README POM Matlab processing (written by Lea - Jan 2022)

**Pre-processing**

*RhythmDevelopment/conversion.m*

Script to batch convert the black and white images (.rsh .rsh .gsd .gsh) transferred from the 9th floor cameras to be able to use in Matlab for later optical mapping processing

**Processing in 2D**

*RhythmDevelopment/rhythm.m*

Software with GUI provided by Dr. Efimov’s group to analyze the 2D optical mapping data (activation, APD, conduction, phase…), open-access but hard-coded (can’t be modified)

*RhythmDevelopment/MaskCropping.m*

Script that allows you to manually outline the 2D heart silhouette on the original black and white images, save the corresponding masks in all 4 views, apply it to the previously generated 2D activation or APD maps and lastly save the cropped maps

*RhythmDevelopment/Crop2DMapSave.m*

Script that allows you to crop the background noise from the 2D activation or APD maps, visualize the outcome and save the new maps. Simplistic version of the above MaskCropping code, doesn’t show you the initial black and white images to draw the silhouette

*RhythmDevelopment/whiteBackgroundNaN.m*

Script that allows after loading an activation or APD map to display it in 2D without a colored background but in transparent/white instead for presentation/paper purposes

*RhythmDevelopment/KeimaBinary.m*

Script used to process the Keima images in 2D: visualize the fluorescence in black and white, normalize the signal, binarize it according to 25/75% intensity, visualize it in 2D and if necessary, crop the outcome to get rid of ventricular fluorescence signal (artefact) resulting from laser reflections

*RhythmDevelopment/**DispersionAPD\_2DAtria.m or DispersionAPD\_2DVentricles.m*

Script that allows you to analyze the 2D APD maps: compute the standard deviation to derive the APD dispersion in a nearest neighbor square fashion (in a 3x3 ROI or 5x5 ROI in atrial or ventricles respectively) and binarize it according to a hard-coded set dispersion threshold (8ms or 10ms for atria and ventricles respectively). The ROI sizes and threshold values can be adjusted manually in each case and potentially optimized in the future.   
For quantification purposes, depending on whether you are running stats in the atria or ventricles, use the corresponding code accordingly “…\_2DAtria” vs “…\_2DVentricles”.

*RhythmDevelopment/APD\_AFvsAFlutter.m*

Script that allows you to analyze the 2D APD maps in all 4 views (useful mostly in tTG/dTG micen when investigating the APD AF vs AFL patterns, cf. HRS Dec 2021 abstract).   
Draw the ROIs in the RA/LA to measure the minimum, maximum, average and standard deviation values of the APD and APD dispersion.

Other Matlab functions within the *RhythmDevelopment* code folder (such as *phaseMap.m*) have been adjusted by Lea to avoid frequent errors and bugs, as well as speed up the process (e.g., phase movies are now directly saved with the cropped 2D silhouette mask among other things…)

**Processing in 3D**

*PanoramicImagingGUIv10/PanoramicImaging\_gui.m*

Software with GUI originally implemented by Pierre to generate calibration matrices, create 3D heart silhouettes and project optical maps onto the corresponding 3D volumes to later be exported and visualized in AMIRA.

This code as well as some separate Matlab functions being called by the 3D panoramic GUI have since then been adjusted by Lea to avoid frequent errors and bugs, as well as facilitate reproducibility (consistent rotation axis selection and similar silhouette) within a same mouse dataset.