

Supplement S2: Image Analysis Manual

This document is supplementary material for the article (currently in peer review):

Brefeld, D., Di Mauro, V., Kellermann, M.Y., Nietzer, S., Moeller, M., Lütjens, L.H., Pawlowski, S., Petersen-Thiery, M. and Schupp, P.J. (2023). Acute Toxicity Assays with Adult Coral Fragments: A Method for Standardization. *Toxics* **2023**, *11*, x. <https://doi.org/10.3390/xxxxx>.

Please cite the paper if you use the image analysis workflow and/or the ImageJ macro for your own analyses.

Introduction

Bleaching of *Montipora digitata* coral fragments was investigated using image analysis. Pictures of the fragments were taken at the beginning and at the end of the experiment (after 96h) with the same camera settings under the same lighting conditions.

This manual illustrates how the resulting images were analyzed.

Preparation of Images for Analysis in ImageJ

Images were taken in Raw format (.CR2). The image files were imported into the open-source photography workflow application “darktable” (v4.0.0; www.darktable.org) to prepare them for further analysis in ImageJ.

Preparation of images was accomplished in 4 steps:

1. Using the metadata editor, the title of each image was set to indicate the fragment ID and the timepoint at which the image was taken (Figure S2-1).

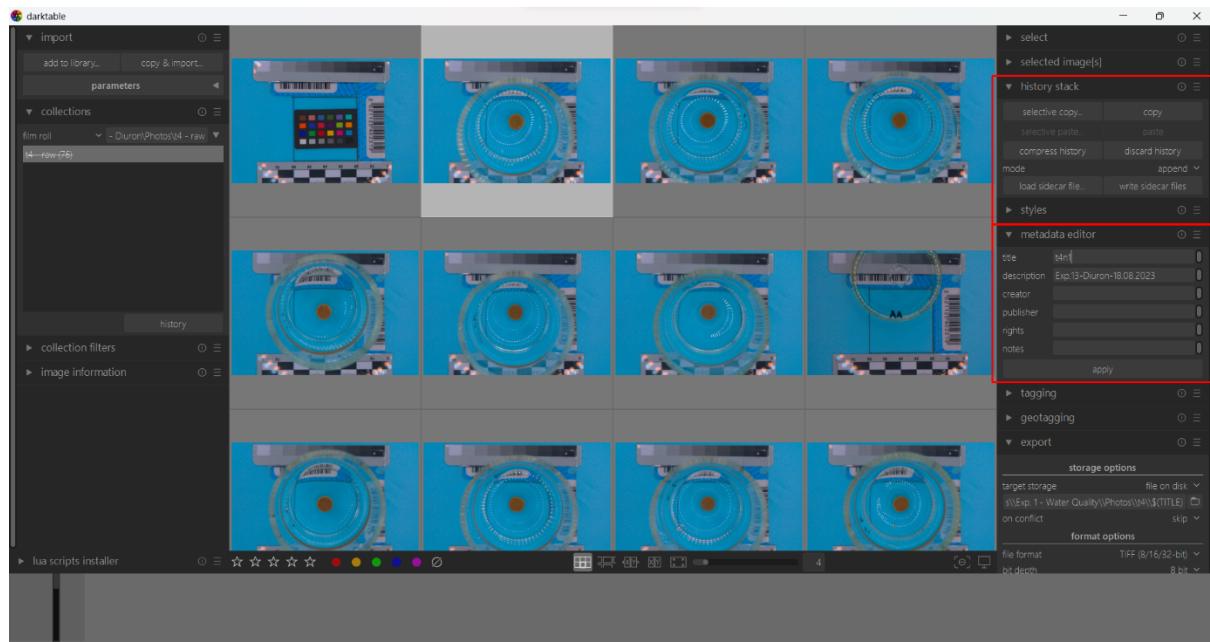


Figure S2-1: Lighttable tab of darktable, showing the imported images. The history stack and metdata editor are highlighted in red.

2. Double-clicking the reference image with the Calibrite color checker (Figure S2-1, first image) opens the darkroom tab that features many image processing options. Since our intention was to keep the images as true to the original as possible, all modules except “input color profile”, “output color profile”, “demosaic”, “white balance”, “color calibration”, “lens correction”, and “raw black/white point” were triggered off (Figure S2-2 & S2-4). Input and

output color profiles were set to AdobeRGB for good compatibility. Demosaic, lens correction and raw black/white point remained at their default settings. The white balance was set manually to the white reference field of the color checker.

The “calibrate with a color checker” option of the color calibration module was used to calibrate the color of the image using the photographed color checker (Figure S2-3).

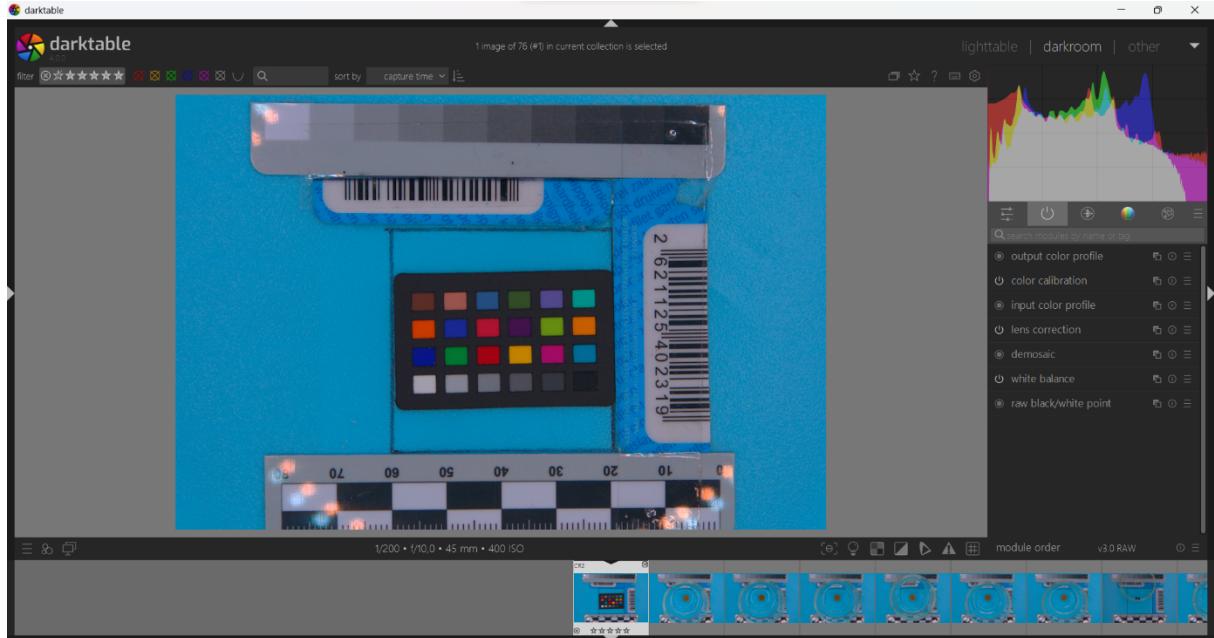


Figure S2-2: Darkroom tab, showing the modules used for image processing.

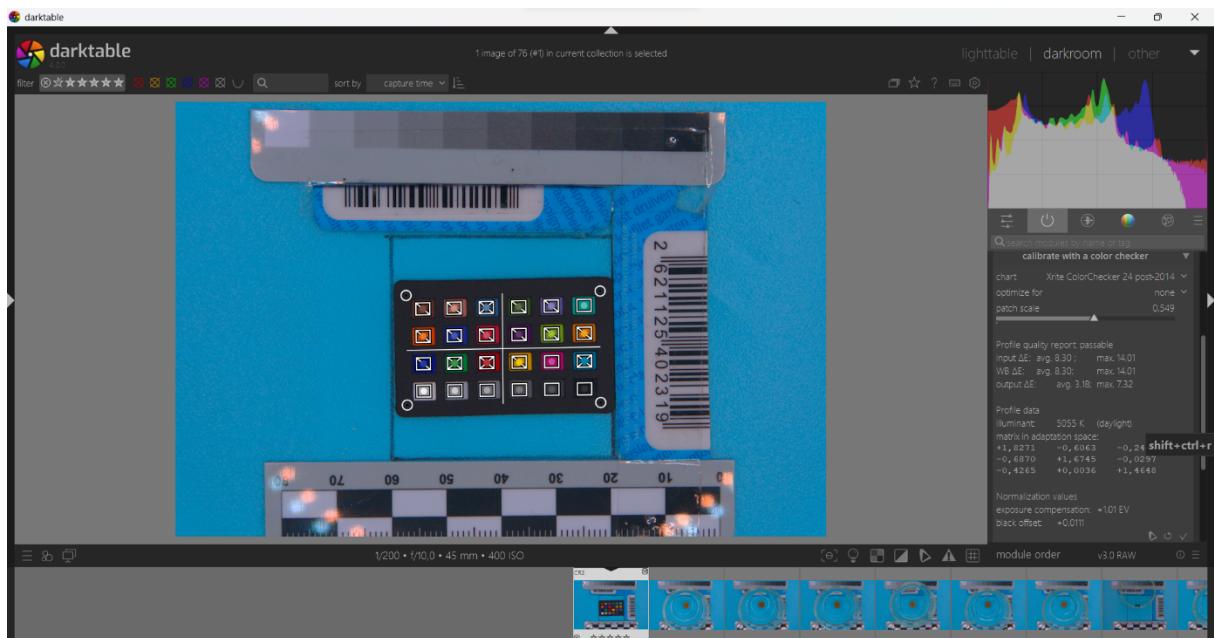


Figure S2-3: Darkroom tab, showing the color calibration module. “Calibrate with a color checker” was used to adjust the image to the Calibrite color checker.

3. Using the “selective copy...” function of the history stack (see Figure S2-1), the processing options were copied from the reference image (Figure S2-4) and applied to all other images using the “paste” function. Since all images were taken under the same conditions, all adjustments made to the reference image should be equally appropriate for the images of the coral fragments.

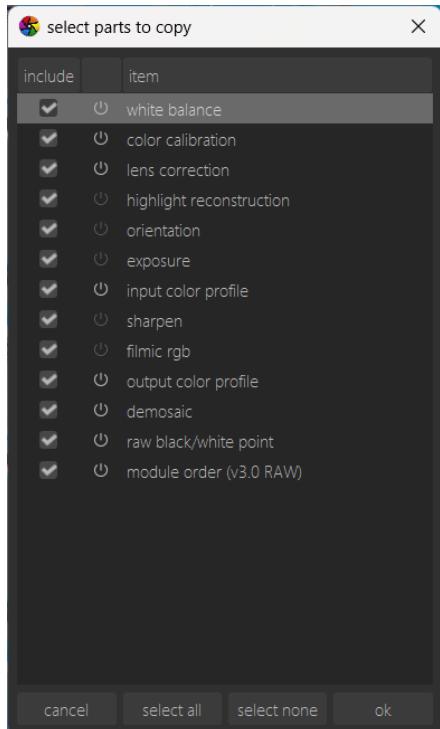


Figure S2-4: Choice-menu of the history stack’s “selective copy...” function. All module preferences were copied from the reference image.

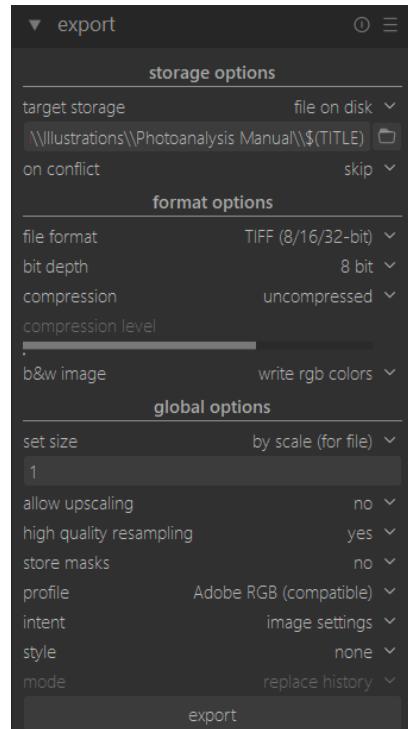


Figure S2-5: Export options. Images were exported as 8-bit uncompressed TIFF-files at full resolution with AdobeRGB profile. The filename was set to be the previously set title.

4. After the adjustments were applied to all images, they were exported as uncompressed 8-bit TIFF-files for further processing in ImageJ. The previously set title was used as filename, so that the image file indicates the photographing timepoint and fragment ID. The resolution was kept at full size (set size by scale = 1). However, it is possible to adjust the resolution (e.g. set size by scale = 0.5) to dramatically reduce file size and processing effort. The profile was chosen as AdobeRGB and all other options remained at default. Take note that setting the conflict action to “skip” will skip any image with the same title except the first occurrence. (Figure S2-5)

Image Analysis in ImageJ

The Fiji distribution [1] of ImageJ [2] was used to quantify the pigmentation of the coral fragments.

Each pixel of an 8-bit greyscale image can take a value between 0 and 255 where 0 is black and 255 is white. As bleached corals appear lighter in color (strongly bleached corals appear almost white), the average greyscale intensity of the coral fragments was taken as a measure for their pigmentation. The bleaching intensity was estimated by calculating the difference between the average grey intensity of the coral fragments at the beginning and at the end of the experiment. The greater the difference between the starting and end point, the stronger the bleaching.

To semi-automatically process the images, an ImageJ Macro was written that incorporates a full image analysis pipeline. Conceptually, the images are first filtered to make them more homogeneous and increase contrast to the background. Then, automatic thresholding functions are applied to segment the images into the foreground (= coral fragment) and the background. Finally, areas of interests (ROIs) created by the thresholds are improved and used for the calculation of the fragments' average greyscale intensity. The GUI of the macro (Figure S2-6) allows to intuitively alter the analysis pipeline to improve segmentation of the object of interest (the macro might also work for other coral fragments!).

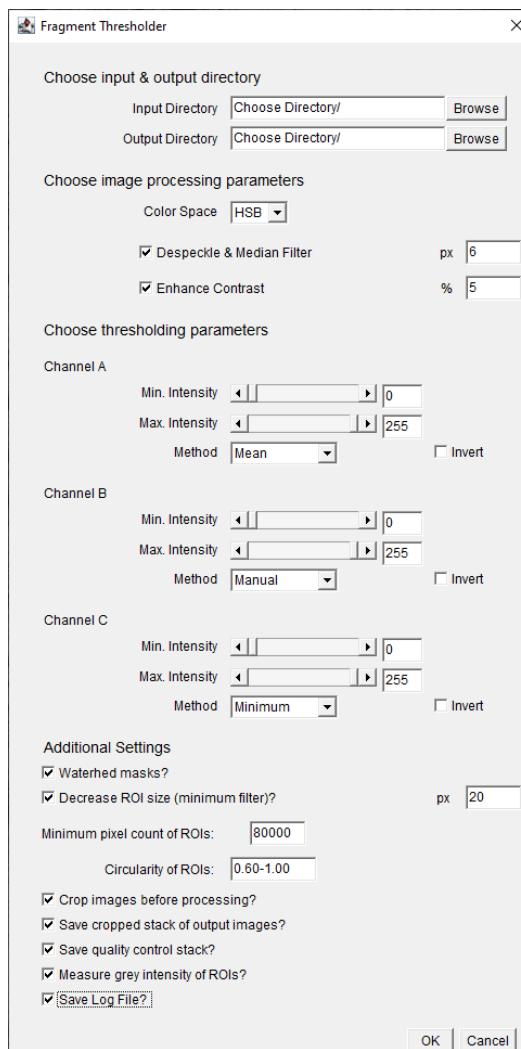


Figure S2-6: GUI of the “Fragment Thresholder” ImageJ Macro.

In the first section of the macro, the input and output directory paths need to be supplied. The input folder should contain only the TIFF-Images of the fragments and the output directory should be another empty folder. If the output folder is not empty, the macro will attempt to clear it (with warning).

In the second section, the image processing parameters must be chosen. It is possible to use the RGB, HSB and Lab color space. We found that applying threshold to the Hue and Brightness channel of the HSB space works well. The “Despeckle and Median Filter” applies the two functions to any slice of the chosen color space stack. We found that a median filter with a size of 6 pixels works well, but stronger or less aggressive filtering is possible, too. The “Enhance Contrast” function applies this function to all image slices with the intensity stated behind. We found that a value of 5% saturated pixels works well for our images but realize that this is very aggressive and might not work for other color spaces or images.

In the third section, the threshold can be chosen for each image channel individually. The functionality is approx. equal to the “Color Threshold...” Plugin found in ImageJ. Only the areas that are segmented in all channels will make up the final masks which allows great flexibility. Thresholds can either be set manually or chosen from the full set of automatic methods that are supplied with the Fiji distribution. We found that using the “Mean” thresholding method for the Hue channel and the “Minimum” method for the brightness channel gives good results for our images.

Note that it is possible to invert each threshold which might be valuable in some cases.

The final section offers a range of “Additional Settings”:

- “Watershed masks?”: Applies the watershed function to the masks which might cut off undesired sections included in the masks of the fragments (e.g. filamentous algae). Note that this function is only helpful for relatively homogeneous fragments. For more irregular fragment shapes, the function will attempt to cut the mask of the fragment!
- “Decrease ROI size”: Applies a minimum filter of the given size (e.g. 20 pixels) to the masks. We use it to exclude the edges of the fragments that are often more variable/irregular in color than the center of the fragments (see Figure S2-8). It is necessary to turn this function off if the whole fragment should be measured (e.g. for size estimation).
- “Minimum Pixel Count of ROIs” and “Circularity of ROIs”: Supply settings to the particle analyzer which calculates the average grey intensity of the fragments. It is possible to exclude areas from the analysis that are smaller or more irregular than the fragments but might have incorrectly been segmented by the thresholder. Note that the number of pixels much depends on the resolution / size of the image that is supplied and must be adjusted accordingly.
- “Crop images before processing”: Offers the option to crop the input image to the area in which the subject is located. This can be used to automatically remove most of the background and increase processing speed (Figure S2-7).
- “Save cropped stack of output images”: Saves a stack with all cropped images of the fragments where the background was removed (= black). This stack can be used for further analysis as a new threshold can easily be applied manually.
- “Save quality control stack”: Saves a stack of all cropped images including an overlay of the ROIs. This stack can be used to assess the quality of the selections. This stack will be open when the macro ran successfully (Figure S2-8).
- “Measure grey intensity of ROIs”: Measures the average grey intensity of each fragment. The results are displayed in the results window (Figure S2-8) and saved as a CSV-file. It is possible to copy the results from the results window directly into excel. The images are analyzed in alphabetical order of the filenames so that it is easy to allocate the results to an existing spreadsheet (as long as all filenames contain the fragment IDs).
- “Save log file?”: Saves the log file that pops up when the macro is started (Figure S2-8).

When all save options are activated, the results folder will contain individual cropped images of every fragment with removed background and their ROIs, as well as a stack of such images that has been cropped to the smallest possible size. Also, the quality control stack with the accompanying ROIs are saved.

If necessary, the quality control stack can be used to adjust the ROIs manually by using the selection brush (available by double-click on oval selections). To do so, the quality control stack and the ROIs of the cropped stack must be opened (Figure S2-8) and the overlays must be hidden (Image > Overlay > Hide Overlay). Now, a click on an ROI in the ROI manager will activate it so that it can be adjusted.

ROIs can be extended by holding the left mouse button and erased by additionally holding down the Control key. Hitting update in the ROI manager will update the current ROI. When finished, all ROIs can be selected and saved into a ZIP-file.

Limitations:

- The macro will only run successfully if only one ROI per image is created. Incorrect segmentation and/or filtering that leads to multiple ROIs in one image will lead to an error message. Changing the options might result in better results. The workflow of the macro can

also be followed manually to try out different color spaces, filters, thresholds, particle analyzer settings etc.

- The contrast between the background and the fragment needs to be strong and clear. If the coral is attached to a substrate, it might be very difficult to separate these. The same goes for algae or any particulate matter – although these can be removed manually relatively quickly.
- ➔ If the performance of the macro is not sufficient it might be valuable to have a look at the “Trainable Weka Segmentation” Plugin that is supplied with the Fiji distribution. The plugin offers the option to train a powerful machine learning model with a subset of your images and subsequently use the model to segment all images.

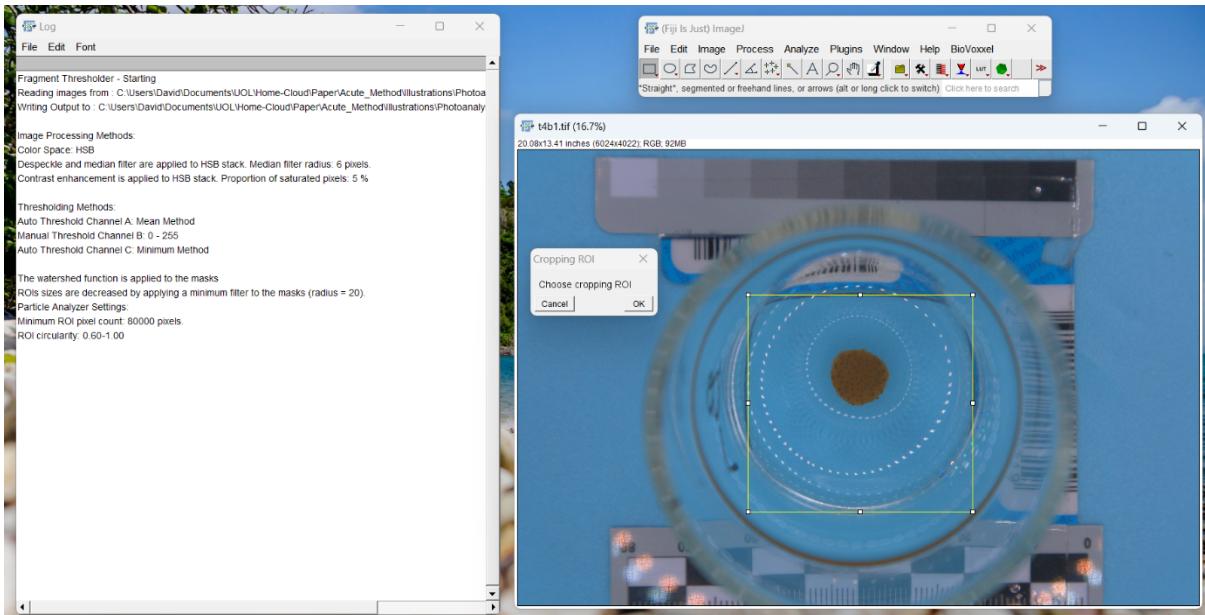


Figure S2-7: ImageJ macro execution. The macro asks to supply a cropping ROI which can be interactively determined using most of the selection tools.

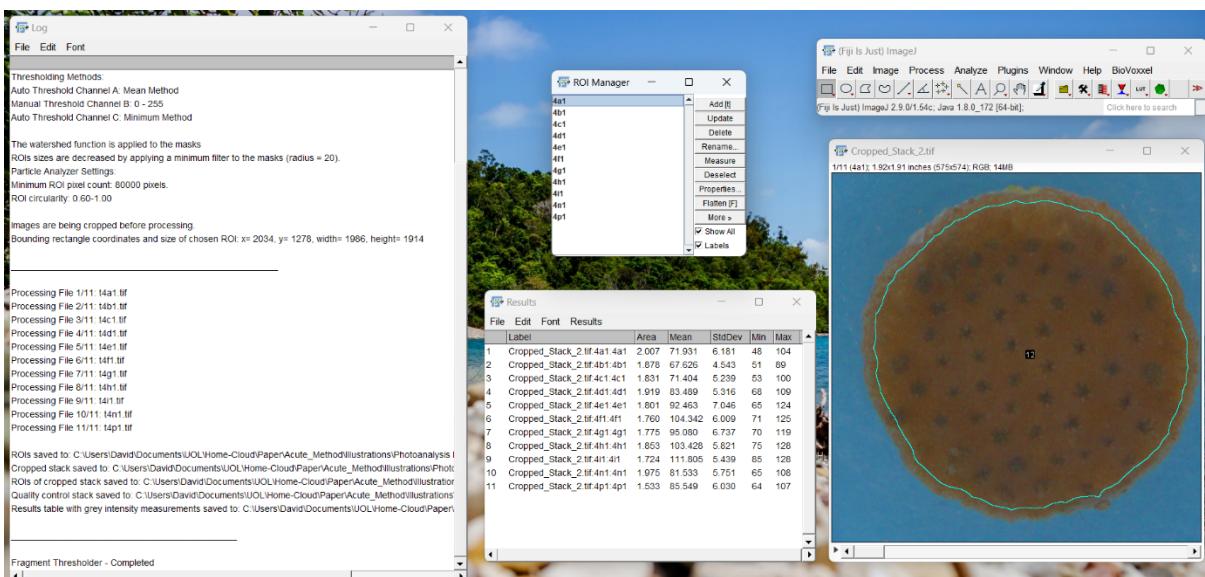


Figure S2-8: Result of a successful macro run. The log shows completion, the ROI manager holds a single ROI for each fragment and the results window shows the average grey intensity of each plug measured inside the according ROI. The image window shows the quality control with all Overlays/ROIs so they can be checked visually.

References

1. Schindelin, J.; Arganda-Carreras, I.; Frise, E.; Kaynig, V.; Longair, M.; Pietzsch, T.; Preibisch, S.; Rueden, C.; Saalfeld, S.; Schmid, B.; et al. Fiji: An Open-Source Platform for Biological-Image Analysis. *Nat. Methods* **2012**, *9*, 676–682, doi:10.1038/nmeth.2019.
2. Rueden, C.T.; Schindelin, J.; Hiner, M.C.; DeZonia, B.E.; Walter, A.E.; Arena, E.T.; Eliceiri, K.W. ImageJ2: ImageJ for the next Generation of Scientific Image Data. *BMC Bioinformatics* **2017**, *18*, 1–26, doi:10.1186/s12859-017-1934-z.