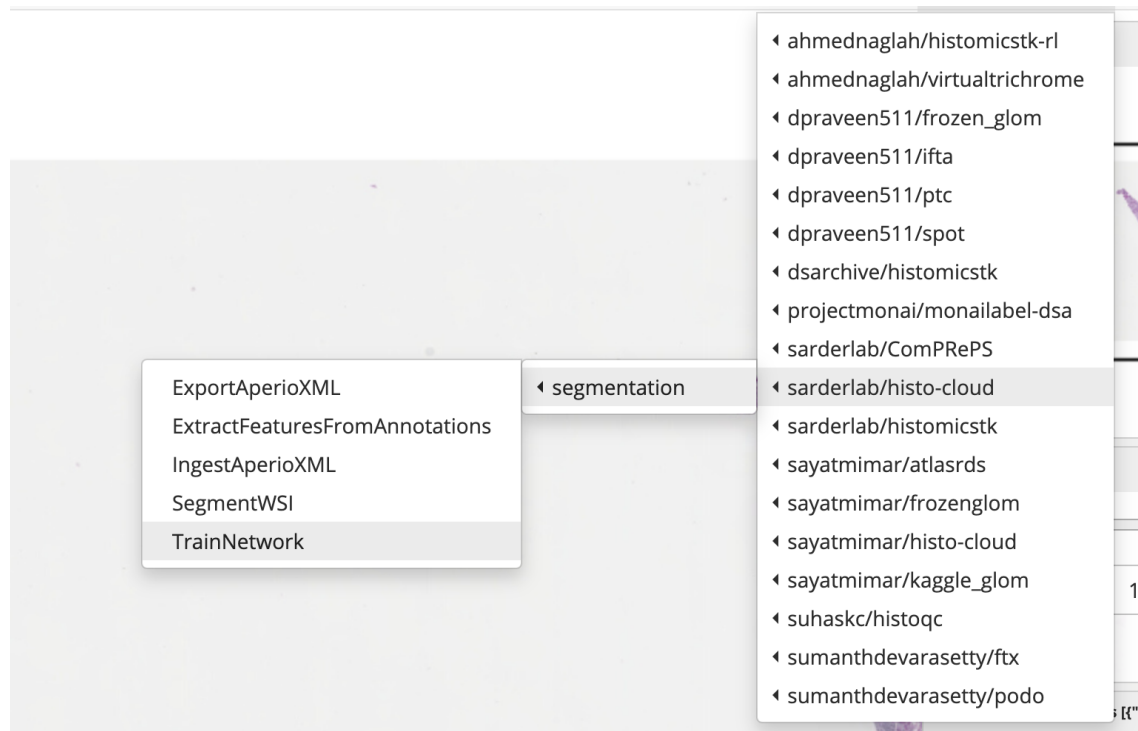
Go to [athena.rc.ufl.edu](http://athena.rc.ufl.edu) to access the Digital Slide Archive and click on 'Register' if you are a new user or 'Log In' if you have an existing account.

### Upload / Access a Whole Slide Image

- Click 'Collections' on the menu to the left to access collections of whole slide images a whole slide image.
- To create a new collection of whole slide images go to 'Collections' and click on 'Create collection', for this you need authorization / admin access.

### TrainNetwork

- On the Whole slide image to segment click on 'Open in HistomicsUI' and it opens a new tab with the selected whole slide image in HistomicsUI.
- Click on Analyses dropdown, select 'sarderlab/histocloud' -> segmentation -> 'TrainNetwork'.



- This will open a new popup on the left-hand side of the screen, where you can input the necessary parameters for the training.

## 1. Input/Output Parameters (IO):

- **Training Data Folder:** Choose the directory containing the slides you want to use for training.
- **Output Model Name:** Specify the desired name for the output model file. By default, this file will be saved in your Private folder.
- **Training Layers:** Provide a comma-separated list of the annotation layer names used for training. Ensure the names match the existing ones. For hierarchies, sub-compartments should follow their parent structures. For instance:  
name1,name2,name3,...
- **Input Model File:** Upload a zip file that contains Tensorflow model files and metadata for the deeplab segmentation network. This will serve as a base for training using transfer learning.

## 2. Whole-Slide Image (WSI) Training Parameters (Advanced):

- **Training Tile Size:** Set the tile size for random cropped WSI regions. Default is 400 pixels.
- **Training Steps:** Define the number of steps for network training. The network will process [steps \* batch size] image patches during training.
- **Global Step:** Specify the global step at the training's start. Useful for continuing training on a previously trained network.
- **Training Batch Size:** Determine the batch size for training on WSI patches.
- **Ignore Class:** Indicate the name of an annotation layer that will be disregarded during training. This lets certain parts of the WSI be ignored for training.
- **GPU IDs:** List GPU IDs that will be used for training. For example, 0.
- **GPU Number of Clones:** Mention the number of GPUs used for training. Adjust only if multiple GPUs are available.
- **Training Patch Scale Rates:** Define the downsampling applied to the WSI patches during training. These values will be randomly used. Example: 1,2,3,4.
- **Network Learning Rate:** Set the base learning rate for the network. This rate will decay over training time.
- **Slow Start Learning Rate:** Indicate the initial learning rate for the network, which should be lower than the base learning rate.
- **Slow Start Steps:** Determine the number of steps the network should be trained using the slow start learning rate.
- **End Learning Rate:** Specify the learning rate's end value. The rate will decay to this specified value.
- **Learning Power:** Set the power for polynomial learning rate decay.
- **Learning Rate Decay Steps:** State the number of steps before the learning rate decays to the end learning rate.
- **Patch Augmentation Percent:** Indicate the percentage of training patches that will be augmented. Note: Augmentation can slow down network training.
- **Fine Tune Batch Norm Layers:** Decide if the batch normalization layers should be fine-tuned. Only set to True if the batch size is sufficiently large.
- **Use Previous Classifier:** Choose whether to use the previous network's classification layer.
- **Last Layer Gradient Multiplier:** For fine-tuning, set a multiplier for the gradients of the last layer.
- **Last Layer Includes Logits Only:** Decide if only the logits' gradients should be boosted.

- **Upsample Logits:** Determine if the logits of the network should be upsampled before calculating the loss.

### **3. Girder API URL and Key (Advanced):**

- **Girder API URL:** Provide the Girder API URL (e.g., `https://girder.example.com:443/api/v1` ).
- **Girder API Token:** Input the Girder token for authentication.

### **4. Starting the Training:**

Once all the parameters are set, click on the `Submit` button to begin the segmentation network training process.