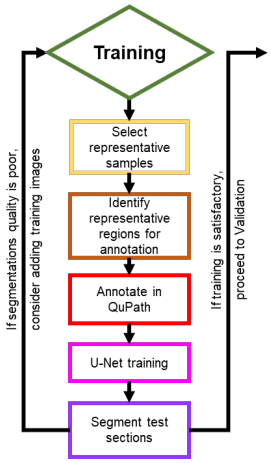
Training

Revision: 12/15/2022 by Madi McElliott and Jeff Beamish



OVERVIEW: This protocol outlines how to prepare a set of U-Net compatible training images for a single stain and then use these images to train a new U-Net model. Repeat this protocol for each stain of interest. This protocol assumes that "Pre-Processing" protocol has been completed for the project you wish to extract the training images from.

BEFORE STARTING:
Set the script directory in QuPath by going to Automate→"Set Script Directory"→selected the folder where QuPath scripts are located. Should include (not all used in this protocol):

- Export Training Annotations.groovy
- Adjust BC.groovy
- Export Section Outlines.groovy
- Separate Sections.groovy

You must have the "U-Net Segmentation" plugin installed in FIJI. If it is not installed, go to Help→"Update…" then select "Manage update sites" and select "U-Net Segmentation" which will add this package to FIJI. You will have to close and restart FIJI after adding.

Unless specified, all macros can either be installed and run OR opened in FIJI and run using the Macro Editor (which is our preference as it makes changing parameters easier, if needed).

You must have a directory of entire section images. See the "**Preprocessing**" protocol for detailed instructions.

STEP ONE: Generate directory for all training files with the following subdirectories (WINDOWS)

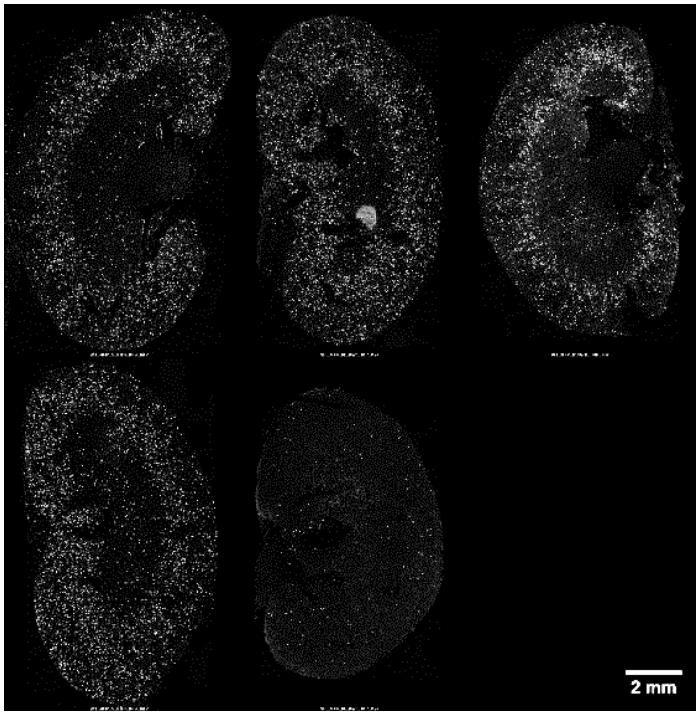
1) "**Original Images Full Size**": These full-size image files of single channel stains must be free of large artifacts, and show representative staining that you want to detect
2) "**Original Images ROIs**": Subset of cropped ROIs from the original full-size images used for training
3) "**QuPath [Stain] Training**": project folder for QuPath, used for annotations
4) "**Annotations (from QuPath)**": All raw annotations exported from QuPath (as *.zip files)
5) "**Annotated Images for Training**": labeled overlay of annotations on original image ROIs to be uploaded to U-Net

STEP TWO: Select single channel images of kidney sections to use for training (WINDOWS)

Select representative samples

*NOTE: This step requires that you already have a directory of whole section, single channel *. tif files. See the "Pre-processing" protocol for detailed instructions.*

1) Open the single channel *.tif files or review the QC montage generated in the "**Pre-Processing**" protocol. Note the example provided is for demonstration only, real project likely will have many more sections to evaluate. An example montage is shown below.



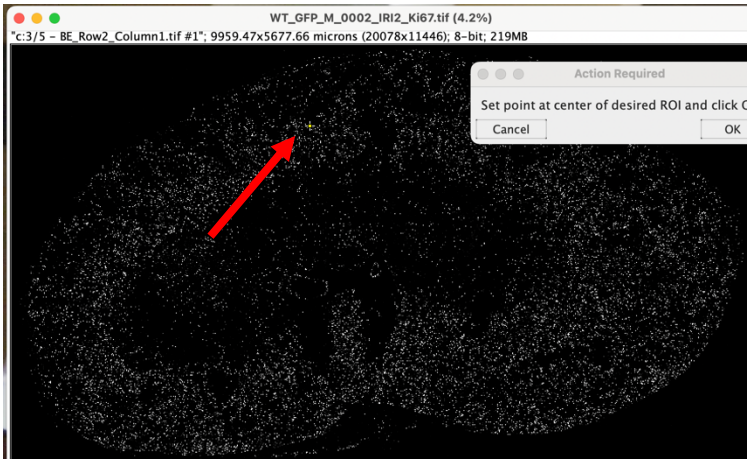
2) Select sections to use for trainings that are free of large artifacts and show representative staining that you want to detect. These files will be in their individual [Stain] folder. See **"Preprocessing"** protocol to generate
3) Copy these selected single channel .tif images into the "**Original Images Full Size**" folder created in **STEP ONE – 1** above.
4) Make sure all images have a per pixel scale set in FIJI. All images must have the same scale. If scale is absent or incorrect, use "Analyze" → "Set Scale…" to set for all images before proceeding.

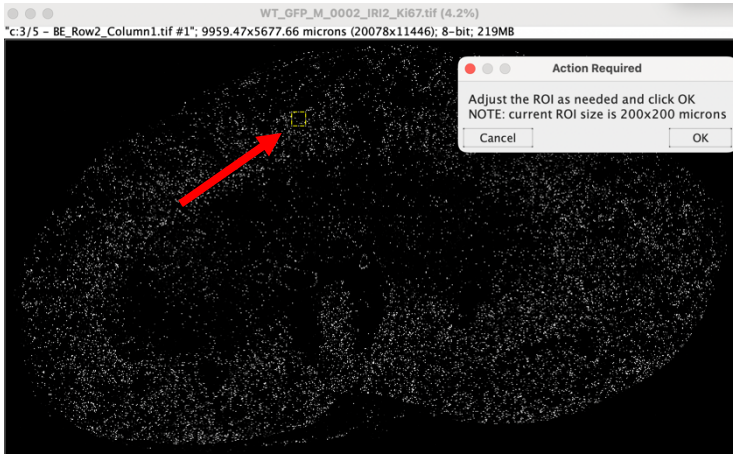
STEP THREE: Generate ROIs for Training (FIJI)

Identify representative regions for annotation

Overall, the goal is to generate at minimum 5 images that represent the images that you'll be evaluating. It is important that the dynamic range of the image's pixel intensities and the resolution match what you're planning to use for the actual images.

1) Install the Macro "Training_STEP_THREE_Make_Training_ROIs.ijm"
    a. Go to FIJI→Plugins→Macros→Install…
    b. Note you can use set this macro as a shortcut in FIJI to allow for repeated ROIs to be made on the same image by adjusting the shortcut key in the macro name in the code (default is "[q]"). You must Install the macro for the shortcut to work. Double check that whatever short cut key you chose does not conflict with existing shortcuts.
    c. The default ROI size is 200 micron square, you may need to adjust this in the code (Line 16) if a different size is needed.
    d. This macro will create a separate subfolder within the "[Stain]" folder called "[Stain]-Extracted Training Images (ALL)" that contains two subfolders of "ROIs (Cropped Images-ALL)" and "ROI Locations"
2) Open the first full-size image from the "**Original Images Full Size**" folder from **STEP TWO - 3**
3) Adjust brightness and contrast to make the stain easy to see
4) Select ROIs to use as training areas. These areas should be free of obvious artifacts and represent the type of targets you want to detect. It is best to avoid areas with ambiguous staining as these will not be good for training.
    a. Run the macro "Training_STEP_THREE_Make_Training_ROIs.ijm" using the shortcut key
    b. When prompted, first click on the general region where you want to put the ROI. Then select OK
    c. The macro will then draw a preliminary ROI. Reposition the ROI as needed by clicking in the center and dragging to a new position. Once positioned, select OK and the macro will save the extracted images into "ROIs (Cropped Images-ALL)" and the coordinates of the cropped ROI will be in "ROI Locations".





5) Repeat **a-c** above. The goal is to choose 5-15 representative images of the [Stain] that encompass the variation in staining. The number of ROIs per full size image should be adequate to encompass the full breadth of staining variation. We generally include a few "negative" areas with only small amounts of positive staining as well.
6) Close the image and load the next image from the "**Original Images Full Size**" folder from **STEP TWO - 3**
7) Repeat steps 3-6 until all of the images within **"Original Images Full Size"**
8) For each set of ROIs for each section use a random number generator to select the image to include. You may also want to select specific images with unique features of interest or negative areas of stain to also include in training. Copy these images (which are located in "ROIs (Cropped Images-ALL)") into a folder "**Original Images ROIs**" created in **STEP ONE -- 2** above.

Annotate in QuPath

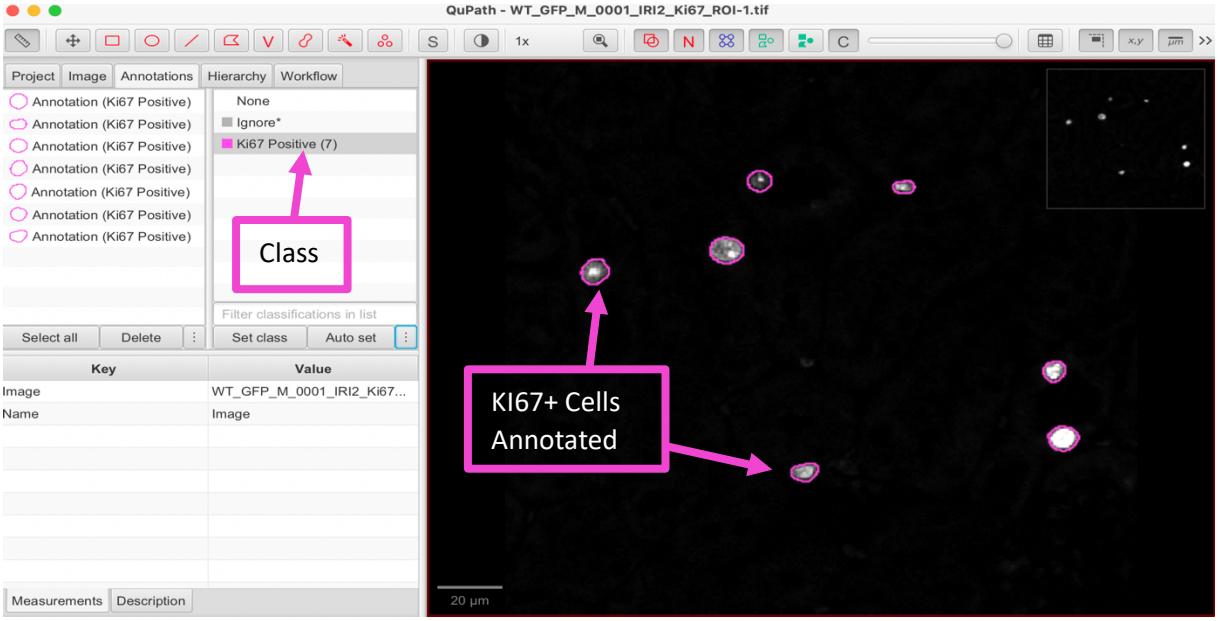
STEP FOUR: Annotate the images (QuPath)

1) Generate a QuPath project using the directory in **STEP ONE - 3** above and load all of the ROI image files from "**Original Images ROIs**" (**STEP THREE - 8** above) into it.
2) If needed in the "Annotations Tab," add JUST ONE class for the target (e.g. "Ki67+"). You can also have a class called "ignore" which you can use to mark areas of the image that are ambiguous and should not be used for training

3) Select the target class (e.g. "Ki67+") then select "Auto set" which will make all new annotations default to this class
4) Use the wand tool (W) or brush tool (B) to annotate the images. The more precise you are, the better the training will be.

Useful keyboard short cuts:

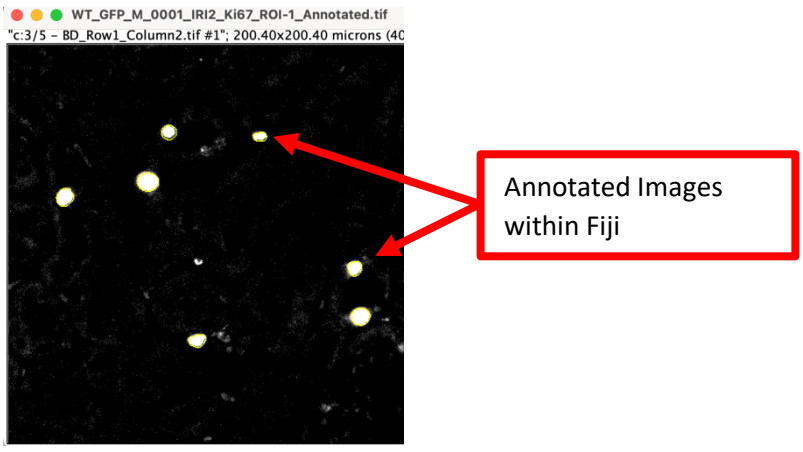
- SPACEBAR-move the image without moving the annotation;
- W-switch to the wand tool (used most of the time)
- B-switch to the brush tool (useful when the wand tool is producing irregular results)
- ALT-hold down while using other tools to REMOVE areas of the ROI
- SHIFT-hold down while using other tools to ADD areas to the ROI that are not directly contiguous
- CTRL AND SHIFT-allows you to approach other annotations without overlapping with them



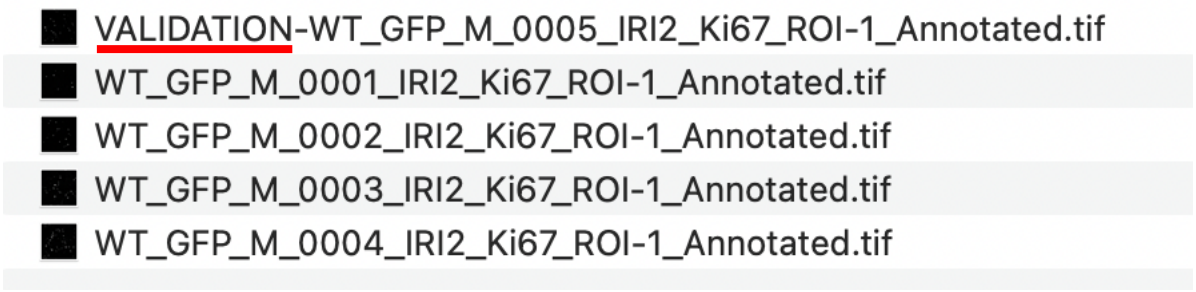
5) Be sure to Save once done annotating all of the images.
6) Run the QuPath script "Export Training Annotations.groovy" using the "Automate" menu in QuPath. This script with generate *.zip files for each image that contain all annotations (which are stored as ImageJ "ROIs"). The *.zip files will be stored in the QuPath project directory.
7) Move these *.zip files into the "**Annotations (from QuPath)**" folder created in **STEP ONE – 4**.

STEP FIVE: Generate the UNET compatible training Images by adding the annotations as an overlay (FIJI)

1) Check that the "**Annotations (from QuPath)**" (filled in **STEP FOUR - 6**) and "**Original Images ROIs**" (filled in **STEP THREE – 8**) have the same number of files and corresponding names. The following macro does not check names.
2) Run the macro "Training_STEP_FIVE_Add_Annotations.ijm". Enter the following:
    a. Name for annotations "[Stain]+" (e.g., Ki67+)
    b. Unannotated Training Image Directory = "Original Images ROIs" (filled in **STEP THREE – 8**)
    c. Annotation Directory = "Annotations (from QuPath)" (filled in **STEP FOUR -- 6**)
    d. Output Directory = "Annotated Images for Training" (created in **STEP ONE – 5**)



3) Select about 10-20% of these images to be used as validation during training. Rename these files with "VALIDATION" as a prefix to the file name to make them easy to find later (Note these files will be used for reference during training only and **ARE NOT** the same as the independent images will be annotated and analyzed for formal validation, see "Validation" protocol).

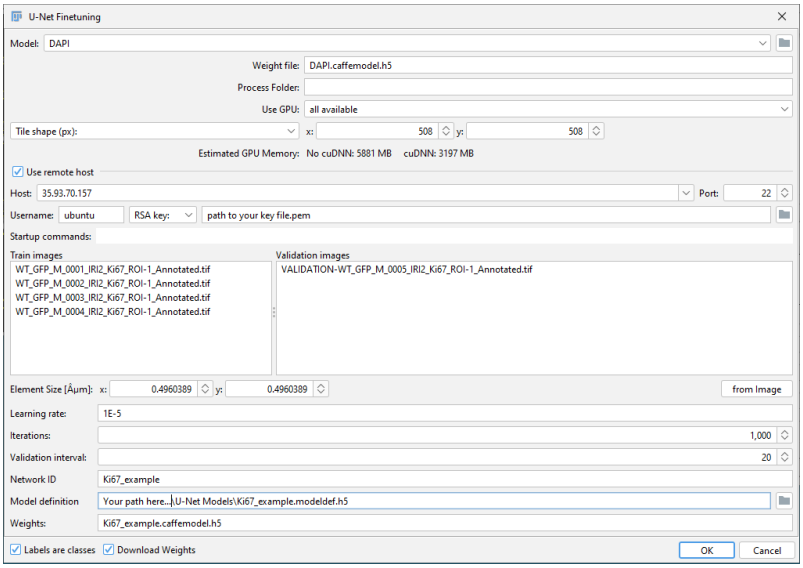


STEP SIX: UNET Training (FIJI)

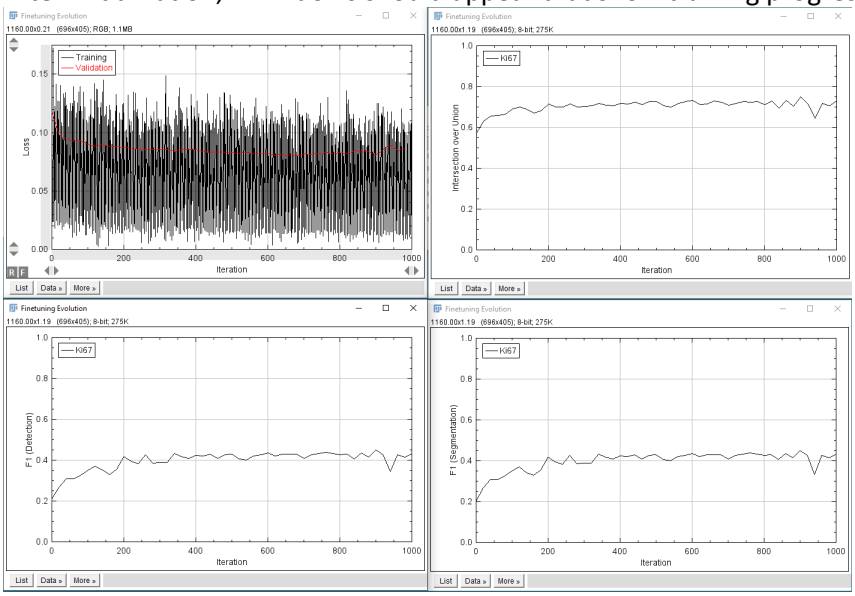
U-Net training

NOTE: instructions below are for an Amazon Web Services EC2 instance, but many other remote options could be configured depending on the expertise of the use. We do not endorse any particular product or service, only report what we have used for our analysis.

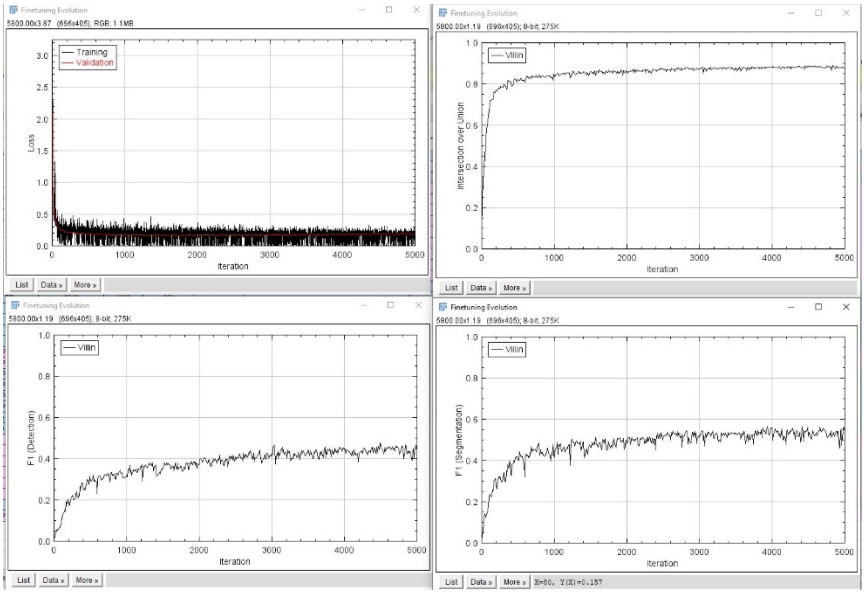
1) Start and configure an Amazon Web Services (AWS) EC2 instance (see "Setting up a remote AWS instance for U-Net Segmentation" protocol). You will need the "Public IPv4 address" and the "*.pem" version of the Key pair file you used when starting your instance.
2) Upload a weight file for a model similar to the target for trainings (for example, to train Ki67 as an example, model, we will use DAPI as an example).
3) Open all of the images in the "Annotated Images for Training" folder created and modified in **STEP FIVE** above
4) Open the U-Net plugin under Plugins→"U-Net"→"Job Manager"
5) Click on Finetuning
6) Under "Model:" Select a model that is similar to the target being trained. The weight file for the model you chose (…caffemodel.h5) needs to be on the EC2 instance already.
7) Under "Weight file:" Copy or type the name of the weight file (for example, DAPI.caffemodel.h5)
8) Under "Use GPU: " select "all available"
9) Select "Tile shape (px):" and enter x: 508 and y: 508 (could adjust if desired—but good default for most applications)
10) Check "Use remote host"
11) Under "Host:" enter the "Public IPv4 address" from the AWS instance. Port: 22
12) Under "Username:" enter "ubuntu" and select "RSA key:" and find the path to the "*.pem" Key pair file you used when starting your instance.
13) The images you opened should be listed under "Train images". Move the pre-labeled "VALIDATION" image from **STEP FIVE – 3** to the "Validation images" list.
14) IMPORTANT: set the element size by clicking "from Image" on the right. This is important as it will allow you to process images of different resolutions in the future. HOWEVER, for this to work the image MUST have the correct scale in all your images (see note in **STEP TWO – 4**)
15) Set "Learning rate:" to 1E-5 to start
16) Set "Iterations:" to 1000-5000 depending on how closely your starting model mimics your target. The greater the difference, the larger the number of iterations that will be needed.
17) Set "Validation interval:" to 20
18) Under "Network ID" type in the name of the new model. For demonstration purposes, we have simplified the model names, but in practice we generally append the target with the date the model was created (For example, "Ki67_01012022")
19) Adjust the "Model definition" path and file name to correspond to the new model by appending the name above with ".modeldef.h5". For example, "C:\folder1\folder2\U-Net Models\Ki67_01012022.modeldef.h5"
20) Adjust the "Weights:" file name. Do not include a path, just the file name. We append ".caffemodel.h5" to the name from 18 above. For example, "Ki67_01012022.caffemodel.h5"
21) Make sure "Labels are classes" and "Download Weights" are checked
22) It should look like this:



23) Click OK to start the training. (If this is the first connection of the day from FIJI to EC2, a security warning will pop up. Click OK)
24) The software will initialize, which will take a few minutes.
25) After initialization, 4 windows should appear that show training progress.



Note: in this example (using the data in the example data set) only a small set of training data with very high similarity to DAPI was used. As a result, this example converged very quickly within about 200 iterations. Below is an example of output with a full training set for Villin:



26) When simulation is complete, the new weight file (…caffemodel.h5) will be saved in the same directory specified in **STEP SIX -- 19** above.
27) Save training progress graphs for future reference. This can be saved within the "Training" Validation folders wherever it is accessible in the future.
28) Close everything

STEP SEVEN: Generate segmentations for qualitative assessment ("Analysis" Protocol, STEP SIX)

Segment test sections

1) *Segment all the images in the "**Original Images Full Size**" folder using the U-Net model and weights you just created. Refer to the "**Analysis**" protocol, STEP SIX for detailed instructions for performing batch segmentations.*
2) Assess the segmentation performance qualitatively using the "Overlays" files in the directory generated by batch segmentation.
3) If performance is inadequate, repeat **STEPS TWO -- SEVEN** by adding more images/annotations to the previously trained model to further fine-tune the performance.
4) If model performance is adequate, proceed to the **"Validation**" protocol