

Mar 23, 2022

# High time-resolution of root tracking of Arabidopsis roots using SPIRO, SmartRoot and BackTracker

Magdalena M Julkowska<sup>1</sup><sup>1</sup>Boyce Thompson Institute

1

[dx.doi.org/10.17504/protocols.io.81wgb6z5olpk/v1](https://dx.doi.org/10.17504/protocols.io.81wgb6z5olpk/v1)

Stress Architecture and RNA Biology Lab BTI

Tech. support phone: +1 (607) 279-6002 email: [mmj55@cornell.edu](mailto:mmj55@cornell.edu)

Magdalena Julkowska  
Boyce Thompson Institute

This protocol describes an alternative way of analyzing SPIRO images for root growth of plants that are transferred to the treatment plates - and thus not imaged from the germination onwards. This protocol combines SPIRO tools, SmartRoot and BackTracker script, as well as R-scripts for data curation. The protocol was thus far tested on Arabidopsis roots - and we intend to test it in the future on tomato roots as well.

DOI

[dx.doi.org/10.17504/protocols.io.81wgb6z5olpk/v1](https://dx.doi.org/10.17504/protocols.io.81wgb6z5olpk/v1)

Magdalena M Julkowska 2022. High time-resolution of root tracking of Arabidopsis roots using SPIRO, SmartRoot and BackTracker. **protocols.io**  
<https://dx.doi.org/10.17504/protocols.io.81wgb6z5olpk/v1>



root growth, high time resolution, Arabidopsis, SPIRO, SmartRoot, BackTracer

\_\_\_\_\_ protocol ,

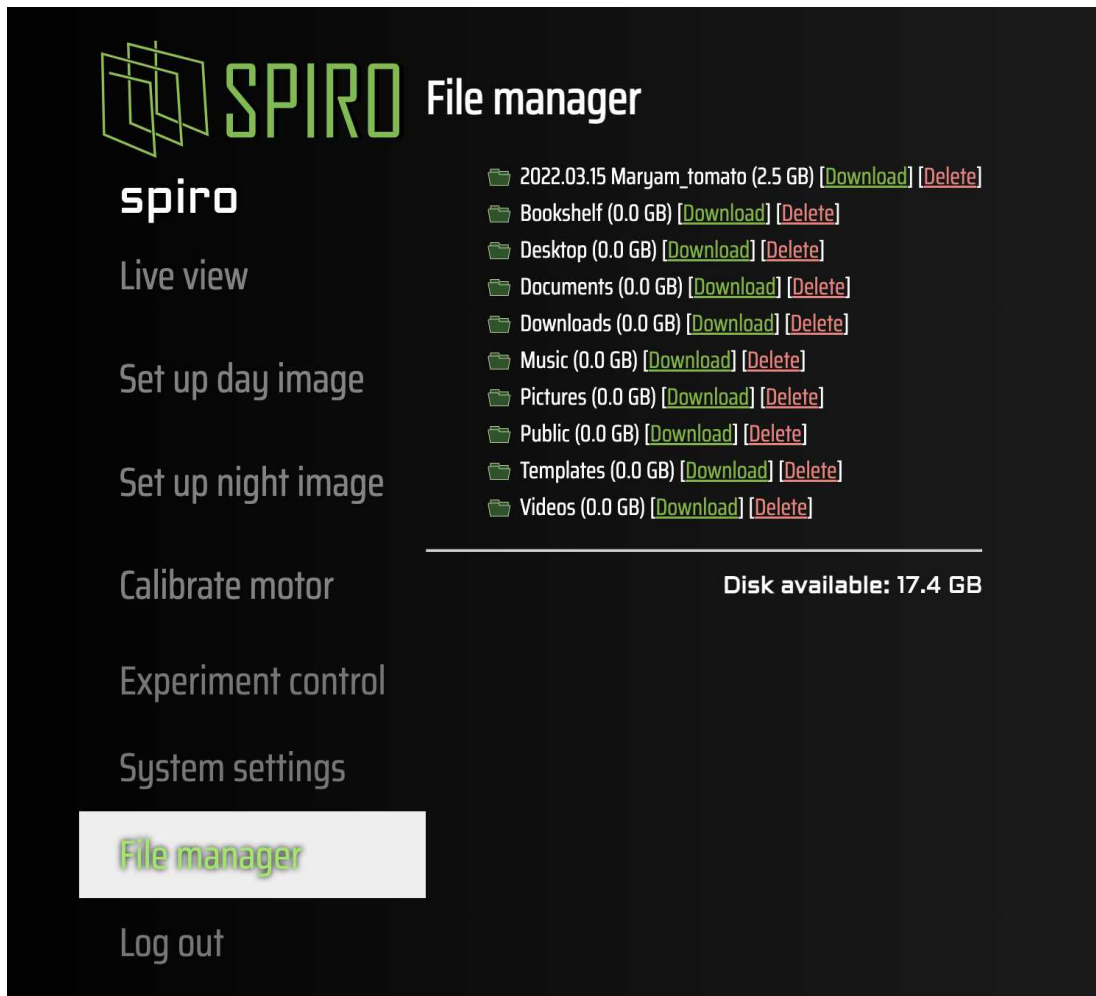
Mar 18, 2022

Mar 23, 2022

59609

- 1 Collect your images captured by SPIRO from the Raspberry Pi by going into the FileManager -

and downloading your experiment:



- 2 Pre-process your images for drifting using the SPIRO pre-process in ImageJ / Fiji - as described on SPIRO Assay github page [here](#). This will result in all of your images being cropped and ALIGNED, and also transferred as multiple layers into one TIFF file.

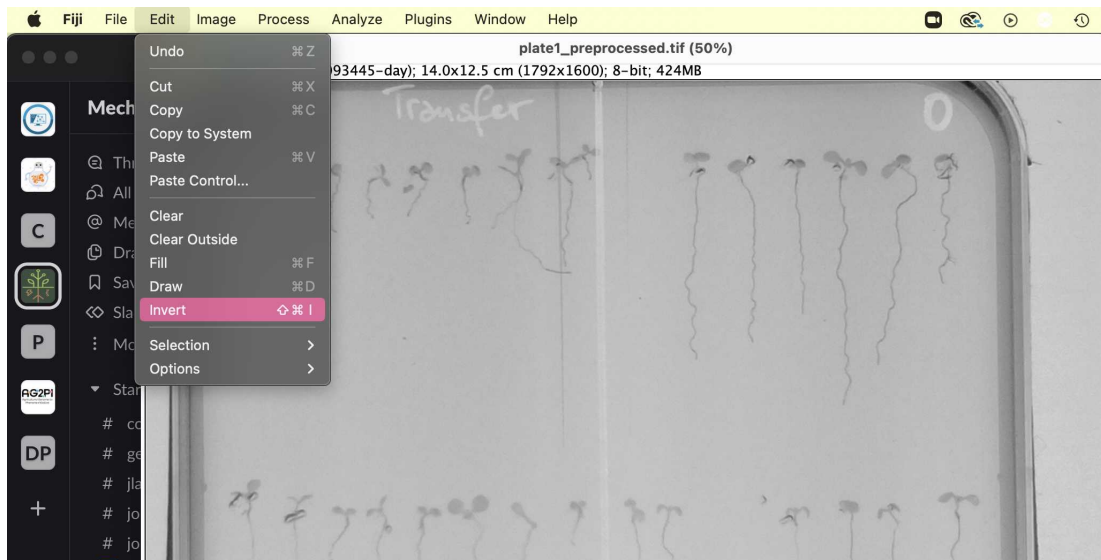


- 3 Now we need to transfer this multi-layer tiff file into individual tif files - preferably containing their timestamp for our future steps.

At this moment - you have two options

- 1) rather painful (manual) solution that will maintain the information and timestamps in the file name OR
- 2) fully automated one - but your images will be renamed into 1, 2, 3, 4 ... and you will lose a timestamp information - if you are confident you will be able to add it back in later in R - go for it - but just beware that this might be a source of potential problems in the future.

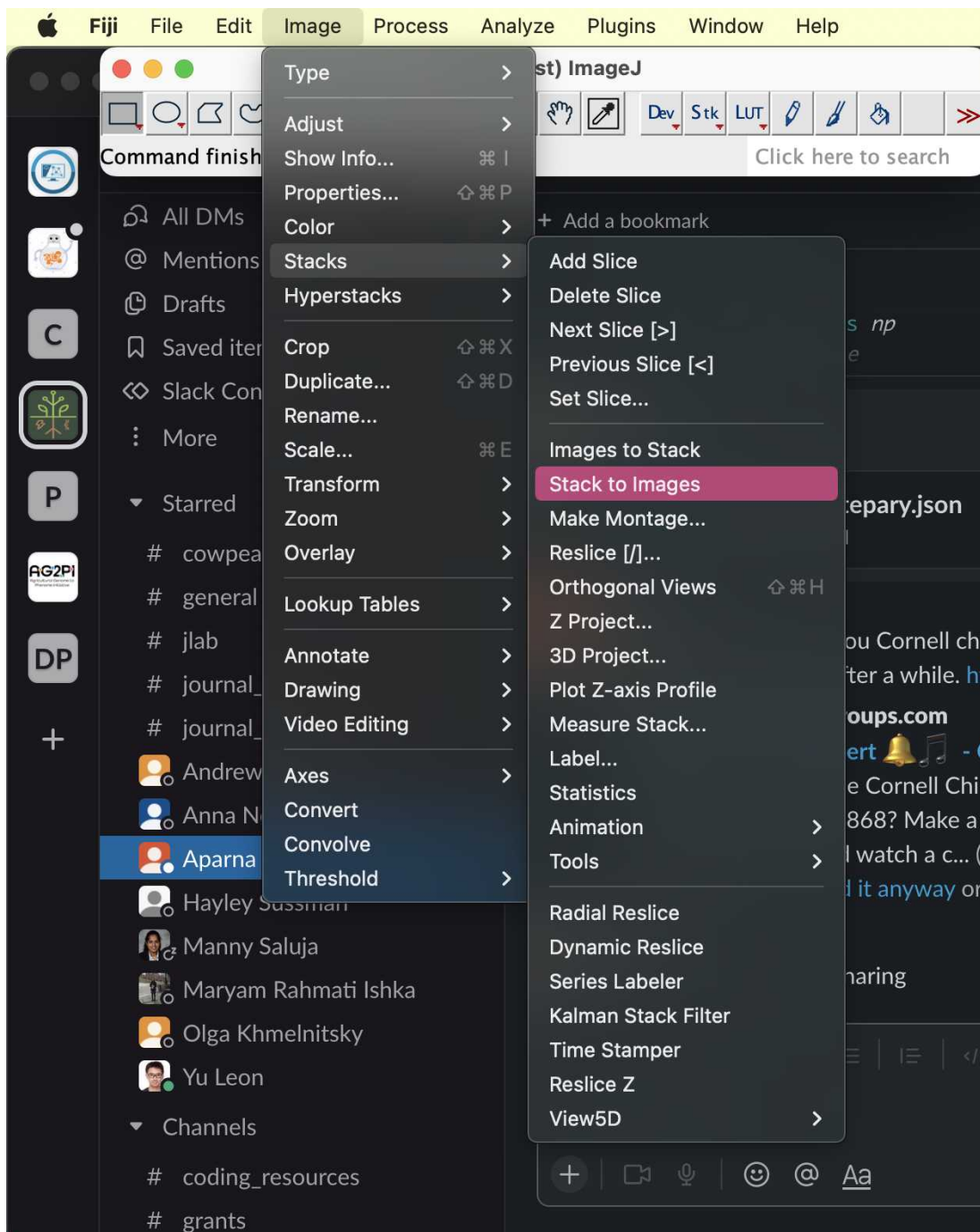
BUT FIRST - For both options - you need to first invert the image - since SmartRoot prefers the root to be DARKER than the background. Open your multi-layer tiff in ImageJ/Fiji and select invert function - save the inverted file in your Experiment folder:



now - you can choose whatever option you feel most comfortable with:

### OPTION 1: manual splitting image using ImageJ/Fiji

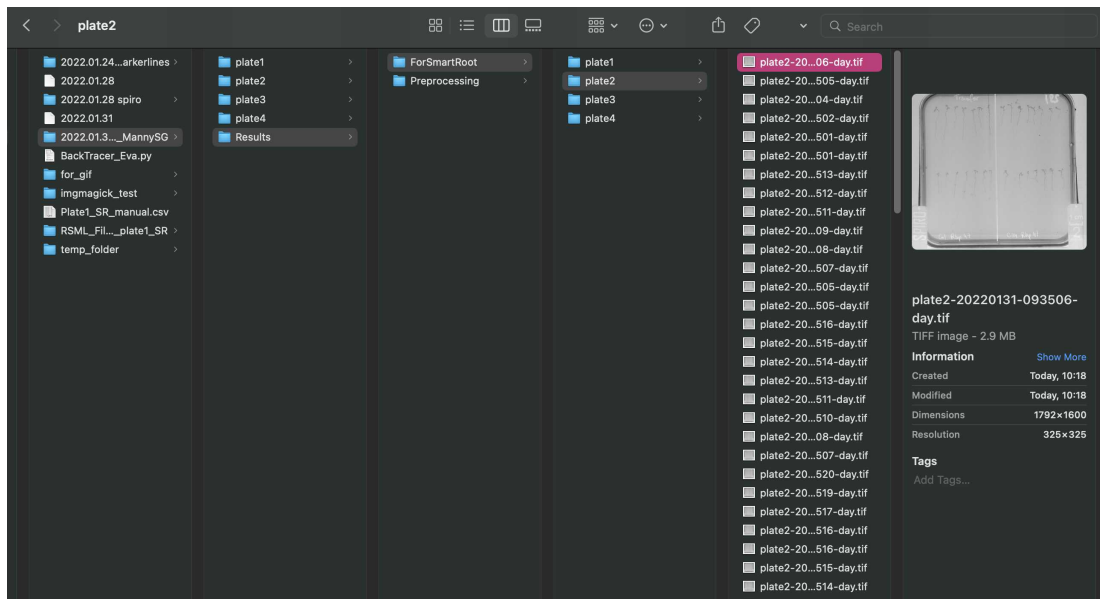
Open the multi-layer-tiff file in Fiji/ImageJ - and select "Stack to Images" like on the screenshot below:



It will open ALL layers as individual images and then save individual images through SAVE KEYBOARD SHORTCUT (Mac = cmd + S; Windows = ctrl + S). It will feel like it takes forever - but its actually not that bad. And also - I really hope to have a more automated solution for this soon.

NOTE - make sure to save individual files in ONE FOLDER per plate - similar to the original SPIRO setup.

As a reference - here is how my folder looks like before I start my SmartRoot part after this option:



## OPTION 2: automatic split of the images using ImgMagick

1) Make sure to install [ImgMagick](https://www.imagemagick.org/) on your computer

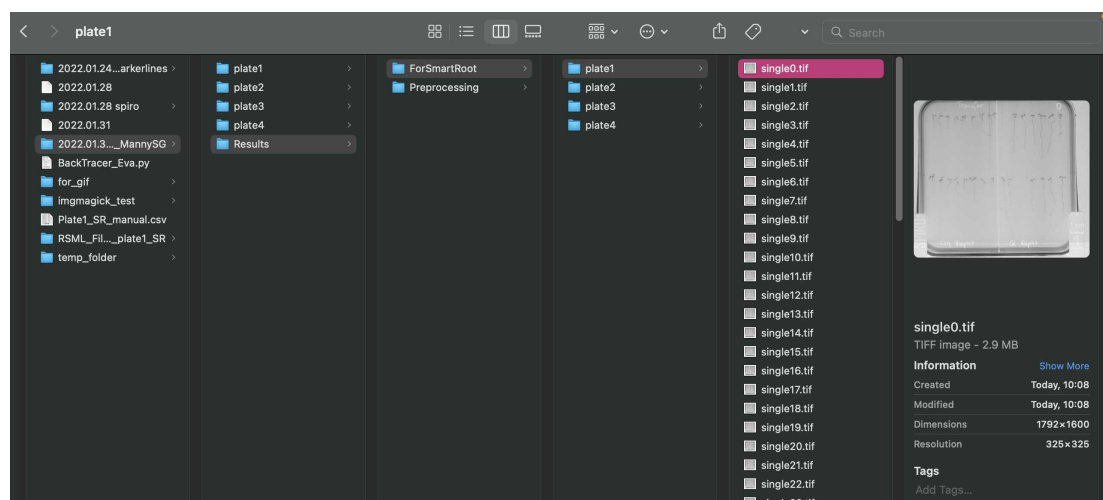
2) Navigate to your folder containing the multi-layered tiff file

3) run the following command - adjusting the file name:

```
convert plate1_preprocessed.tif single%d.tif
```

4) put all single layer tiff file into one folder per plate - and maybe move it into your SmartRoot Experimental directory.

As a reference - here is how my folder looks like before I start my SmartRoot part after this option:





- 4 Trace the LAST file of each plate - containing the longest roots - using [SmartRoot](#) plugin for ImageJ / Fiji.

Do NOT put any effort in naming the roots - they will be renamed anyways afterwards - so save your time. We will figure it out what genotype is where later.

I advise NOT to trace the roots that are entangled within each other in the last image and/or end in a shoot - they need to be all clearly distinguished roots - not growing into the other root.

Here is a screenshot of the roots that I did/did not trace on my plate:



- 5 Make sure to install all the dependencies on your computer for BackTracer:

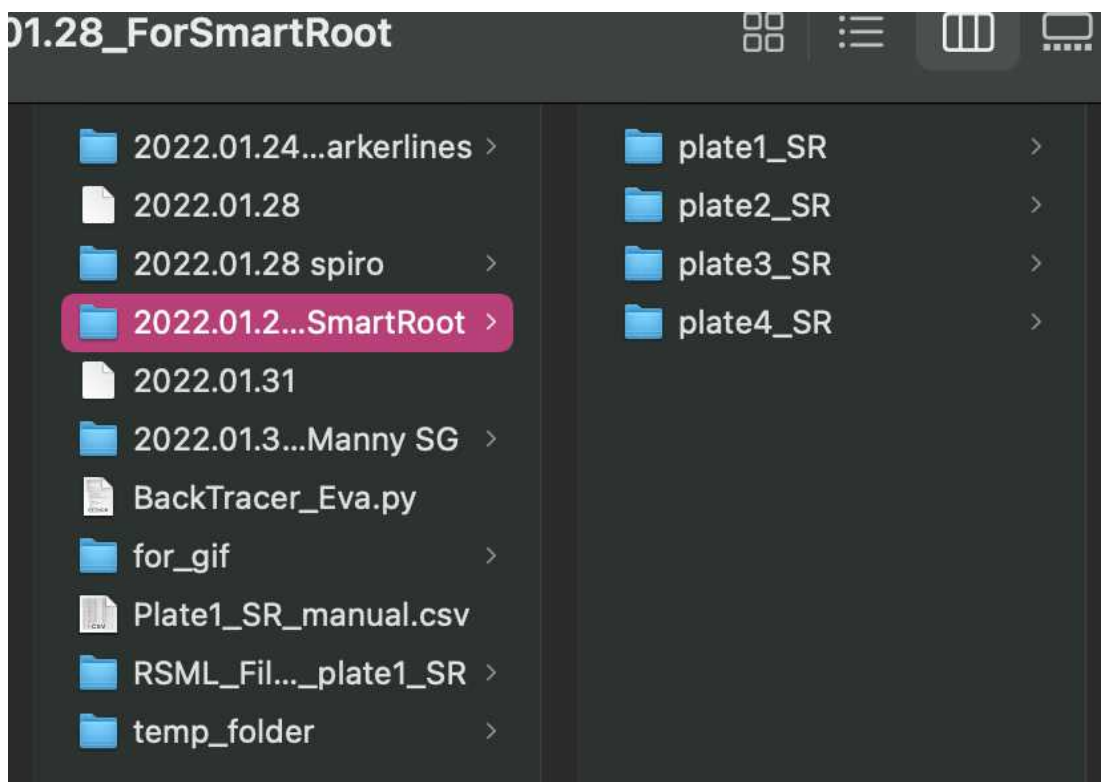
```
scipy  
opencv-python  
numpy
```

```
BeautifulSoup
re
XML
shutil
```

- 6 Once you have everything installed - download the python script from the [BackTracker](#) repository, and open it in a code-writing software like [Atom.io](#).

In the python code - you will need to change TWO things:

1) adjust the image repository to the folder containing the files with your plate images (and SmartRoot traced the last image - you will get rsml file once the image is traced).

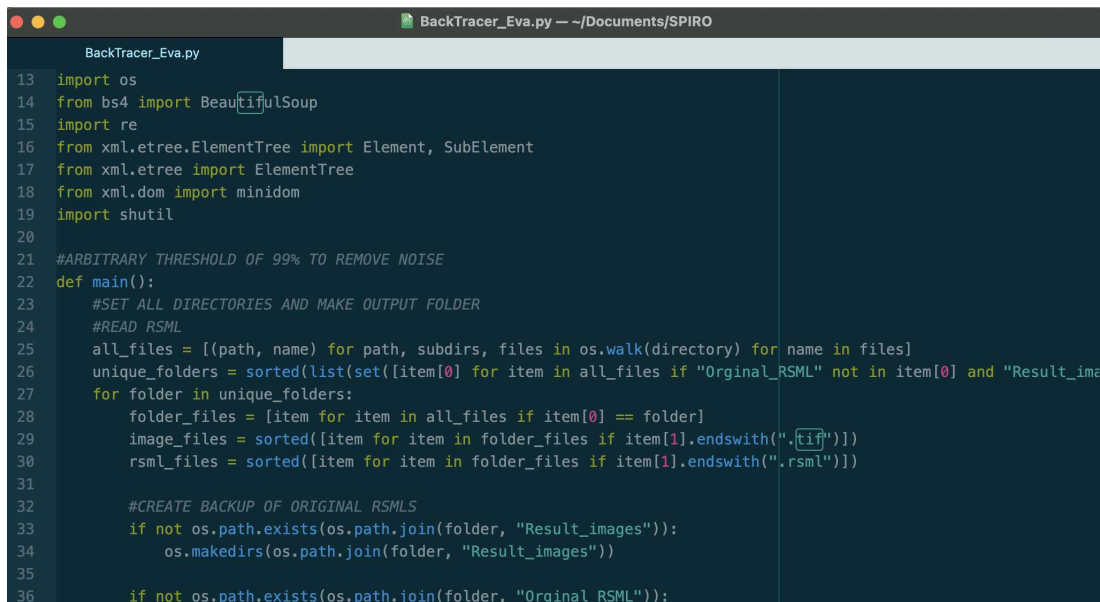


High-lighted repository is my "experiment repository"

2) Make sure to replace all "tiff" by "tif" - since the BackTracker is using "tif" in their original file format.

I only found one location where it needs to be changed (line 29) - but double check if it isn't popping anywhere else:





```

BackTracer_Eva.py
13 import os
14 from bs4 import BeautifulSoup
15 import re
16 from xml.etree.ElementTree import Element, SubElement
17 from xml.etree import ElementTree
18 from xml.dom import minidom
19 import shutil
20
21 #ARBITRARY THRESHOLD OF 99% TO REMOVE NOISE
22 def main():
23     #SET ALL DIRECTORIES AND MAKE OUTPUT FOLDER
24     #READ RSML
25     all_files = [(path, name) for path, subdirs, files in os.walk(directory) for name in files]
26     unique_folders = sorted(list(set([item[0] for item in all_files if "Original_RSML" not in item[0] and "Result_images" in item[0]])))
27     for folder in unique_folders:
28         folder_files = [item for item in all_files if item[0] == folder]
29         image_files = sorted([item for item in folder_files if item[1].endswith(".tif")])
30         rsml_files = sorted([item for item in folder_files if item[1].endswith(".rsml")])
31
32     #CREATE BACKUP OF ORIGINAL RSMLS
33     if not os.path.exists(os.path.join(folder, "Result_images")):
34         os.makedirs(os.path.join(folder, "Result_images"))
35
36     if not os.path.exists(os.path.join(folder, "Original_RSML")):

```

You can find this file - where I already adjusted all the "tiff" to "tif" and you just need to change your directory in line 8 of this script:

 [BackTracer\\_Eva.py](#)

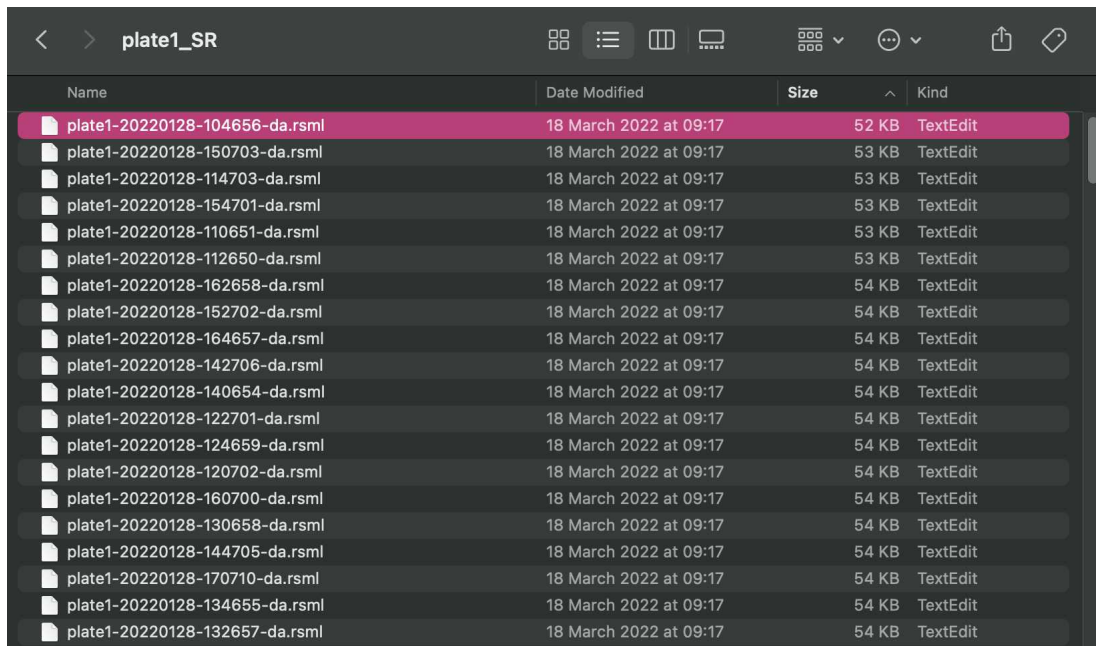
- 7 Make sure to save changes to the file - and run the python file by typing in your terminal / Conda window:

```
python path/to/your/python/code BackTracer.py
```

Running this step might take a while, so be patient, go make yourself coffee / tea or stretch your legs and arms.

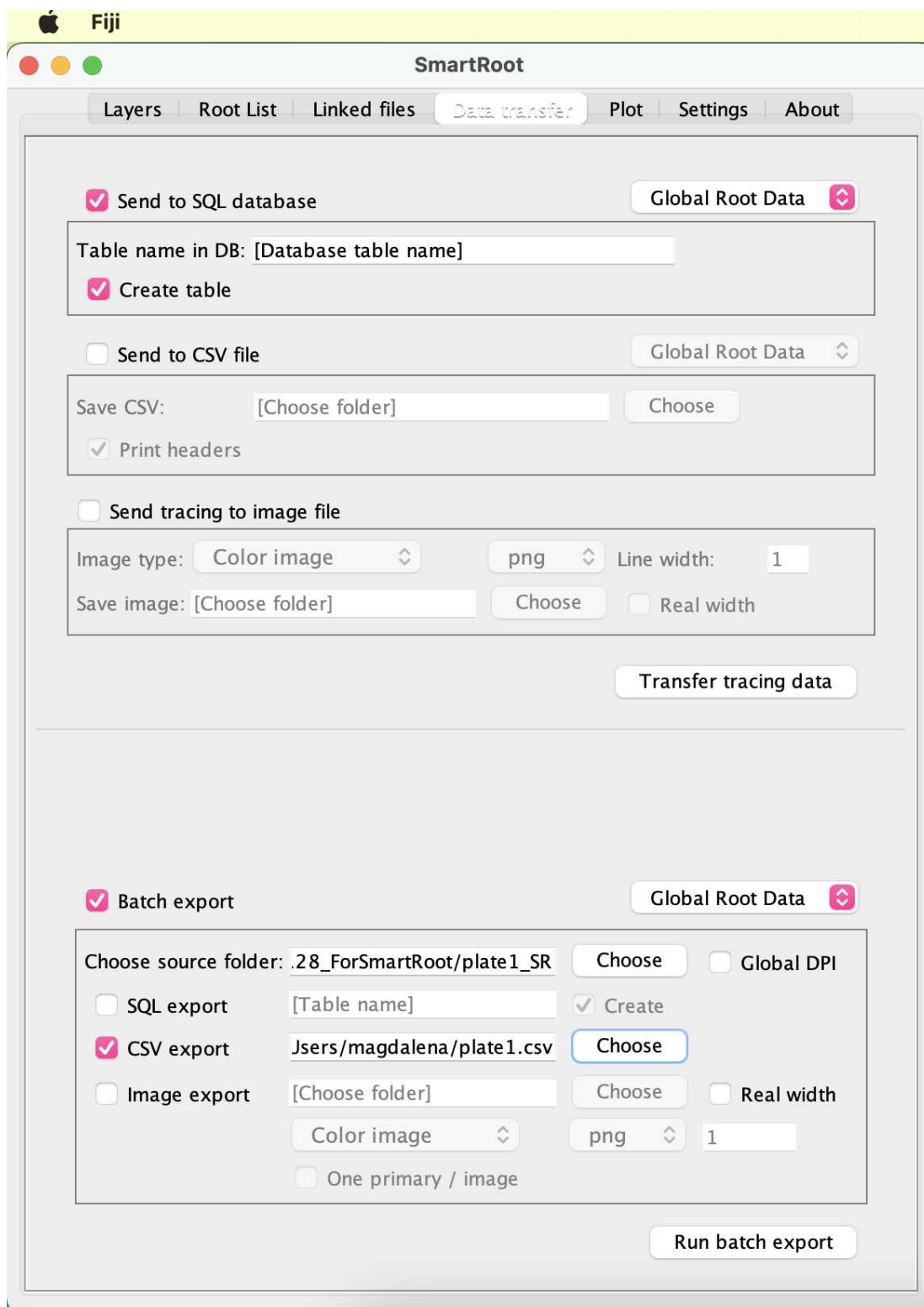
- 8 Make a Quality Control - by going into your original Experiment folder - and figuring out if you have any EMPTY RSML files. You can easily see it by the size of the rsml files. As long as the files are NOT empty - it will be fine.

You can easily see the size of the files by changing the view in your Finder window:



Name	Date Modified	Size	Kind
plate1-20220128-104656-da.rsml	18 March 2022 at 09:17	52 KB	TextEdit
plate1-20220128-150703-da.rsml	18 March 2022 at 09:17	53 KB	TextEdit
plate1-20220128-114703-da.rsml	18 March 2022 at 09:17	53 KB	TextEdit
plate1-20220128-154701-da.rsml	18 March 2022 at 09:17	53 KB	TextEdit
plate1-20220128-110651-da.rsml	18 March 2022 at 09:17	53 KB	TextEdit
plate1-20220128-112650-da.rsml	18 March 2022 at 09:17	53 KB	TextEdit
plate1-20220128-162658-da.rsml	18 March 2022 at 09:17	54 KB	TextEdit
plate1-20220128-152702-da.rsml	18 March 2022 at 09:17	54 KB	TextEdit
plate1-20220128-164657-da.rsml	18 March 2022 at 09:17	54 KB	TextEdit
plate1-20220128-142706-da.rsml	18 March 2022 at 09:17	54 KB	TextEdit
plate1-20220128-140654-da.rsml	18 March 2022 at 09:17	54 KB	TextEdit
plate1-20220128-122701-da.rsml	18 March 2022 at 09:17	54 KB	TextEdit
plate1-20220128-124659-da.rsml	18 March 2022 at 09:17	54 KB	TextEdit
plate1-20220128-120702-da.rsml	18 March 2022 at 09:17	54 KB	TextEdit
plate1-20220128-160700-da.rsml	18 March 2022 at 09:17	54 KB	TextEdit
plate1-20220128-130658-da.rsml	18 March 2022 at 09:17	54 KB	TextEdit
plate1-20220128-144705-da.rsml	18 March 2022 at 09:17	54 KB	TextEdit
plate1-20220128-170710-da.rsml	18 March 2022 at 09:17	54 KB	TextEdit
plate1-20220128-134655-da.rsml	18 March 2022 at 09:17	54 KB	TextEdit
plate1-20220128-132657-da.rsml	18 March 2022 at 09:17	54 KB	TextEdit

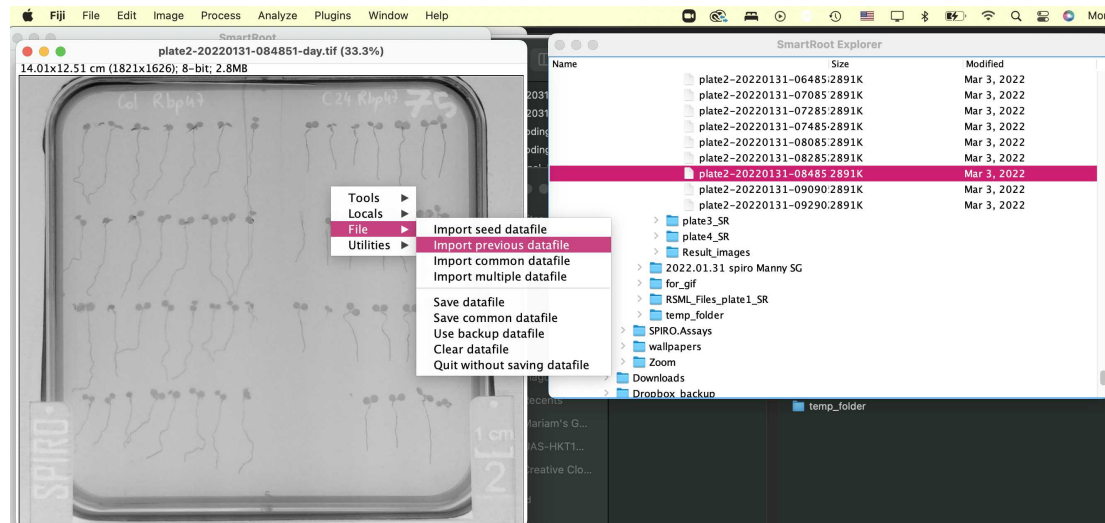
- Export your rsml files as csv file one plate as the time through SmartRoot Batch export window:



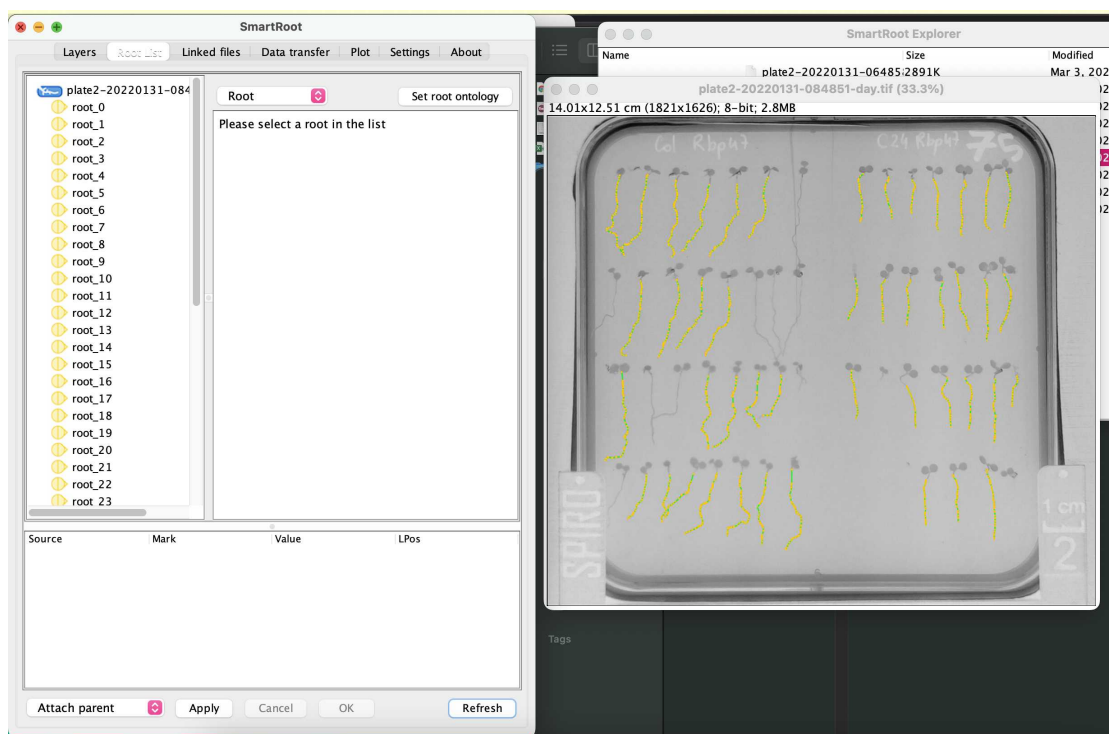
10 The last thing that we need to do in SmartRoot is to figure out what root ID corresponds to your individual genotypes.

1) Open the penultimate image in your image stack and add tracing data by the BackTracer by selecting with the right click "Import previous datafile" - and select the file that was generated

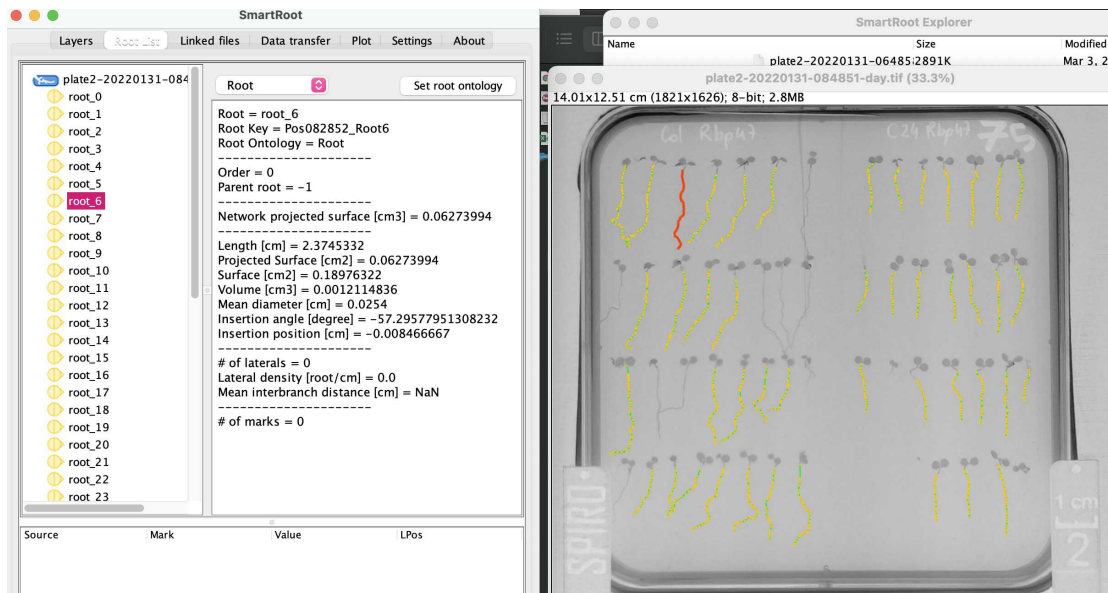
by the BackTracker.



2) Once you imported the automatically traced RSML file - go to RootList on the SmartRoot menu and click "Refresh" at the bottom of the root list menu - the root names will appear automatically:



3) By selecting individual roots - you will be able to tell what root is belonging to what genotype



4) make a decoding file - containing the root name as in the rsml file as well as its actual genotype and/or condition - I attach my own decoding file below as an example:

 [decoding\\_roots.csv](#)

11 Import the data into R and for analysis, you can follow the R-pipeline developed by Magda [here](#).

NOTE: There is a big chance there will be MANY outliers / odd time points - so you will spend a significant amount of time in R analysis on data curation.

I have a number of data curation methods described in my pipeline - please use whichever you think will apply the best for your specific scenario and the precision of this automatic tracing.