The present tools were created to automate several measuring tasks on fluorescent images of dyed caudal fin of the medaka fish. Working directory is set to C: drive (can be changed in **StartupMacros.txt** and **recalibration.R**). Although no examination has been conducted, it should be the case that these tools would perform similarly for other ray-finned fishes, especially the zebrafish (*Danio rerio*).

In the ImageJ folder, paste **StartupMacros.txt** in **macros** (all macros scripted or called in this file will be installed upon opening imagej, otherwise a macro has to be installed manually every session).

* “Make text”: Creates a text-form bit map (.txt instead of .bmp) for the image as is (output raw.txt) and its filtered outcome using the Gaussian kernel (output blur.txt). Upon running this macro, a window pop-up asks for the kernel’s sigma value; Generally, this value can be optimized per batch of images.
* “Stump fit”: As the fin regenerates, the portion anterior to the amputation plane (i.e. stump) continues to grow. This complicates measuring the plane’s length as rays grow at different rates at different periods. This tool allows one to select the fractures on each ray, from which *an* amputation plane would be estimated by OLS. If it’s unclear for one ray which fracture is indicative of the incising position (since fractures can manifest for any number of reasons), none should be selected – OLS is fairly robust for this purpose and of the 14-16 rays, leaving 2 or 3 out is not an issue.
* “Get Sr”: Sr stands for *regenerated area*. User delineates the fin’s shape using the line tool beforehand. “Get Sr” would take this input to produce the value of the area above the stump that has been estimated by “Stump fit” (by generating a polygon ROI).

***This polygon ROI must be maintained so that the area of* de novo *ossification can be measured. Since ray bone mineral is dyed with fluorescence, ROI deliniating the rays can be generated by thresholding. The following two are alternative methods:***

* “Get Sm”: Sm stands for *mineralized area*. Running “Get Sm” doesn’t produce the area per se but produces thresholding values using the T-point algorithm (output T-point) and the Triangle algorithm (output T triangle and D, for a sanity check). Since alizarin red was used, the weights for the color channels R, B, G were set 1, 0, 0 respectively. User should calibrate these weights so as to reflect their dye of choice.
* Recalibration.R: Run this R script in entirety (in R, not ImageJ). After applying “Make text” to the “Get Sr”-generated polygon, simply input *a* into R. In the background, the program runs an application of Arce et al’s method (DOI:[10.1038/srep02266](http://dx.doi.org/10.1038/srep02266)) – this method tackles the thresholding problem for non-homogeneous fluorescent subjects, where the *inner* intensity at places can be lower than the *outer* intensity. ***R is used because ImageJ is not optimized to handle such large matrices*.** The output **PnS.txt** can be opened in ImageJ then applied with the default thresholding tool.

Sample images available via https://drive.google.com/drive/folders/12h7iaSeJS1nptxfx24cdapj5ovo6txYj?usp=sharing