Introduction

To locate the wells, we create a 12 by 8 matrix containing the coordinates of each well, equally spaced.

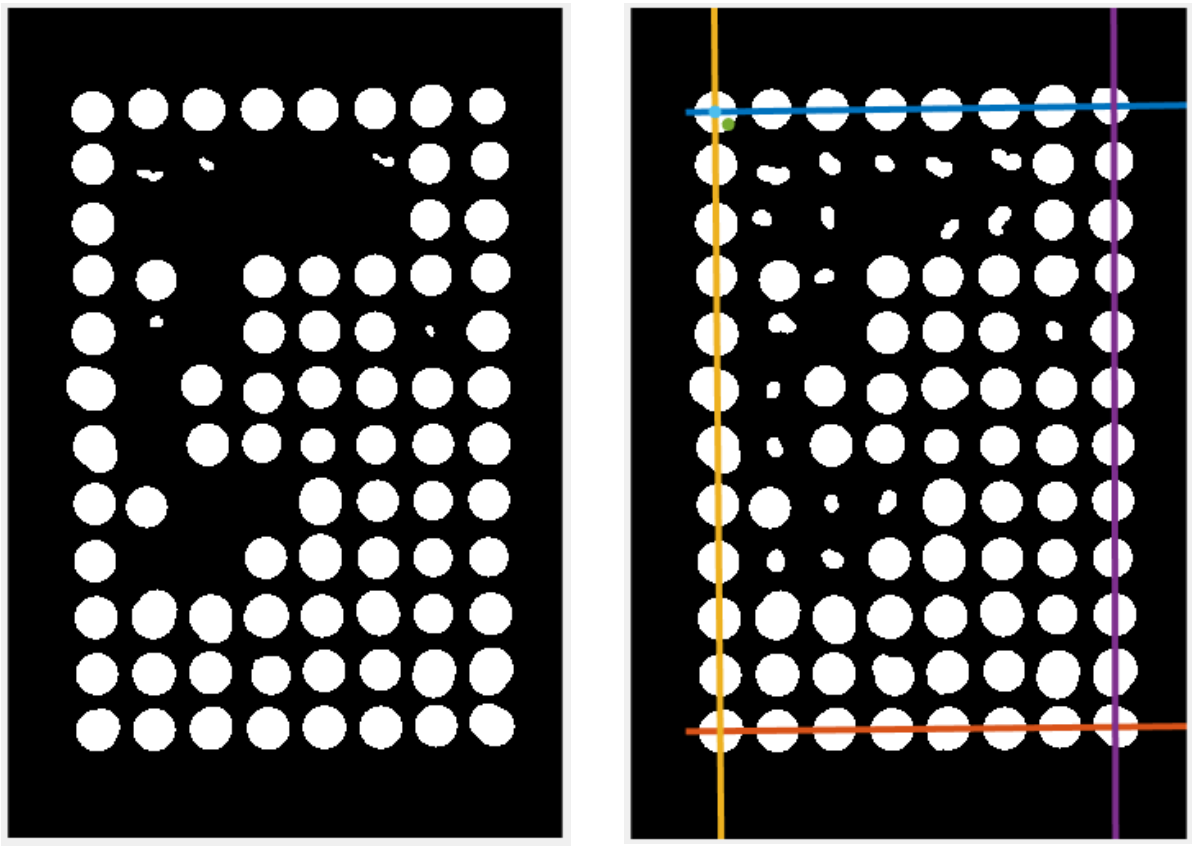
We use the locations of the duckweeds to aid us in locating the wells with precision. We use the following relationship to find the green intensity of each pixel:

$$GI = \frac{255 + G - \frac{R + B}{2}}{2}$$

green threshold -> find location of well (if they have duckweed) and then define inclination using linear interpolation

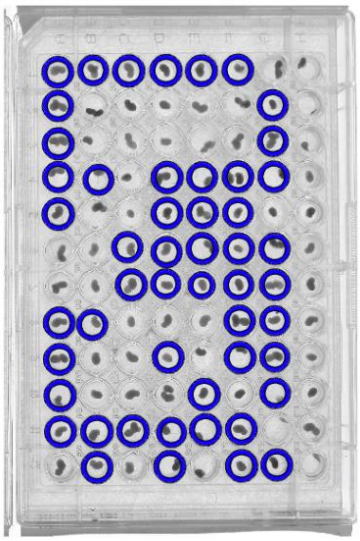
Using the approach, we can roughly locate the duckweeds. A binary image of the well plate is generated. Points that are too close to the border of the image are eliminated.

The location of a number of wells is determined firstly using the circular Hough transform. For our images, this approach can only identify a portion of the wells. We use this and the approach described previously to generate a binary image.



We fit lines of best fit for all the wells in the extremities of the plate (vertically a1 to a12, h1 to h12, and horizontally from a1 to h1 and a12 to h12) to find the tilt and to locate the wells

We then compare the distance between the centers of the identified circles with those of the location matrix. Since only a portion of all the wells are identified, we use this approach to move the location matrix by minimizing the difference between the two.



can find some wells, but not all

Once the wells are properly located, we crop the well plate image into the individual well images, where we conduct in-depth image segmentation to characterize the morphology of duckweed.

Image segmentation

The locations of each well are saved, and individual well images are cropped. Next, we implement an algorithm to identify the duckweed in each individual image using color threshold settings, as well as the green intensity GI.

1) We first extract all pixels whereby $GI \geq GT$, where GT is a pre-defined green threshold.
2) We then use the HSV color threshold values on the same image to extract the pixels that fall between the specified values
3) The pixels that have been selected at either steps 1, 2 or both are both chosen and a binary mask is created, where pixels that are part of the duckweed area are assigned a value of 1, and pixels that are not are assigned a value of 0.
4) Binary mask is filled to remove any small holes in the binary image. Larger holes (i.e. when there is empty space in the image, but which is completely surrounded by duckweed) are eliminated from the binary image based on a hole threshold value.
5) Binary image is smoothed via convolution to remove sharp edges
6) Number of particles (isolated binary images in a single well) are counted. If a particle contains fewer than a pre-determined number of pixels, the particle is eliminated from the binary image (i.e. all its pixels are turned to 0)
7) We adopt similar approach as step 6, verifying that the average green intensity of the pixels in the original image that correspond to 1 in the binary image is above the green threshold, and that the centroid of each particle is no more than 90 pixels away from the center of the image.

GI good for inside

HSV channel good for edges

Color threshold conducted based on green intensity, as in step 1 (middle) and HSV channels, as described in step 2 (right). Green intensity helps define most of the duckweed area, whereas HSV allows for better definition of the boundaries/interface of the duckweed.

combine both

As shown in step 4, correction fills in small gaps, while ensuring that larger holes, as the one in the middle, are kept out of the duckweed area

smoothen interface after combining HSV and GI

Smoothening of the binary image removes sharp edges, as described in step 5

Frond analysis

To count the number of fronds, we use skeletonization, which simplifies a binary image into 1-pixel lines. After pruning for shorter branches, we can estimate the number of fronds.

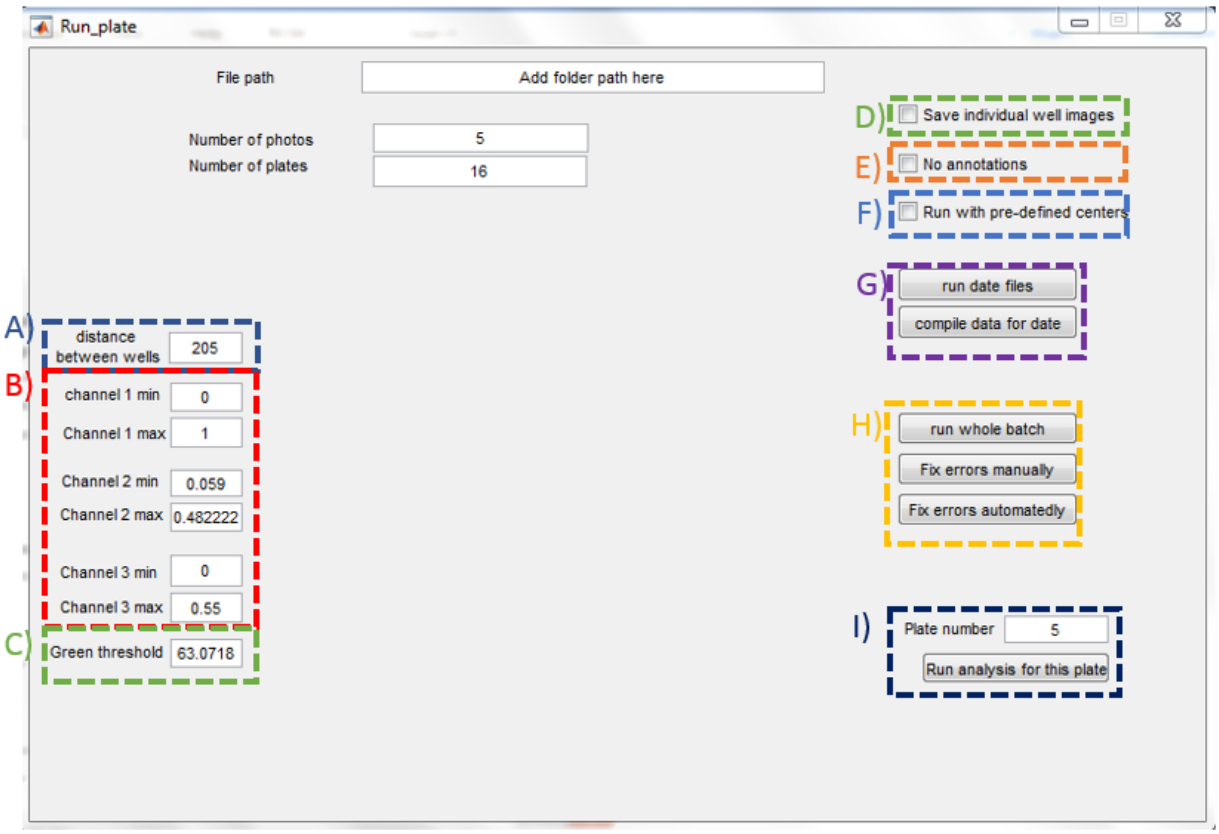
Post processing

After determining all features from each of the 96-wells, all information is saved into a single excel spreadsheet, along with the images of each well. A check for truncation error is also conducted, by verifying if any of the binary particles touches the edges of the image. If so, an error file is created, and the user can correct it be running a manual correction.

To combine the data for each date, all the data for each plate is copied and pasted into a single full_data file. For further analysis, the data for each repeated photo on the same day (i.e. 5) are averaged.

Instructions – Image processing software

The user-interface of the software is shown below:



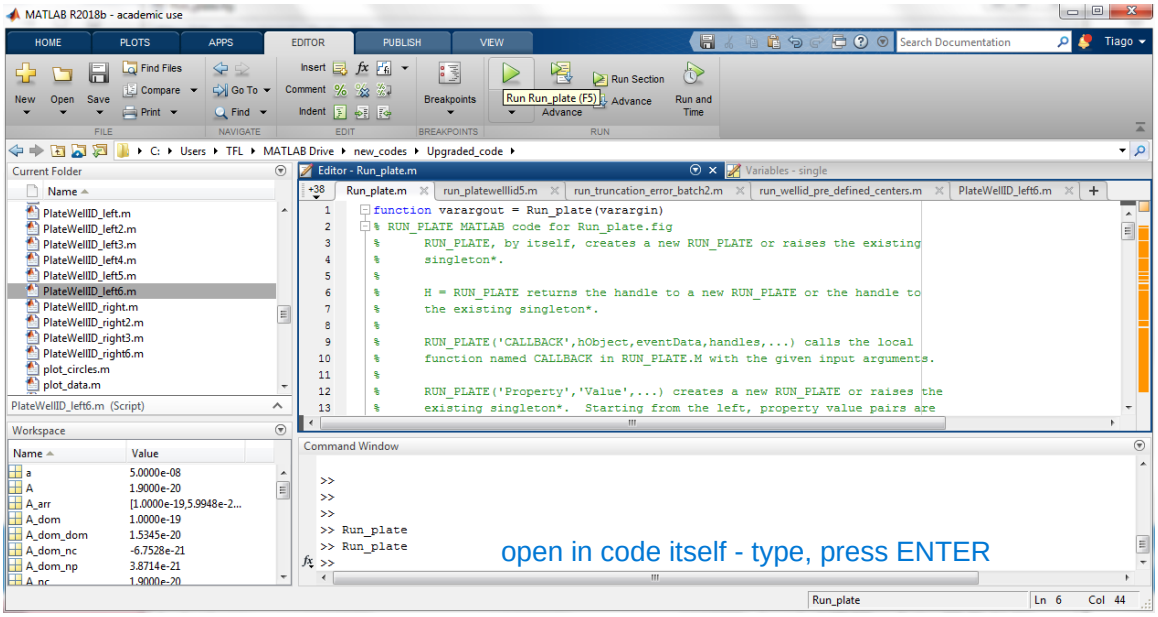
A)	Define distance between the center of two wells, in pixels. You can use trial and error to find it
B)	HSV channel thresholds, where channel 1, 2 and 3 correspond to Hue, Saturation and Value respectively. Each pixel has to be within these limits in other to be considered duckweed pixel
C)	Green threshold limit, which is used in combination with HSV values to enhance feature detection of the duckweed
D)	Check this box if you want to save the individual well images. Note that doing so requires more memory storage, but it is recommended for optimizing color thresholds using the color threshold app
F)	Check this box if you had run the same data previously, or if you have the 'Plate#_centers.csv' in the corresponding data folder. This will improve speed considerably, as the code will not have to search for the wells again. If this is the first time images are being processed, leave it unchecked
G)	Here, you can run a whole date folder containing several images of plates. The lower pushbutton is for compiling the data for a date folder into a single excel file
H)	Here, you can run a whole batch, which should contain one or more date folders. The middle button is for fixing errors using the manual correction tool. The lower button uses the same automated approach as to find the wells to try to fix errors. You may have to run these two buttons multiple times to clear out all the errors

I)	To run an individual plate, first write the desired plate number and then press 'Run analysis for this plate', where you will be prompted to locate the image file in your disk

Running batch/date folders

To open it, follow the steps below

- ➔ **Open MATLAB or Open 'Run_plate.m'**
- ➔ **Press Run**

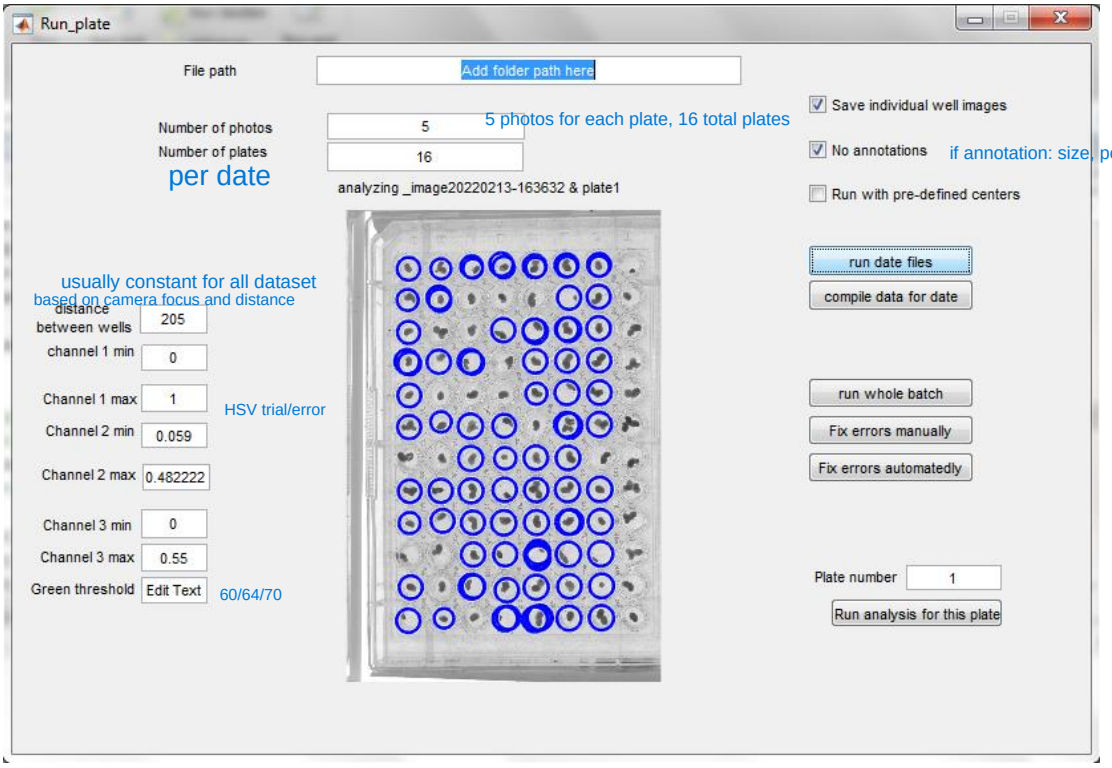


- ➔ **Define parameters for image analysis:** including color threshold, whether you want to save individual images for each well (more memory required, but recommended to view how the code is performing), annotations (leave unchecked for first round to verify if code is doing a good job at selecting the duckweed) and run with pre-defined centers (if you have not run the code for the dataset before, leave it unchecked)

- ➔ **Add file path on the top.** Whether it is a batch (a folder containing many dates), or a date file (a folder with plate images directly from the image capturing system), click on the desired button

A window directing you to the folder selected will open, allowing you to confirm your selection.

Once confirmed, the code will start running through all the images, and create a data folder within every date folder, where your images are stored.



Within each data folder, there are sub-folders named after all the images. Within each sub-folder, you may find the individual well images, with the plate number, row and column id, the data for each plate, location data for each plate, and a picture of the well plate with the located wells.

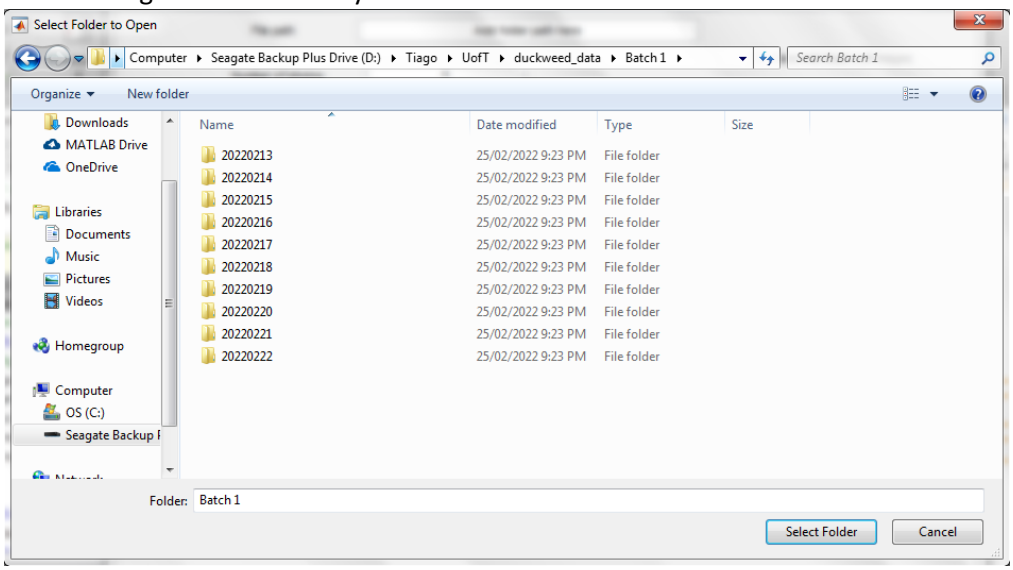
If the software is unable to find the well locations, or if it detects truncation error, it will create an error file. Later, you can select to fix errors by clicking in one of the buttons in the GUI, which will look for these error files and attempt to fix them.

Fixing batch errors

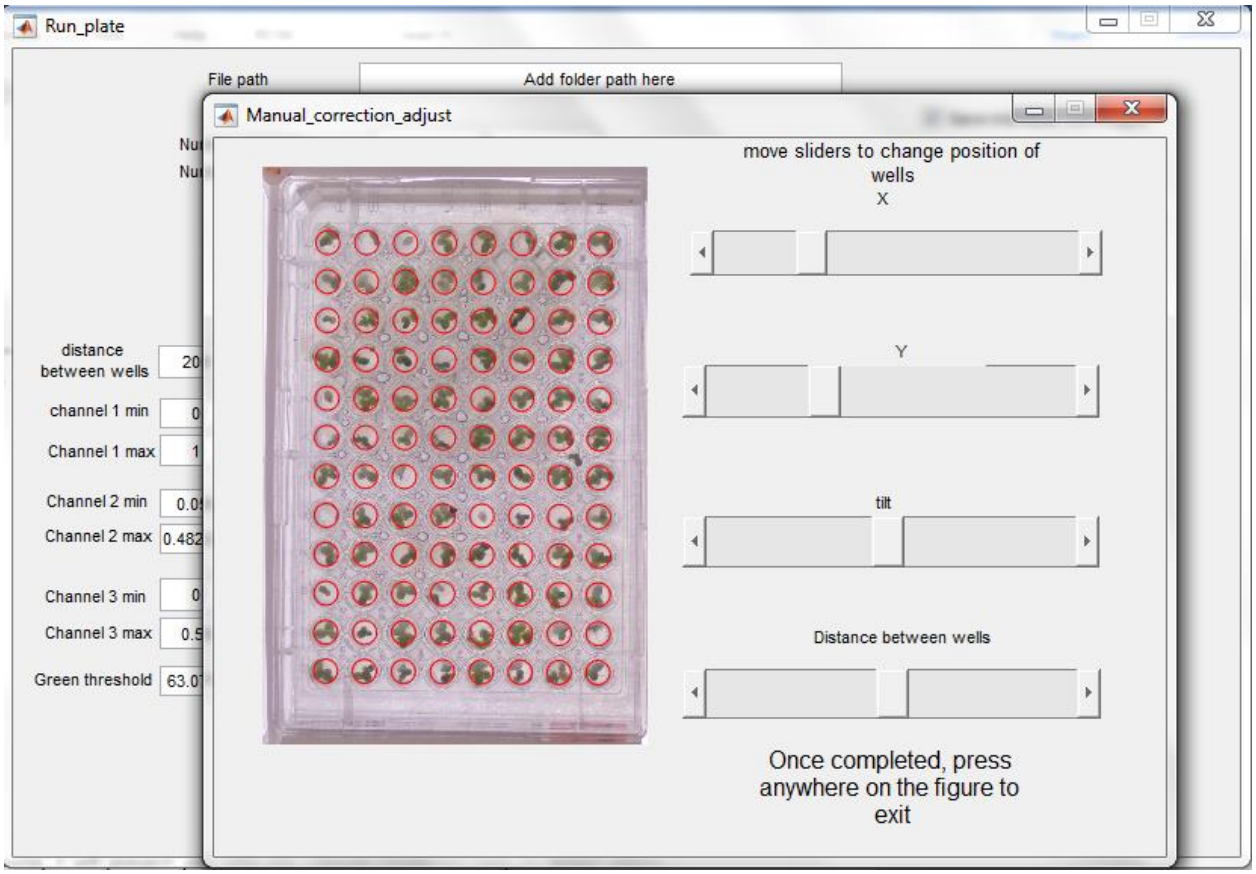
To go through a batch and fix errors that the code has identified, press one of the options (‘Fix errors manually’ or ‘Fix errors automatically’)

For manual corrections:

1) You will be prompted to select a batch folder. Once you press ‘Select’, the code will immediately start looking for error files in your dataset.



2) If any error is found, the manual correction tool will pop out, as shown below:



3) Follow the instructions on the app, by moving the position of the wells in the horizontal and vertical directions, changing the tilt/inclination of the plate, and increasing or decreasing the distance of the wells. If you don't see any red circles when the app starts, wait a few seconds before adjusting the sliders.
4) Once you have a good match between the red circles and the wells, with most of the duckweed inside of the red circles, you can click anywhere on the plate figure to exit.
5) The code will now take the new position values and use them when processing the images of the duckweed for the plate you have worked on, once all the errors are checked.

need to first fix errors

Creating full data for date

To compile the data obtained after running all the images for a specific date, click on 'compile data for date', and select the date folder

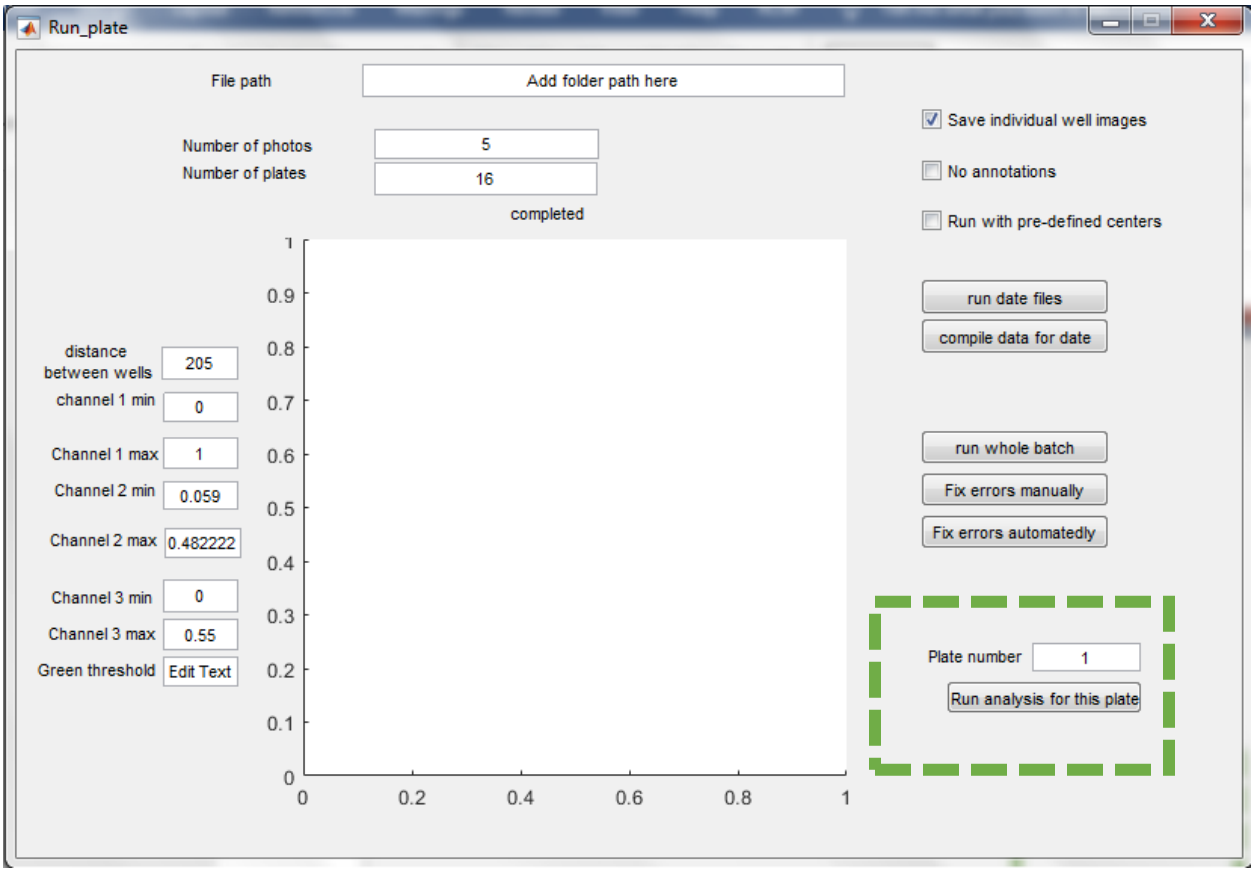
A message will appear on the app. Wait until the message changes to 'Finished creating full data file for:…'

An .xlsx file named 'full_data' will be generated in the data folder. Please note that the row numbers, which range from 97 to 103, are equal to 'a' to 'h'.

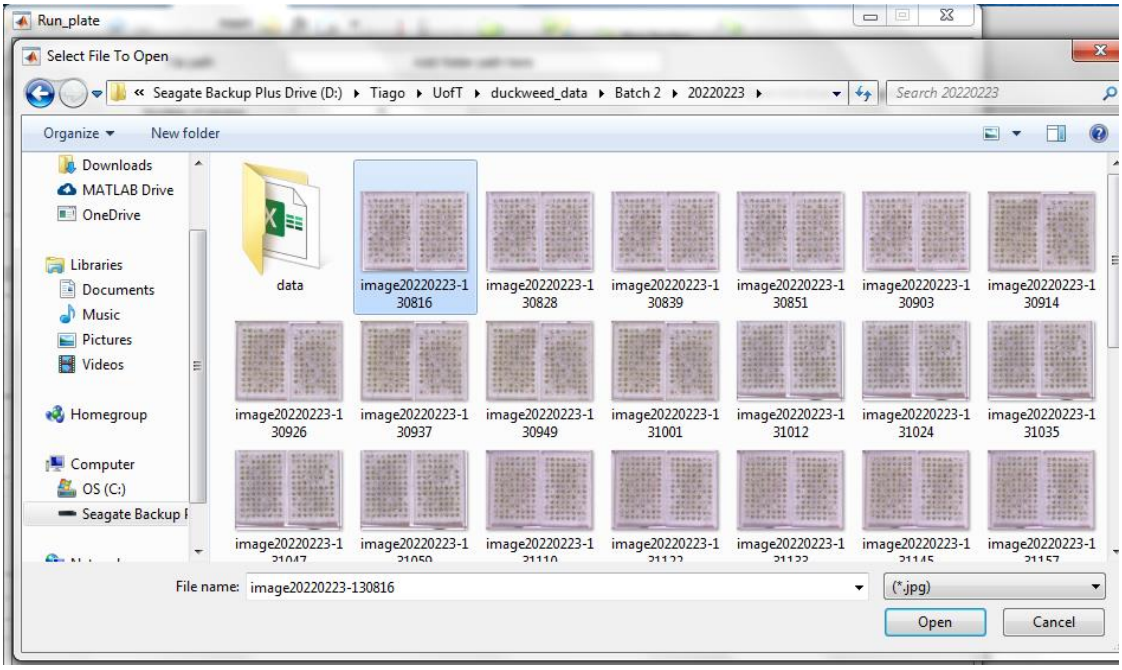
Running individual plates

You can choose to run an individual plate, instead of a whole directory.

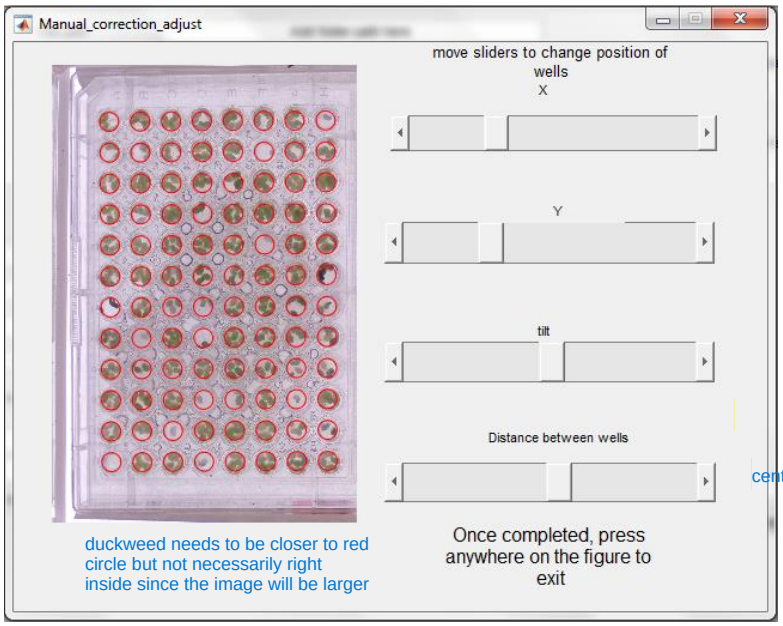
1) Make sure that the plate number is the desired one on the bottom left side of the window



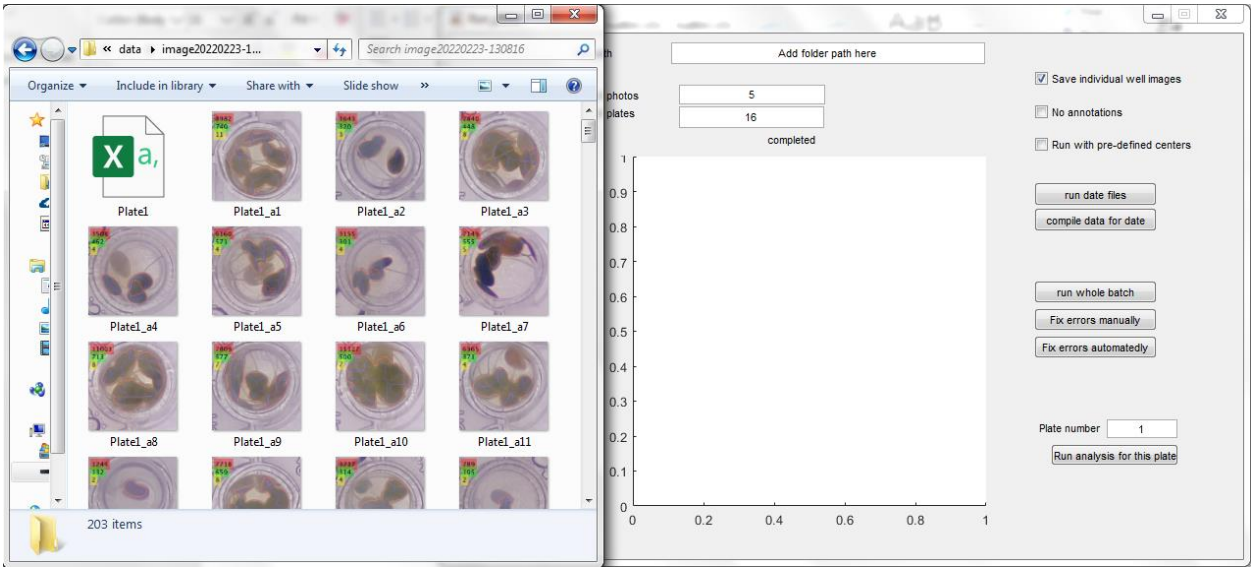
2) Next, click on 'Run analysis for this plate', where you will be prompted to select the picture in the folder. Select the desired photo by pressing 'Open'



3) The code will split the image into two, and open the manual correction. If this plate has been run before, the previous positions of the wellplate will be displayed in the manual correction. Make the necessary adjustments and press anywhere on the image when completed.



4) Data will be automatically processed and saved. Once it's done, a message will show up in the run_plate image processing app saying 'completed'



Compiling the whole data into a single file

- ➔ Open compile_data_table2
- ➔ Change the location where your data is located (and add '\Batch ' after) and the location and name of the file containing the dataset once it is compiled, as highlighted below:

```
%initializing the table where the whole dataset will be appended:
full_data_file=[];

%% define the location where data is being stored here:

location_with_batch="D:\Tiago\UofT\duckweed_data\optimized_settings\Batch ";
save_location_name="D:\Tiago\UofT\duckweed_data\optimized_settings\complete_data.xlsx";
%%
for batch_num=1:4
%path_name=strcat('C:\Users\TFL\Downloads\batch',num2str(batch_num));
path_name=strcat(location_with_batch,num2str(batch_num));
m=dir(strcat(path_name));  m=struct2table(m);
rows = (m.isdir==1);
vec_files=table2array(m(rows,1)); vec_files=vec_files(3:end);
var_names={'wellrow_arr', 'wellcol_arr', 'area_arr','frondnum_arr', 'frondnumfft_arr', 'maxF_arr', 'minF_arr', 'pe


for date_num=1:size(vec_files,1)
date_val=cell2mat(vec_files(date_num));
```

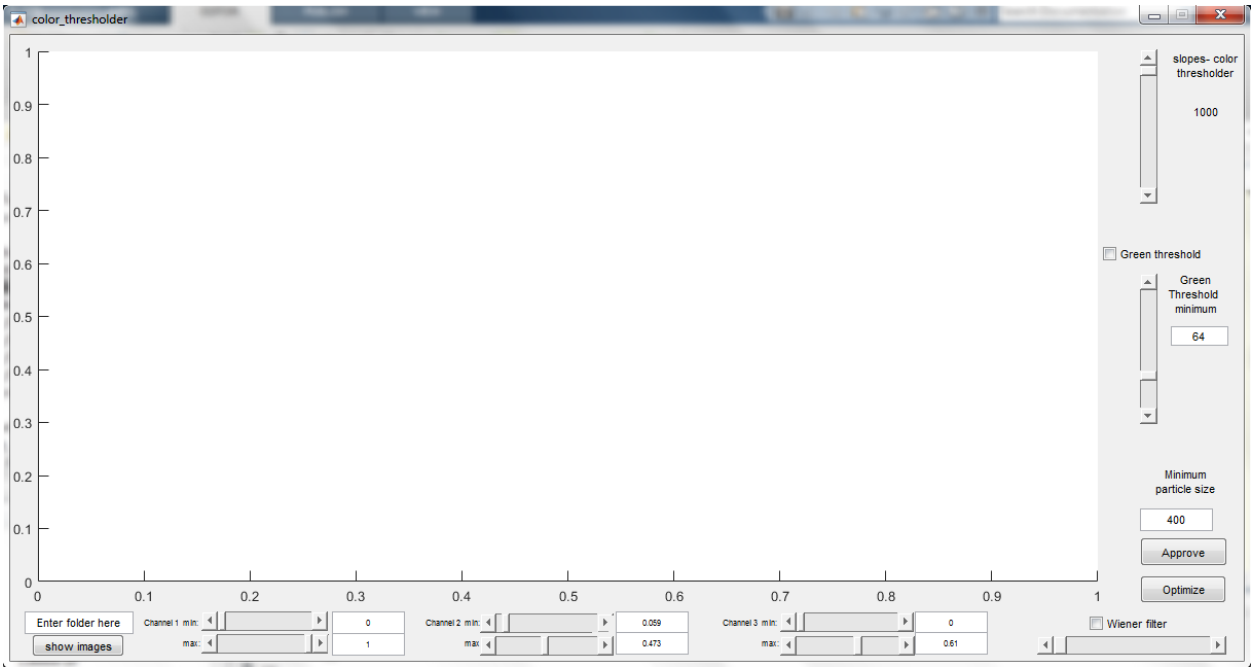
Warnings:

- ➔ **Always make sure all the files have been properly uploaded. A missing file may cause an error or lead the software to mis-assign the plate number to the actual plate**
- ➔ **To stop the operation of the code in the middle of the run, go back into the MATLAB window, click on the command window and press CTRL+C (same as copying shortcut)**
- ➔ **Never open two GUI's at the same time, unless one GUI opens another one (i.e. the image processing GUI can open the Manual correction GUI)**
- ➔ **To use the well images in the color threshold app, make sure to check the "No annotations" box**

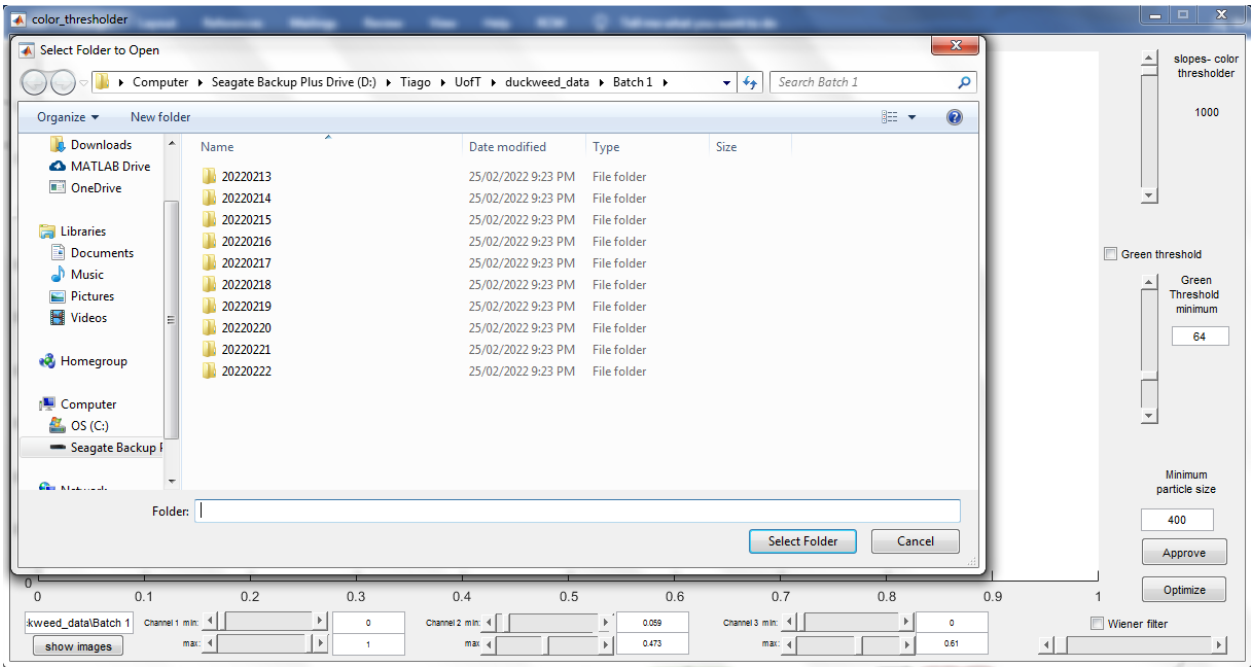
Color threshold app

detect how well colour threshold settings are working for a random set of images

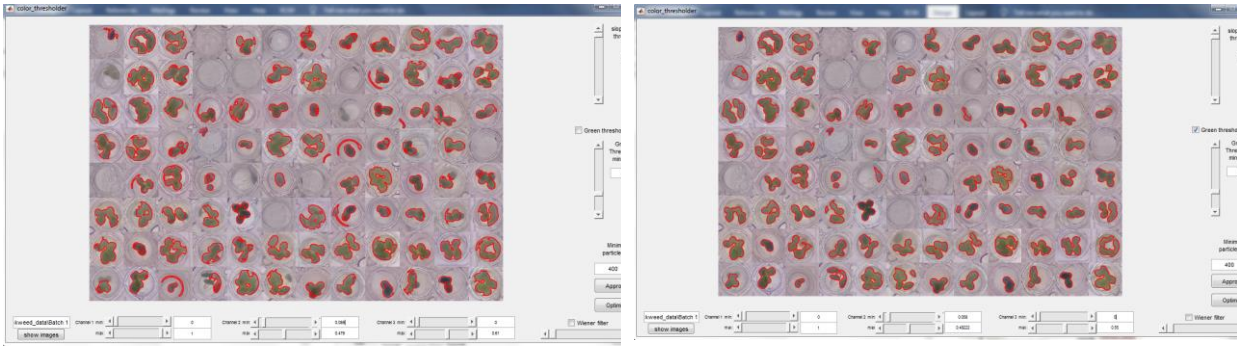
- ➔ **Open color_thresholder.m, either in MATLAB or in the folder where the file is at.**
- ➔ **Press run**



To run the code in normal mode, add the batch folder name in the field on the bottom left side of the window and press ‘show images’. A window showing the folder will appear. Confirm the folder by clicking on ‘Select Folder’



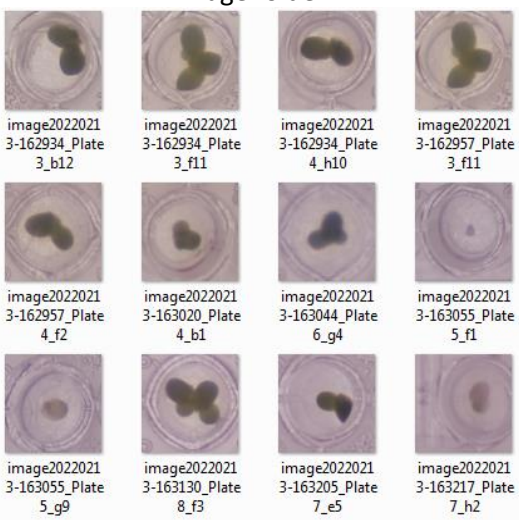
A random set of 96 individual well images will appear on the screen. Now modify the channel values and the green threshold minimum, and click on ‘green threshold’ checkbox.



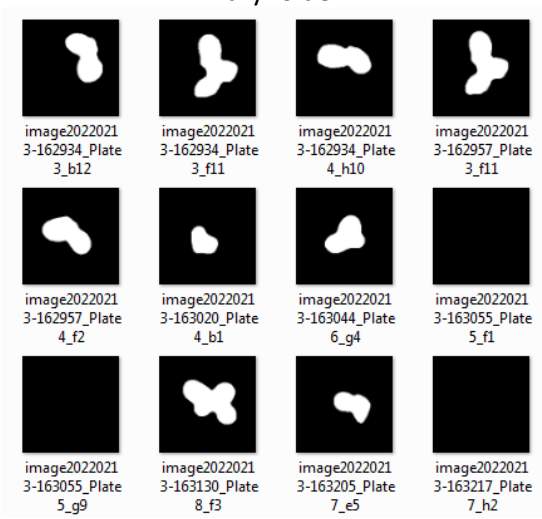
Improving color settings

1) Create two folders in any directory of your choice, one named ‘images’, and one named ‘binary’
2) Open ‘approved_wells.m’ and change the variables ‘destination_training_images’ and ‘destination_training_binary’ with the full location address of ‘images’ and ‘binary’, respectively.
3) Save the M file, now go back to the color threshold app.
4) To add images to the training set to improve color threshold settings, find an image where you think the code is properly identifying the duckweed area. Click on ‘Approve’ and then click on the specific image you chose.
5) The raw image and its binary version will be saved in the folders you had chosen in steps 1) and 2).

Image folder



Binary folder



6) To improve color threshold settings using the training set, go back to the color threshold app and click on ‘Optimize’ in the bottom right side of the app. The process may take a few minutes, but may last longer depending on the size of your training set. A message saying ‘Optimizing thresholds…’ will show up. Wait until the message is cleared.
7) Once completed the optimization, the optimized channel values will automatically change in the color threshold app.

smaller error = better performance

Checking performance over time

1) You can also check how the color threshold is performing over time for a number of randomly chosen duckweed. This will also produce a gif file showing the growth of duckweed over time. To do this, write ‘g’ in the ‘Enter folder name’ space on the bottom left of the window, and press ‘Show images’. You will be prompted to select a folder (batch directory).