Quick N' Dirty

Ilastik Workflow Manual

Segmentation And Measurement Of Leaf Coloration Area

Version 0.1 01/30/18

Marc Mathias Clausen

Purpose

Segmenting and measuring the areas leaf coloration (Red, Green, Yellow) and putting the area values in a spreadsheet format.

Preset Conditions

System minimum recommended specifications:

For preparing for bulk processing:
4 cores, 16 GB ram (no advanced graphics processor needed)

For bulk processing:
Anything higher than above

Operating system (tested with):

centos/oracle/red hat 7.4

Programs (tested with):

Ilastik 1.2.2

ImageMagick.x86_64 0:6.7.8.9-15.el7_2

(May require installation of dependencies: ImageMagick and libXScrnsaver)

Note on Images:
To get the most consistent results, it is important the images have as few variations as possible, other than the leaf areas we want to measure. A evenly colored background with a strong contrast to the area of measurement helps in segmenting the leaves from background accurately.

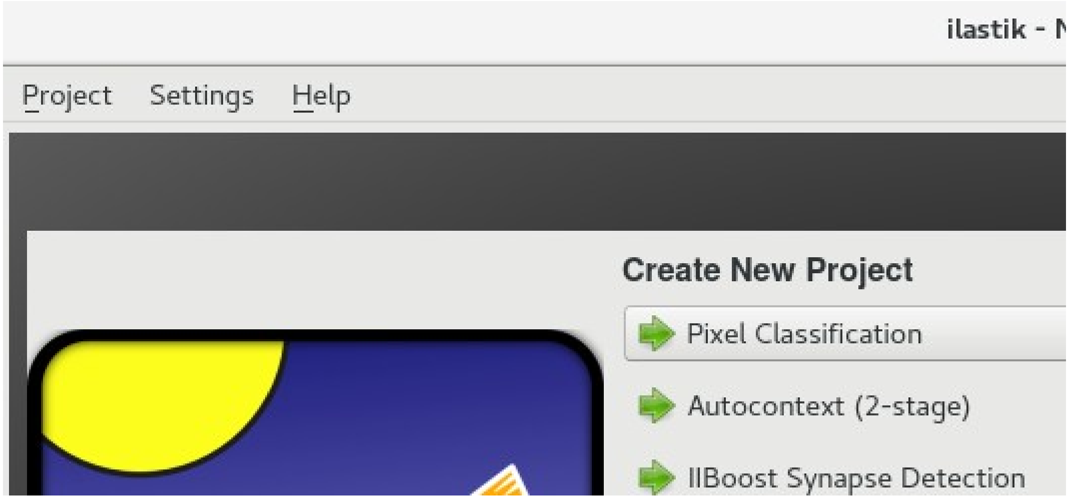
Installation/Preparation

Get ilastik from their website.
Unpack the archive
Open a terminal and run it like ./run_ilastik.sh

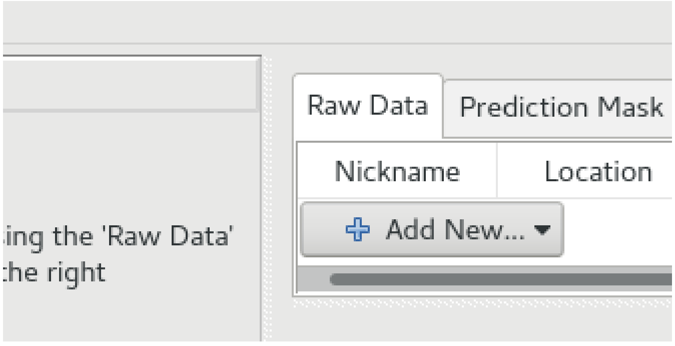
Pixel classification

Goal: Tell Ilastik which pixels (color signatures) we are interested in.

From the start page of ilastik:



1. Create New Project -> select Pixel Classification
2. Choose where to save your project file
3. In the tab 'Raw Data' -> click 'Add new...' -> choose seperate images

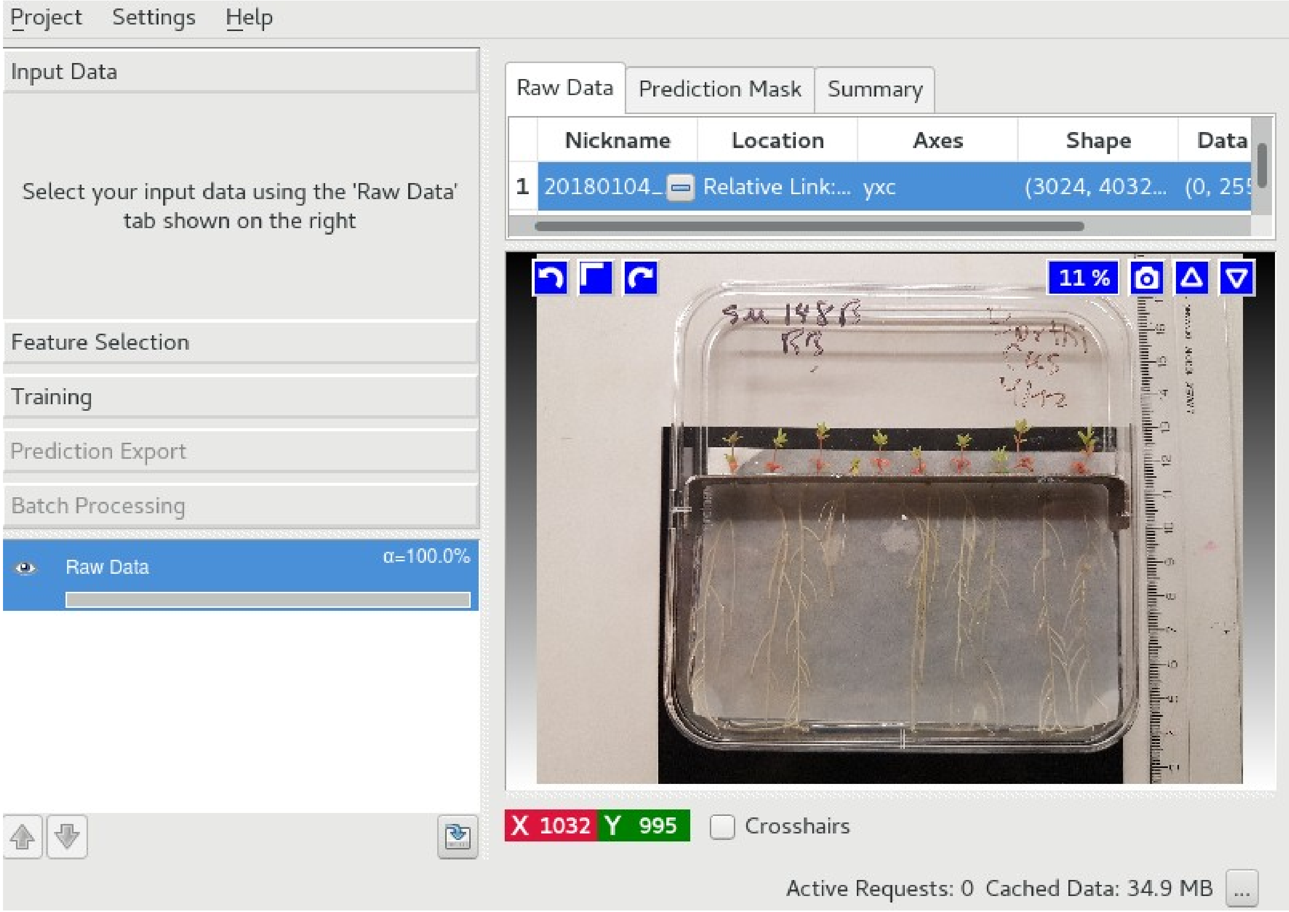


The images you want to select are those which generally represent your set of images, so get a wide selection of images, for example spread over time the picture is taken. Example shown below. Red would be your selected images:

Monday Tuesday Wednesday Thursday Friday Saturday Sunday

Note on performance: Avoid importing your whole set of images (it will use all of your RAM. I recommend importing 3-10 images for training the program.)

4. Now your images for training the program are imported.



Select your input data using the 'Raw Data' tab shown on the right

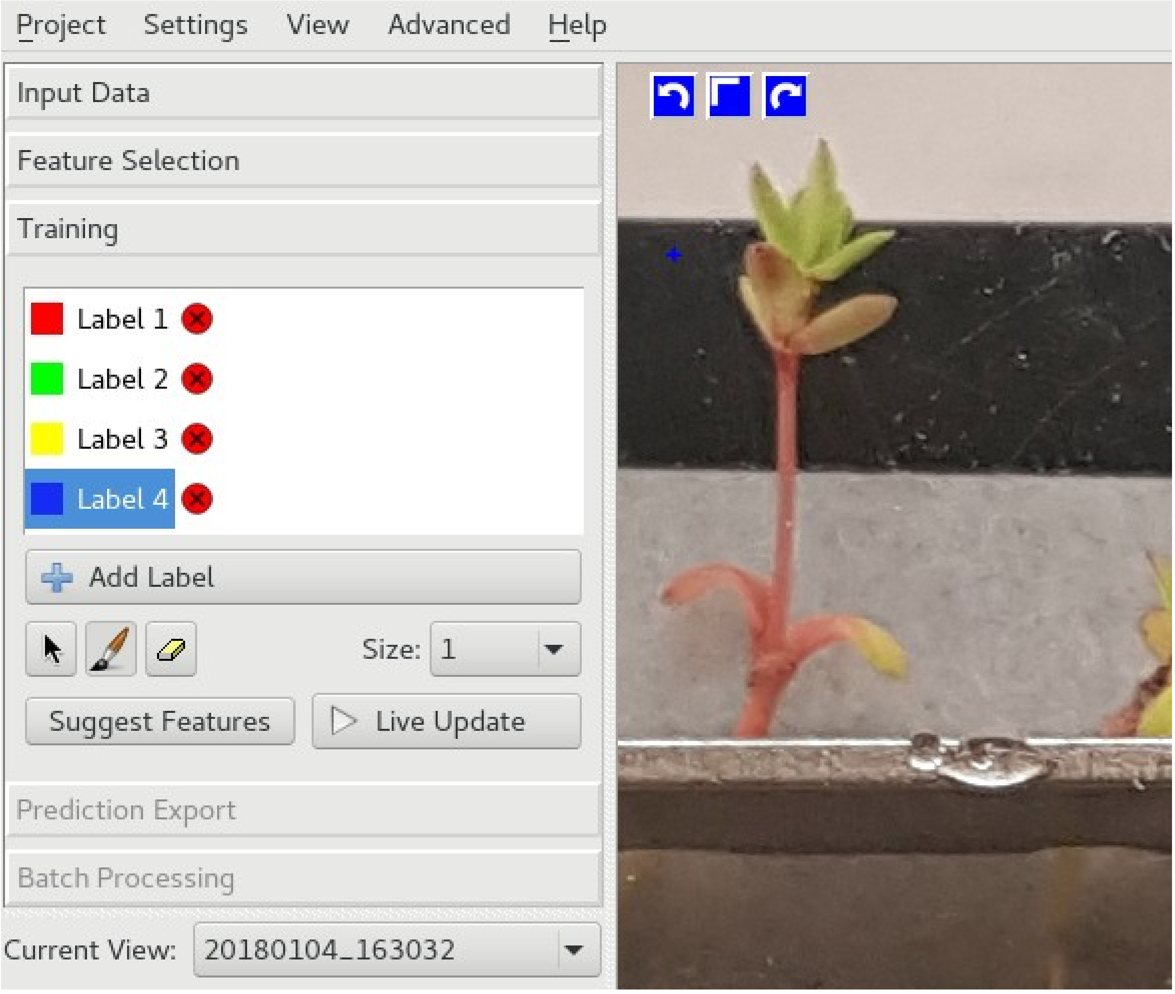
	Nickname	Location	Axes	Shape	Data
1	20180104_	Relative Link:...	yxc	(3024, 4032...	(0, 25

X 1032 Y 995

Active Requests: 0 Cached Data: 34.9 MB

5. Now click 'Training'

6. Add 4 labels. I will use the Red Green and Yellow to represent the leaf color areas I want to measure. Blue is for everything background.

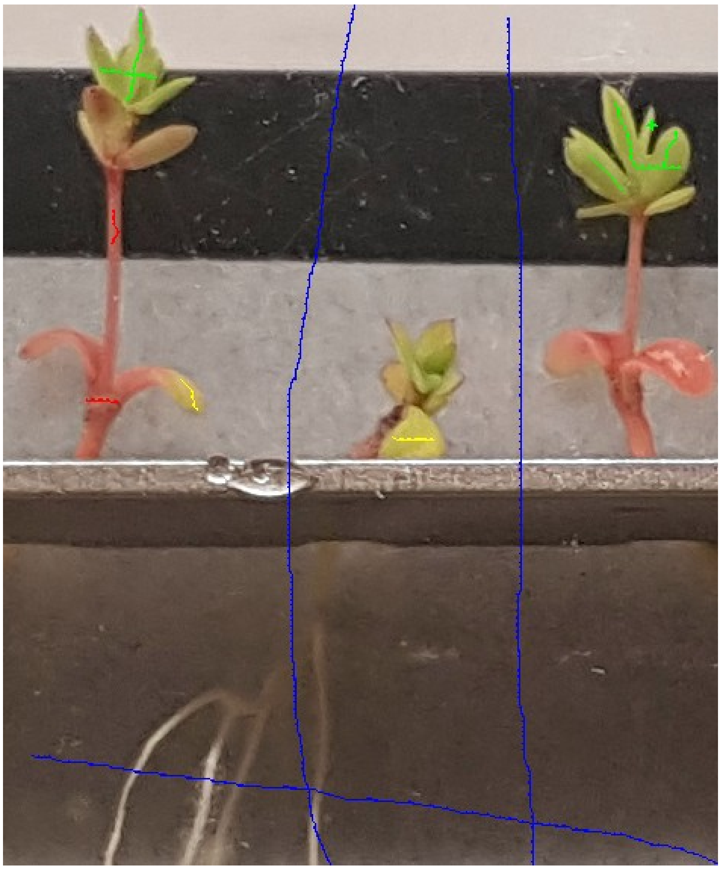


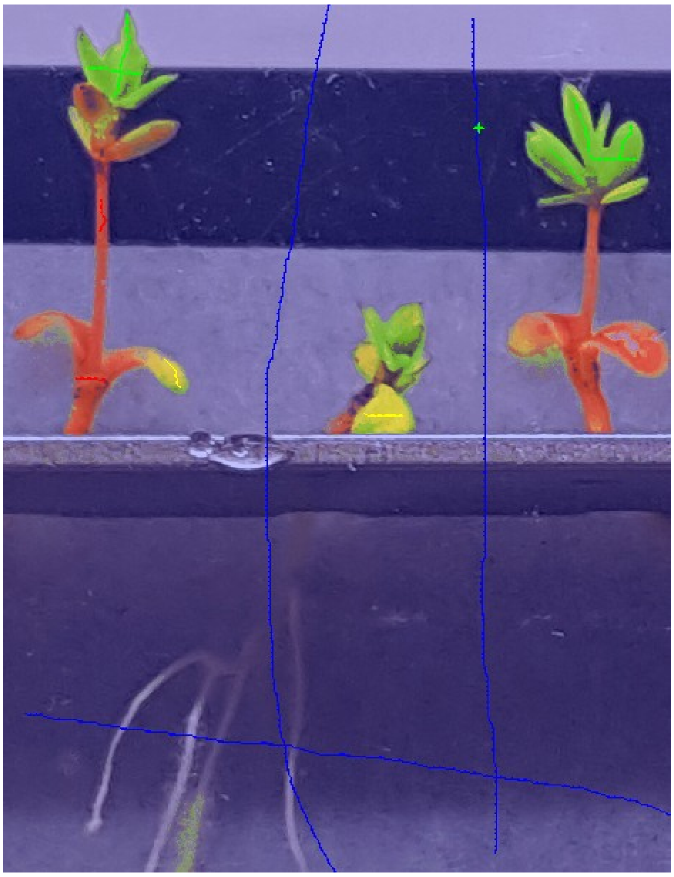
7. Select a label and draw on it's corresponding pixeltype (color you wish to label, eg. Green leaf area). You can draw on the right side view.

Note on training: Be very specific when you apply labels (drawing lines). Do not draw with the green label where there is yellow leaf, background, or red leaf. Etc.. - Failing to avoid this may result in unwanted noise.

8. Click 'Live Update' , found under your list of labels

Below is an example of labels drawn and on the right, the resulting live update view. The Live Update view will show you how the program is interpreting your labelling.

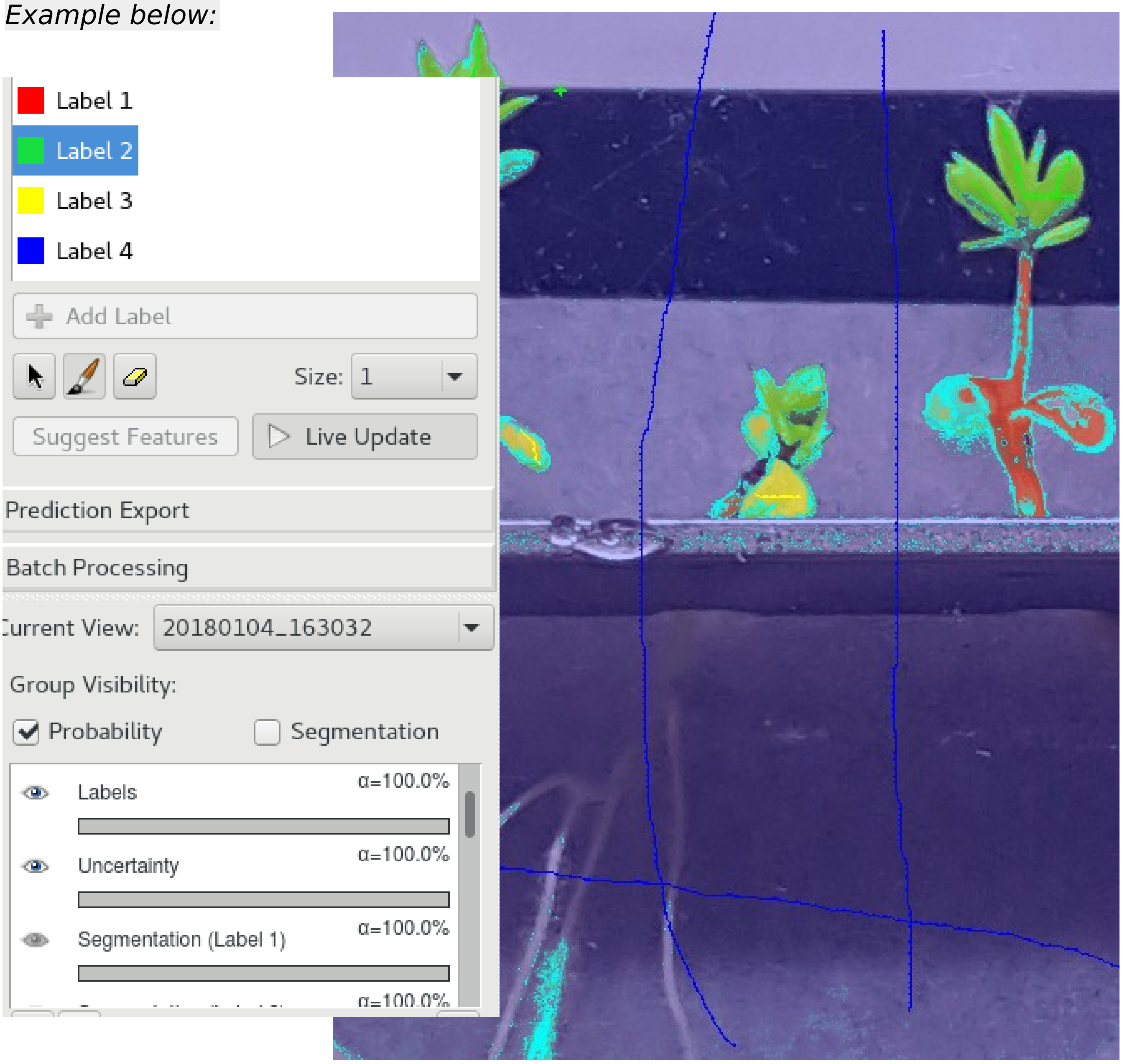




As you can see on the picture to the right, there is green detected on the roots. This you will need to correct by drawing Blue (what we want to be background). By doing so you are telling the program that you label this area as 'Blue'.

Note on uncertainties: *Click the 'eye' next to Uncertainty to show areas where the program is unsure what label to classify the pixels as. If these uncertain areas are belonging to a label, training these areas to the correct label will benefit the overall result a lot.*

Example below:



Size: 1

Current View: 20180104_163032

☑ Probability ☐ Segmentation

9. When you are happy with the result. Save the project file (It's a .ilp file), for use in the next steps.

Run Batch Prediction / Processing Images

Now it's time to add the magic source

You run a bash script I've created to get the areas from your labels, using also the saved project file from before.

Run the *ilastik_pixel_segment_area.sh*

These are notes from the script file itself describing how to write the correct arguments:

# 1 : absolute path to run_ilastik.sh
# 2 : Name or absolute path to project file (.ilp)
# 3 : Path to folder containing ONLY the input images (ending with a /)
# 4 : number of labels used

Example of how to run the script on the terminal:

```
sh ilastik_pixel_segment_area.sh '/home/user/ilastik-1.2.2post2-Linux/run_ilastik.sh' MyProject.ilp ./inputfiles/ 4
```

And voila! If the command runs without errors, You should have a nice .txt file waiting for you in the timestamped result folder, along with the images created as a result of the prediction.

Notes on input images:
Tested only with .jpg files as input

Appendix X: Entire code of the script:

```
#!/bin/bash
# Marc Mathias Clausen 2018
#
# ARGUMENTS..
# 1 : absolute path to run_ilastik.sh
# 2 : Name or absolute path to project file (.ilp)
# 3 : Path to folder containing input images (ends like / )
# 4 : number of labels used

DATE=$(date +%Y%m%d_%H%M%S)
absoluteilastik=$1
inputfiles=""$3""*
project=$2
outputdir=./""$DATE""_ilastik/
output="$outputdir"{nickname}_results.jpeg
labels=$4
echo $DATE

sh $absoluteilastik --headless         \
                    --export_source="Simple Segmentation"                      \
                    --project=$project                                         \
                    --export_dtype=uint8                                       \
                --output_format=jpeg                                           \
                --output_filename_format=$output                               \
                    --pipeline_result_drange="(0.0, ""$labels"".0)"            \
                    --export_drange="(0, 255)"                                 \
                $inputfiles

FILES=""$outputdir""*.jpeg
for f in $FILES
do
  echo "Measuring $f ..."
  echo "Saving Data from $f ..."
  convert $f +dither -colors $labels -define histogram:unique-colors=true -format "%f\n%c\n" histogram:info:\
| sed -e 's/^[[:space:]]*//' | cut -f1 -d":" | paste -s >> ""$outputdir""""$DATE""_nice_data.txt
done

# The output will be shown as top label from ilastik being the
# value of the first column, second from the top is second column etc.
# the white color of the output directory images corresponds to the bottom label in ilastik
```