Rhythm GUI User Manual

MATLAB software for analyzing optical mapping data

5/25/2012

Washington University in St. Louis – Efimov Lab
By Jake Laughner, Matt Sulkin, Birce Onal, Xinyuan Sophia Cui, Jed Jackoway, Connie Shao

Rhythm GUI User Manual

Contents

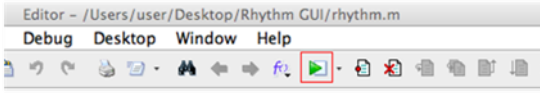
General Purpose

Rhythm is a MATLAB-based graphical user interface to display, condition, and analyze data in an easy-to-operate graphical environment. This program was designed to load and process data from the MiCAM ULTIMA imaging system from SciMedia. This current version is optimized to process voltage data only. Later versions of this software package will include analysis of voltage and calcium data collected from multiple cameras. To obtain a copy of the source code, and for any questions and suggestions, please email igor@wustl.edu. For customization, please see the Appendix.

1. Get Started

Open the Rhythm GUI folder and double click on the *rhythm.m* file.

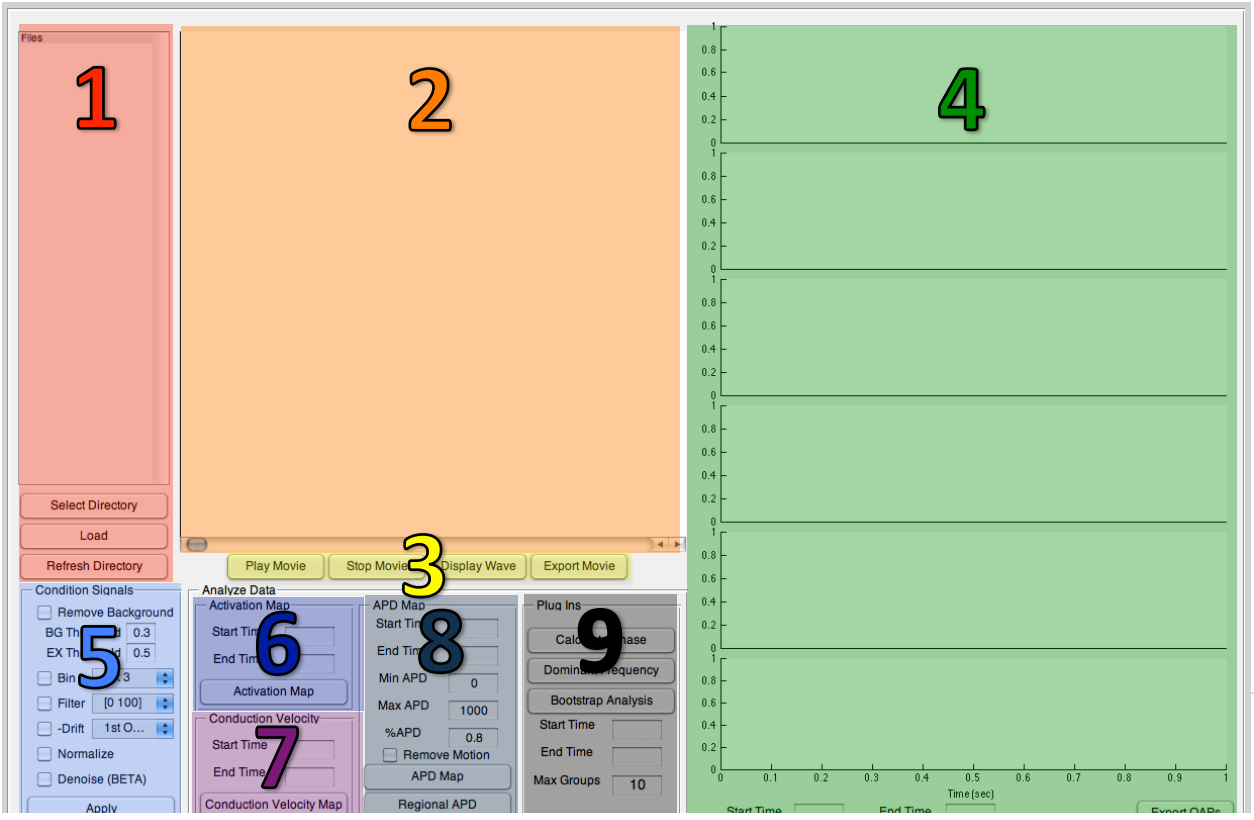
Run the algorithm using the green play button.



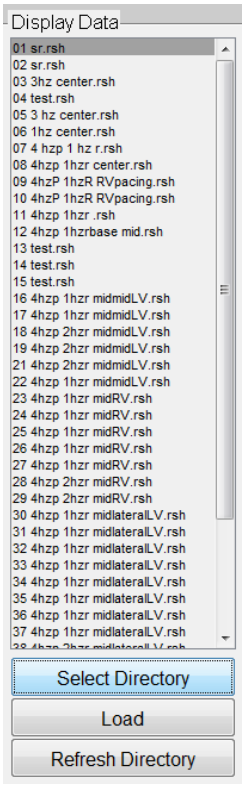
Select the *Change Folder* option if an error pops up.

The software is organized into seven sections:

1. Load Data
2. Movie Screen
3. Display Control
4. Signal Screens
5. Condition Signals
6. Activation Map
7. Conduction Velocity
8. APD Map
9. Plug Ins



1. Load Data



1. Select Directory

Choose the folder that contains all .rsh data files. All the .rsh files in the folder will be displayed. Note that .rsh is only a pointer file that allows the program to locate the other related files, such as .rsd and .rsm files. These files are raw output data from the camera and all contain valid information; do not remove any of them from your folder.

2. Load

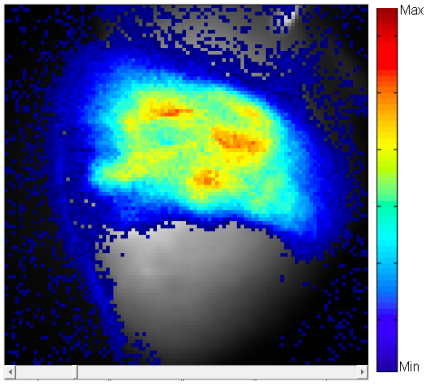
To perform analysis on a set of data, select the name of the desired file in the file list and click the *Load* button. The movie screen automatically displays the first frame of the raw data.

- Data without a .mat file: The program will convert the raw data into a .mat file first, then load it
- Data with a .mat file: Data will load directly
- No data: Error

Refresh Directory

Click 'Refresh Directory" to update the .rsh files in the selected directory. This button is useful during an experiment when the data list is changing.

2. Movie Screen



Movie Display Window

The *Movie Display Window* is a 100 x 100 pixels screen that displays the 2D data at one particular frame. The grayscale background shows the camera's field of view and the relative position of the tissue. The color map superimposed on the grayscale background shows the intensity of fluorescent emission. The color bar uses red to indicate the maximum and blue to indicate the minimum intensity. Pixels lacking a color map layer are treated as background channels with no significant data.

The scrollbar can adjust the current frame that is displayed on the Movie Display Window. If the user clicks on the white area, the bar immediately jumps .02 seconds towards the location clicked.

3. Display Control

Play Movie and Stop Movie

After *Play Movie* is clicked, the Movie Display Window continuously displays frames from the current time onward until the *Stop Movie* button is clicked or until the last frame is reached. If no data has been loaded there will be an error message.

Display Wave

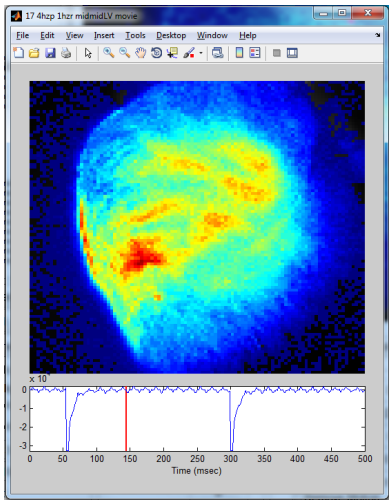
The *Display Wave* function allows users to view the fluorescent emission intensity of one pixel for the entire time duration on the right Signal Screen

1. Click Display Wave
2. Click on the heart to get the emission intensity of that pixel
3. Click and drag the pixel around to change its location
4. Repeat from step 1 to add more pixels
    a. When all signal screens have been filled, additional signals will start to fill in from the first signal screen

Export Movie

To export the movie displayed on the movie screen, first specify the time interval that you wish to export in the *Start Time* and *End Time* numeric entries in the Signal Screen panel. After the *Export Movie* button is clicked, a figure screen will pop up as shown in the figure below.

The top color plot is an exact replica of the movie screen. The bottom plot shows the data from the ECG channel and the red vertical bar indicates the current time that is displayed on the movie screen. The x-axis in the popup window is always set to start from zero.



Note: The software assumes the ECG signal comes from channel 1 of the camera raw data. The user can easily change the source of the ECG data from the **source code**. Open the *rhythm.m* file from the file list and move to line 202 that states "*handles.ecg = Data.channel1;*" and replace *Data.channel1* with your desired channel followed by a semicolon.

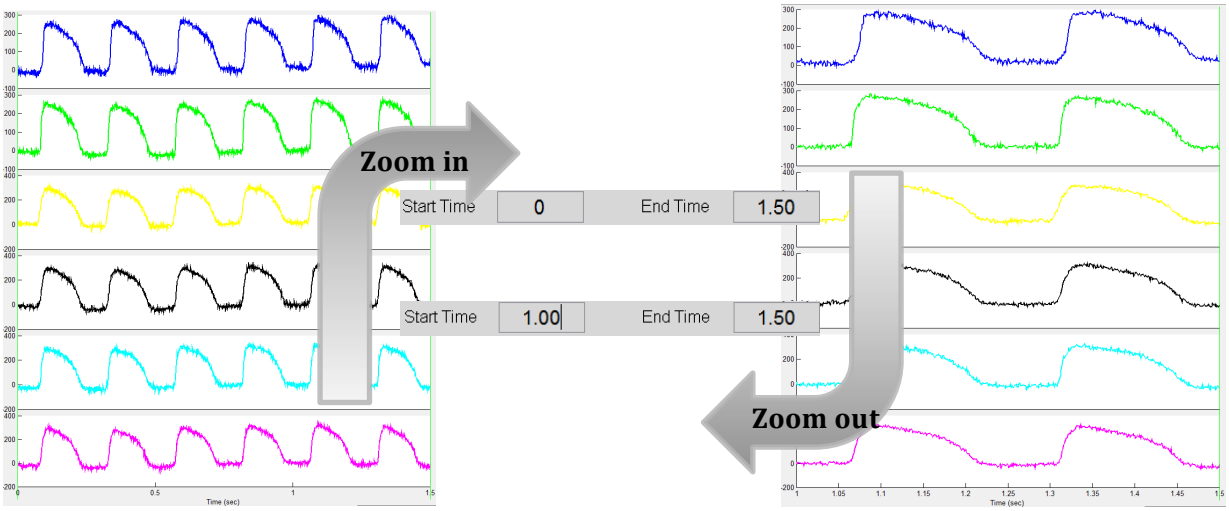
After the movie is created, the popup window will close. The movie is saved to the same folder as the data source (.rsh file) with the word "Movie" at the end of the file name.

4. Signal Screen

Signal Display Windows

The interface contains six *Signal Display Windows*, each with a unique plotting color. The x-axis shows the time scale and the y-axis shows the amplitude.

Below the six Signal Display Windows, the user may zoom in on the Signal Display Windows by specifying the *Display Start Time* and *Display End Time* zoom options.



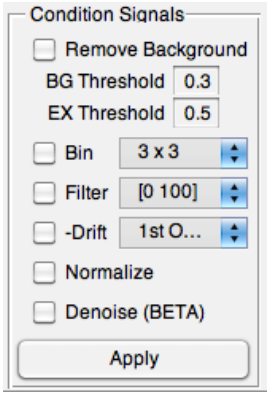
Sweep Bar

The *Sweep Bar* is the red vertical line that overlays all Signal Display Windows. The Sweep Bar indicates the time of the frame displayed on the movie screen.

Export OAP's

The *Export OAP's* function creates a popup figure of the activated Signal Display Windows. The window is an exact replicate of the Signal Screens without the sweep bar.

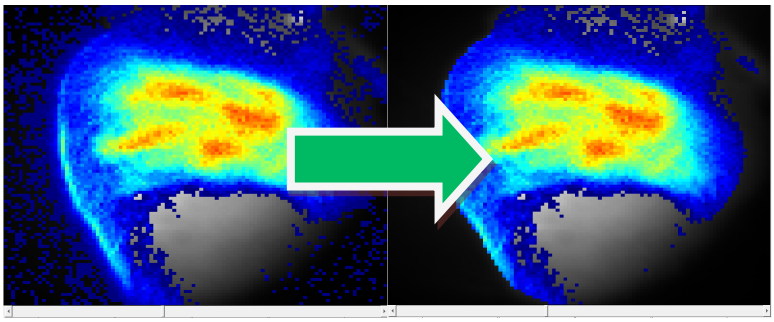
5. Condition Signals



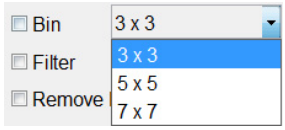
The Condition Signals panel consists of five check boxes (Remove Background, Bin, Filter, Remove Drift, and Normalize) and one Apply button. The check boxes may be selected in any combination, but the conditions will be applied in the order they appear after *Apply* is clicked. Only the original data will be processed.

Remove Background

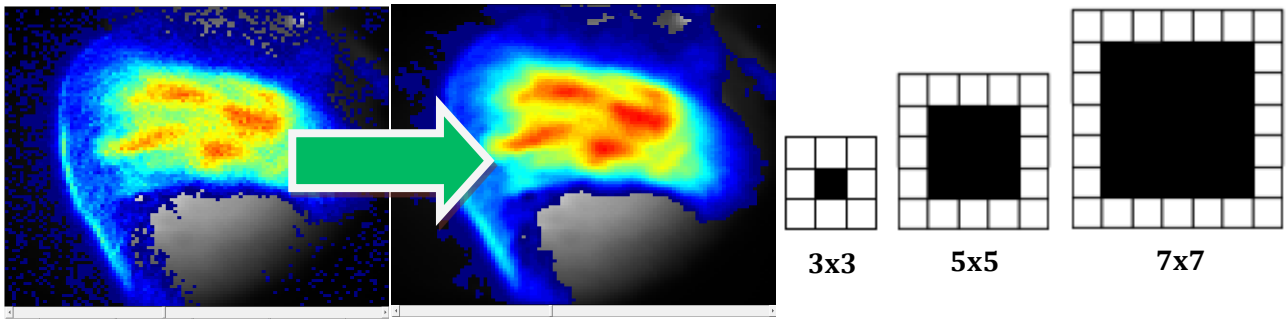
This function segments the black and white image of the tissue taken by the CMOS camera and detects pixels below the BG Threshold value. It then uses the EX Threshold to remove groups of pixels less than EX of the total image size. EX Threshold and BG Threshold values are set to values between 0 and 1. The figures below demonstrate the effect of removing the background of the original signal.



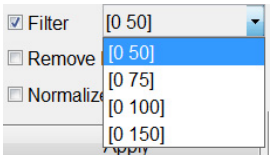
Bin



The *Bin* function sums the amplitudes of adjacent pixels, divides the sum by the number of pixels, and puts the average value in the center pixel. Increasing the bin size increases the smoothness. Decreasing the bin size increases sharpness.

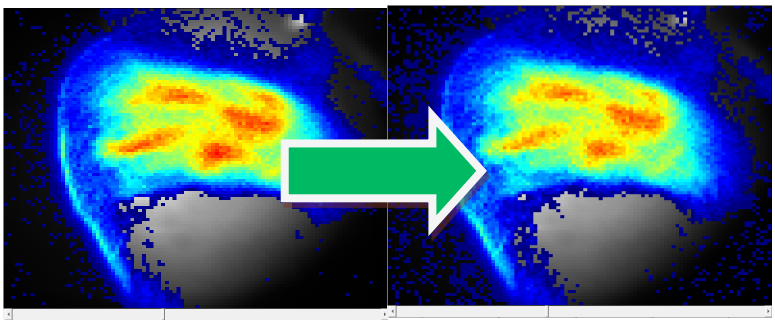


Filter



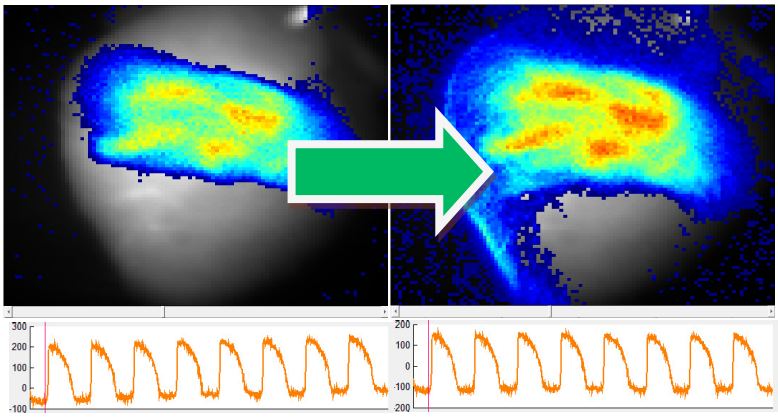
The *Filter* function operates zero-phase forward and reverse digital filtering with an n^{th} order band pass filter between a low and high passband threshold frequency. The default *Filter* is set at a 50^{th} order band pass filter between 2/Fs and 100/Fs. By clicking the drop-down menu next to *Filter*, the high band of the filter can be adjusted to 50, 75, 100 and 150 Hz. The figures below demonstrate the effect of a [0 100]

bandpass filter. All filters are implemented with the Parks-McClellan-Remez Exchange algorithm and made to be zero-phase.



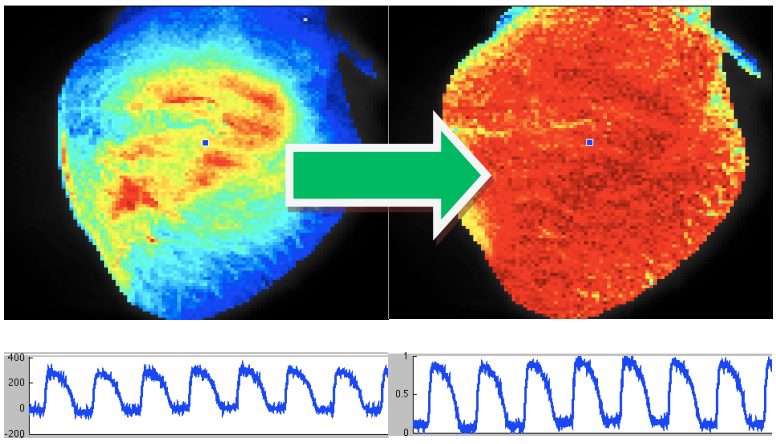
Remove Drift

The *Remove Drift* function adjusts a slanted waveform to level off the baseline.



Normalize

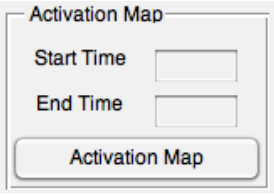
The *Normalize* function adjusts the data set to normalize data points to be within 0 and 1. Normalization should be performed once the background noise pixels are removed. Below is an example.



Denoise (Beta)

The *Denoise* function applies a Tukey window to the CMOS data to minimize edge effects in the frequency spectrum of the signal. Single-sided power spectrum is calculated per pixel. Any DC power is removed from the power spectrum. Spectral power > 50 Hz is summed per pixel since noisy pixels will contain more spectral power compared to pixels containing action potentials. An empirical threshold is hard-coded to identify and remove noisy pixels and mask the data. ** Please note that this function is still in beta and should be used carefully. This function assumes the data has not already been filtered above 50 Hz and will fill holes in the mask.

6. Activation Map

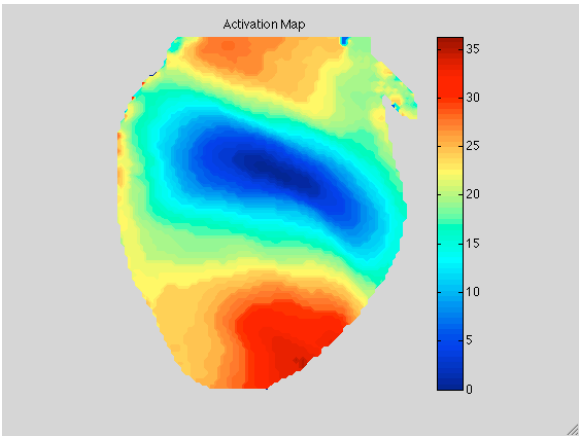


Start Time and End Time

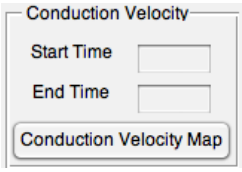
The *Start Time* and *End Time* entries restrict the data into a specific time interval (isolate an action potential), reflected by green sweep bars in the Signal Screen.

Activation Map

The *Activation Map* button plots the activation map for the time duration as specified in the *Start Time* and *End Time* entries. The function automatically conditions the signals (*Remove Background, Bin, Filter, Remove Drift,* and *Normalize*) before building the activation map. The figure represents the activation map without isolated artifacts due to noise; all but the largest connected region has been removed. The color of each pixel indicates the time at which the maximum derivative occurs.



7. Conduction Velocity



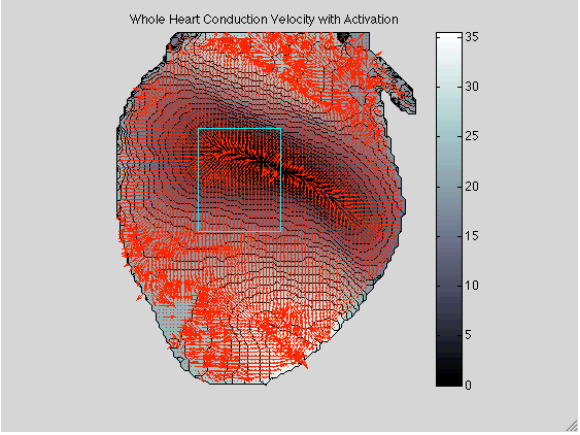
This function calculates the rate of electrical impulses for a single action potential upstroke described by one pixel. It calculates the maximum derivative of the upstroke for each pixel and fits a third-order polynomial to the activation map.

Start Time and End Time

The *Start Time* and *End Time* entries restrict the data into a specific time interval (isolate the upstroke), reflected by blue sweep bars in the Signal Screen.

Conduction Velocity Map

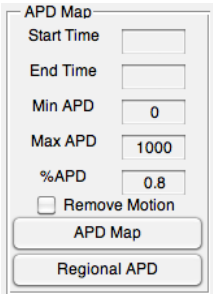
The *Conduction Velocity Map* button plots the conduction velocity map for the time duration as specified in the *Start Time* and *End Time* numeric entries. Click *Conduction Velocity Map* button. Select a rectangular area of interest on the display window with the cursor.



The figure indicates the largest connected region of processed signal with vectors represented by superimposed red arrows. The direction and magnitude of the arrows indicates the path and rate at which the tissue is being activated. The color of each pixel indicates the time at which the maximum derivative occurs.

Upon completion of the conduction velocity map, five variables are displayed in the Command Window of MATLAB. These variables report the median velocity (medV), standard deviation of the velocity (stdV), median vector angle (medAng), standard deviation of the vector angle (stdAng), and the total number of vectors described in the conduction velocity map (num_vectors).

8. Action Potential Duration Map



It is recommended to first filter the data using the Condition Signals panel before constructing an action potential duration map.

Start Time and End Time

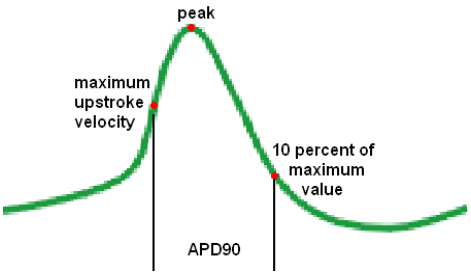
The *Start Time* and *End Time* entries restrict the data into a specific time interval (isolate an action potential), reflected by pink sweep bars in the Signal Screen.

Min APD and Max APD

The *Min APD* and *Max APD* entries allow the user to control the range of action potential duration in milliseconds, based on the data being analyzed. The entries in must be numeric values greater than zero.

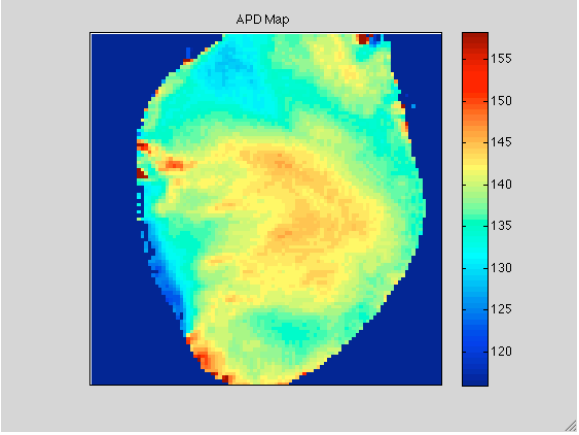
%APD

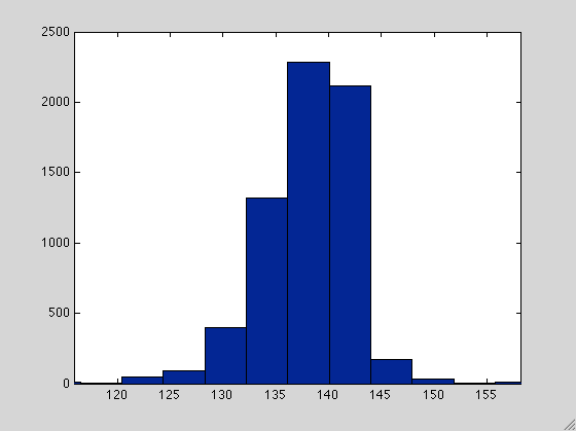
The *%APD* button allows control over analysis of the time duration, from the maximum upstroke velocity to the first point that is a specified percentage of the maximum value after the peak. The figure to the right demonstrates the action potential duration of interest for an APD_{90} map, which plots the time duration from the maximum upstroke velocity to the first point less than 10% of the maximum value after the peak.



APD (Action Potential Duration) Map

Upon selection of the *APD Map* button, two figures, resembling the ones below, pop up. The figure on the left is the APD_{90} map without artifacts; the figure on the right is a histogram reflecting the frequency of signals displayed in the map.

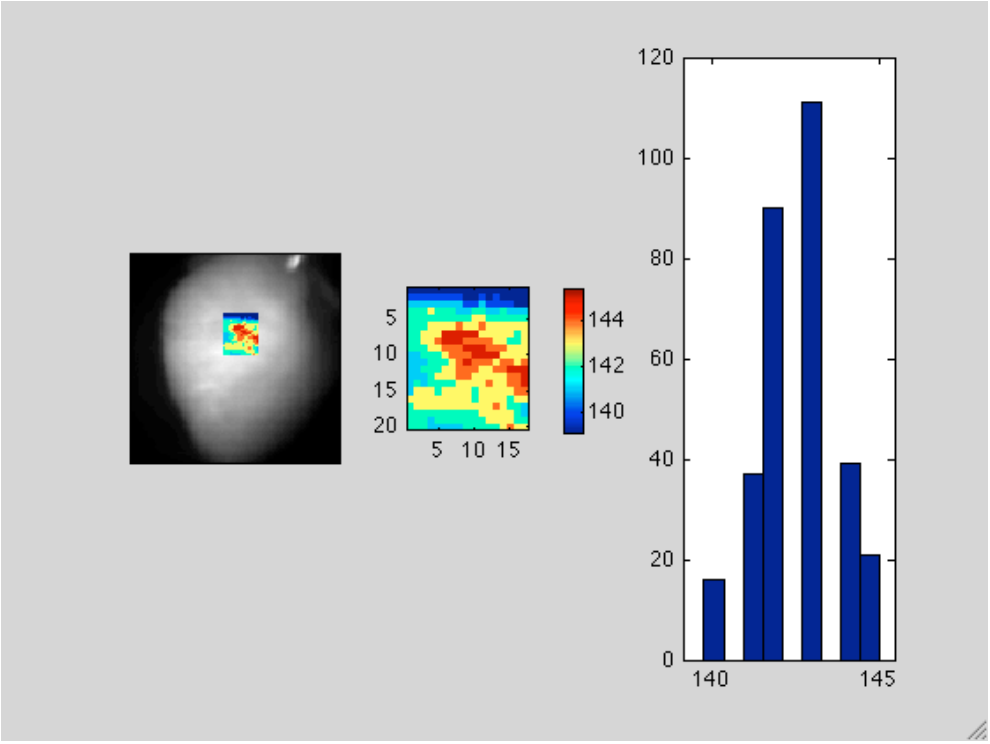




Regional APD

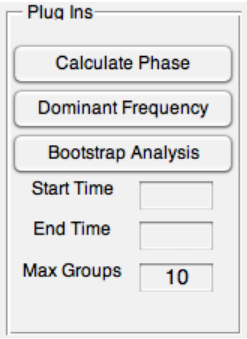
Click *Regional APD* button. Select a rectangular area of interest on the display window with the cursor. The first figure is the grayscale background of the camera's field of view with a superimposed action potential duration map calculated only for the indicated area of interest. The second figure is a magnified version of the action potential duration map, with the color of each pixel describing the time

necessary for that pixel to reach the indicated percent of the action potential duration. The third figure is a histogram displaying the frequency of action potential durations from the region of interest.



Upon completion of the conduction velocity map, two variables are displayed in the Command Window of MATLAB. These variables report the action potential duration mean (apd_mean), and the standard deviation of the action potential (apd_std).

11. Plug Ins



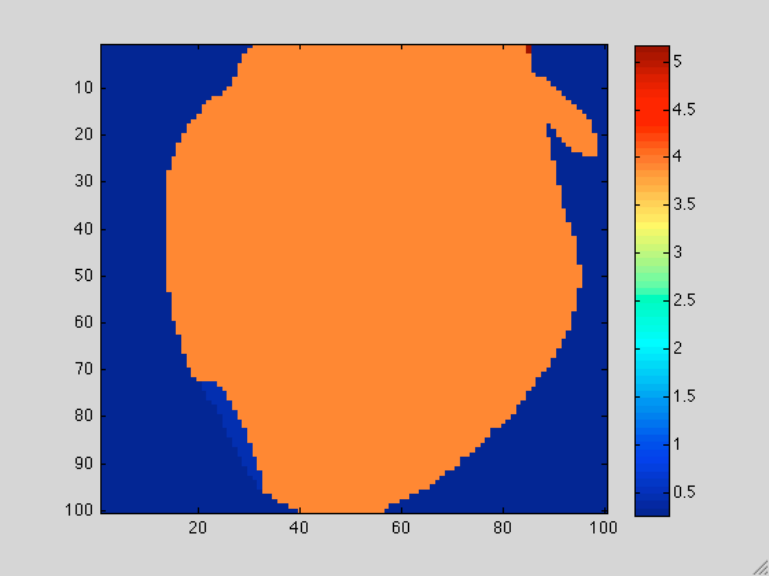
Rhythm GUI can be used widely for optical mapping data of different settings and thresholds. Change the following parameters for optimal display: Calculate Phase, Dominant Frequency, Bootstrap Analysis.

Calculate Phase

The *Calculate Phase* function provided is an optional plug-in that generates a phasemap movie in MATLAB relating action potential behavior to that of a sinusoidal wave. It is beneficial for those interested in the analysis of spatiotemporal changes and the progression of the excitation wave across the tissue.

Selection of the *Calculate Phase* button in the Plug Ins panel initiates the algorithm and popup message "*Calculating Phase Map".* Completion of the algorithm yields a new window with the MATLAB film, which can be saved by clicking File > Save As and specifying the intended destination for the film.

Dominant Frequency



The *Dominant Frequency* function provided is an optional plug-in that estimates activation rate of the entire tissue. Treating the action potential patterns as signals that can be decomposed into sine waves, the activation rate is estimated and corresponding color attributed to the largest connected region without artifacts. Selection of the *Dominant Frequency* button initiates the algorithm and popup message *"Calculating Dominant Frequency Map."* Completion of the algorithm yields a new figure as seen below.

Bootstrap Analysis

The *Bootstrap Analysis* function uses a Gaussian mixture model to determine different subpopulations of pixels in the data. Currently, this function calculates and spatially segments an APD map for the data. A user must window a single action potential similar to creating an APD map as described above using the *Start Time* and *End Time* textboxes. Additionally, a user must specify the max possible groups of subpopulations of pixels (default = 10). The Bootstrap Analysis function will then determine the most likely number of subpopulations in the data and show a spatial segmentation of the APD map. This function can be altered to analyze any parameter of interest or multiple parameters.

Appendix

This was made for the Micam Ultima System by SciMedia (http://www.scimedia.com). If different camera systems are used, the following files must be included:

- cmosData: 100 x 100 x time array of data
- BG = Background image
- ECG

Changes will also need to be made to 'Load File', 'Select Directory', and CMOSconverter.m to find and load the appropriate file extension.

Rhythm is organized as follows:

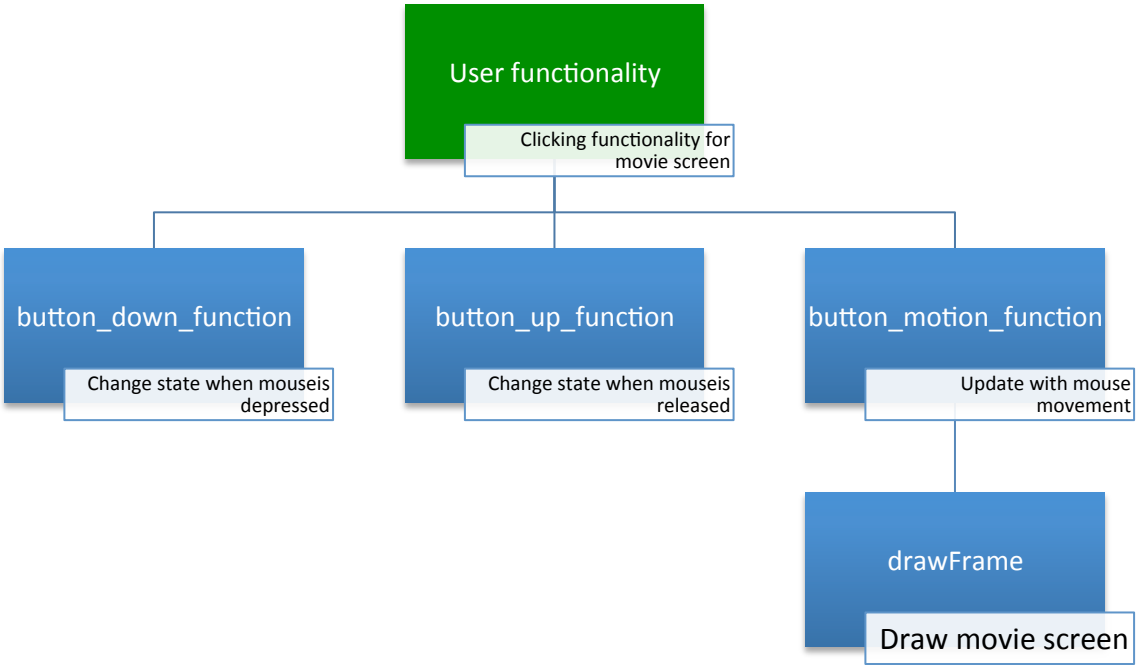
1. Visuals are created on the graphical user interface
   - `%% Create GUI structure`
2. Global variables (handles) are declared

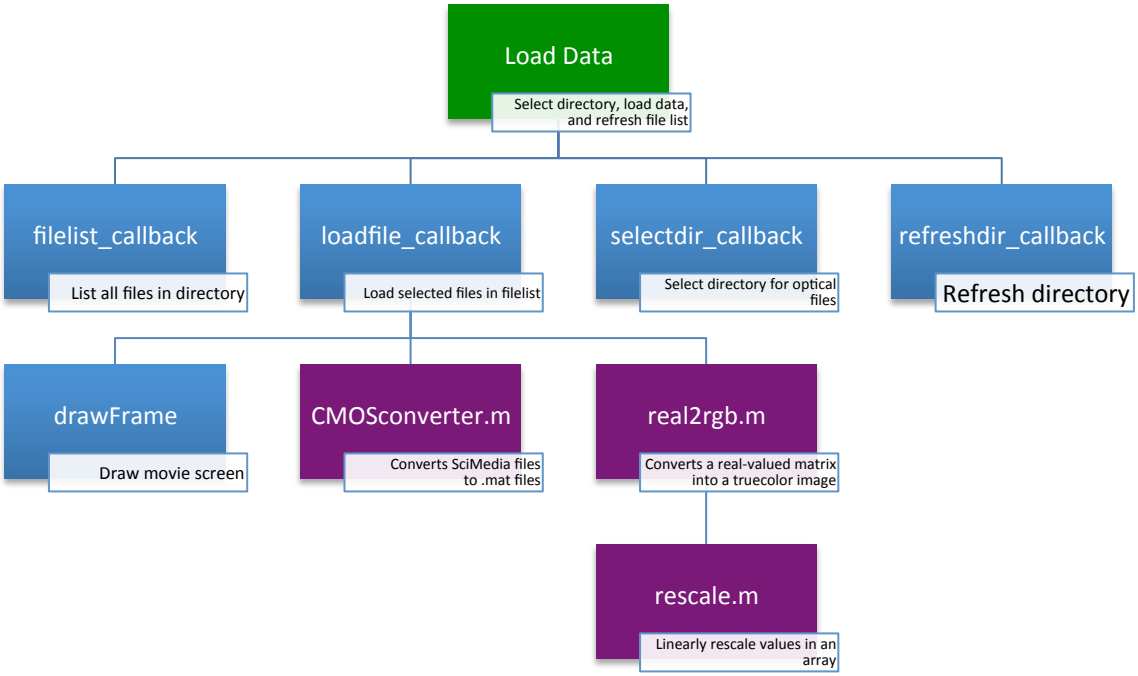
- `%% Create handles`

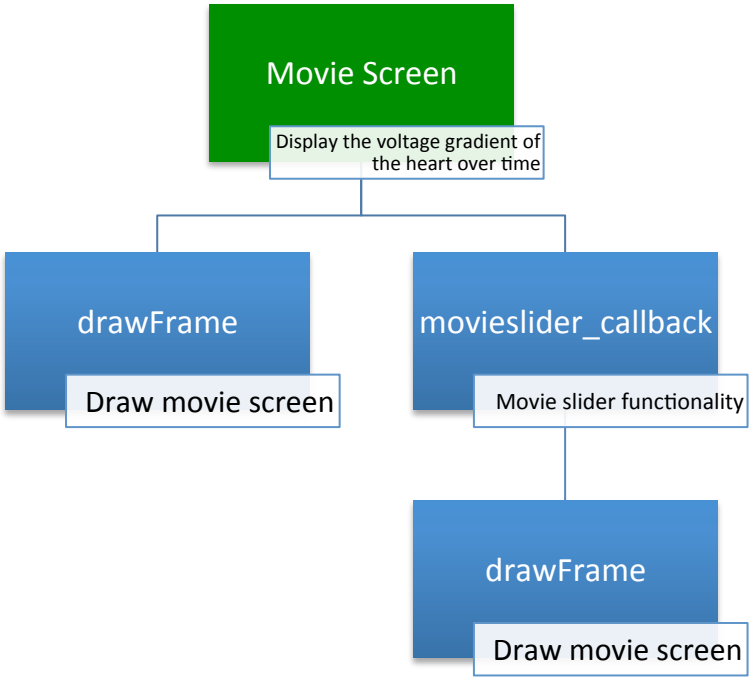
3. Callbacks (button functionality) are defined
   - `%% All Callback functions`
   - Each callback accesses different .m files which can be found in the File List to the left

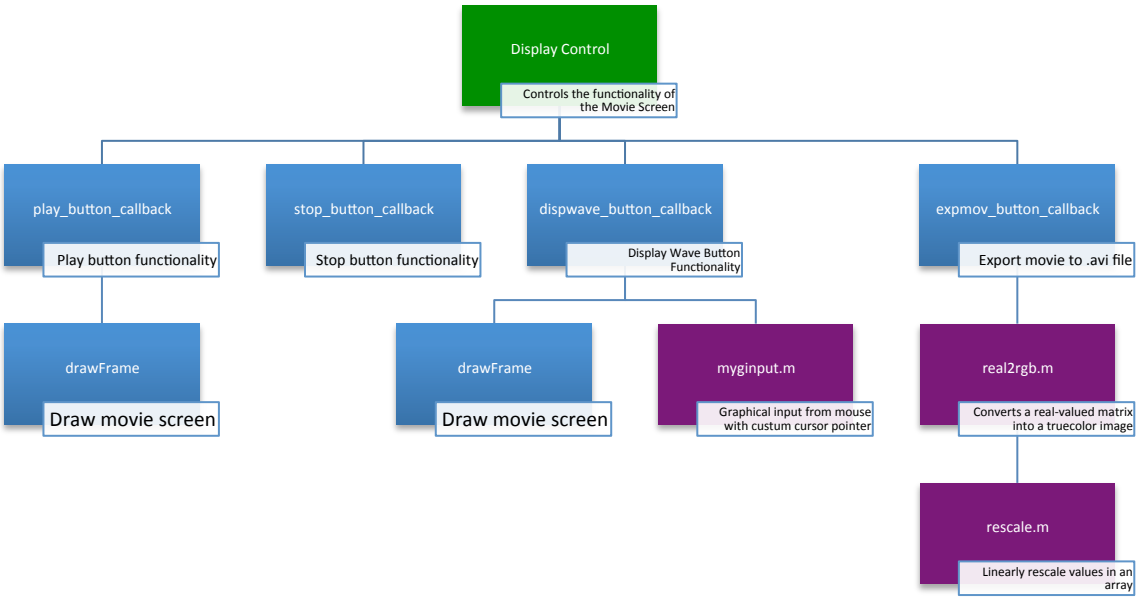
Callback organization:

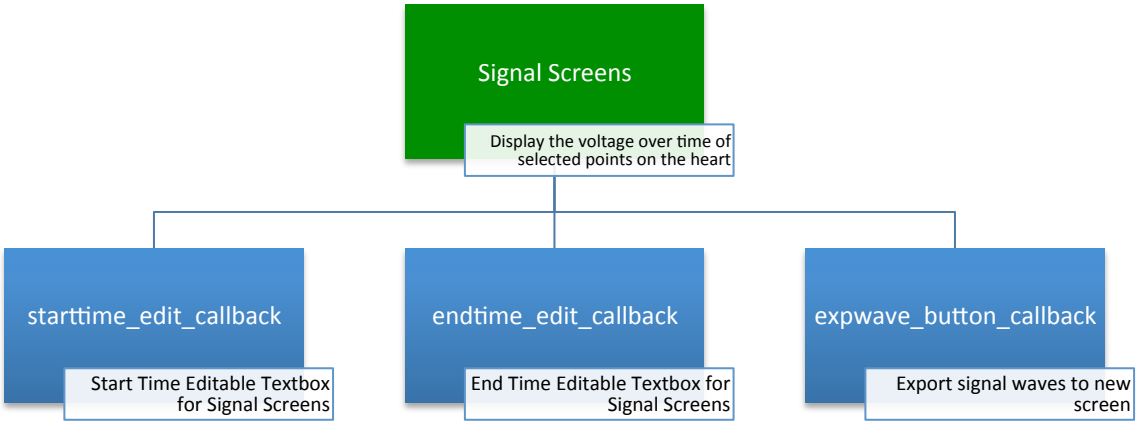
Category of functionality = green
Callback function = blue
.m file = purple

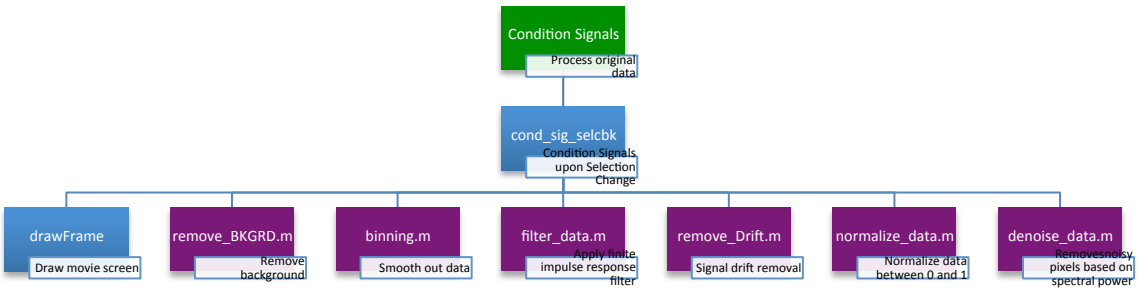


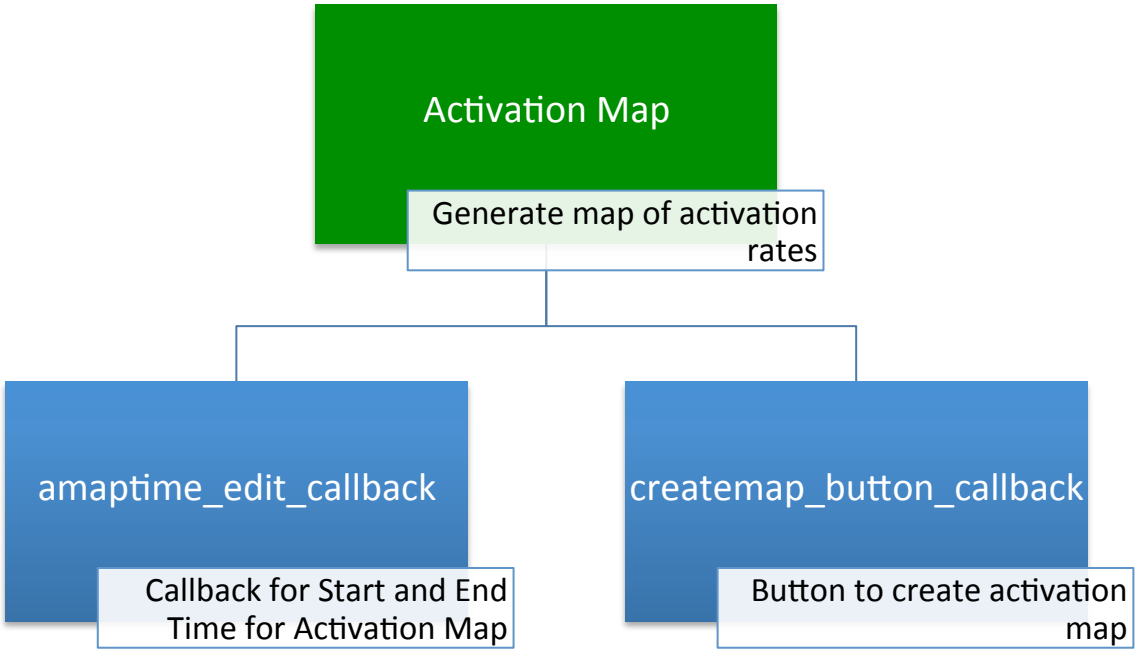

Load Data
Select directory, load data, and refresh file list
filelist_callback
List all files in directory
loadfile_callback
Load selected files in filelist
selectdir_callback
Select directory for optical files
refreshdir_callback
Refresh directory
drawFrame
Draw movie screen
CMOSconverter.m
Converts SciMedia files to .mat files
real2rgb.m
Converts a real-valued matrix into a truecolor image
rescale.m
Linearly rescale values in an array

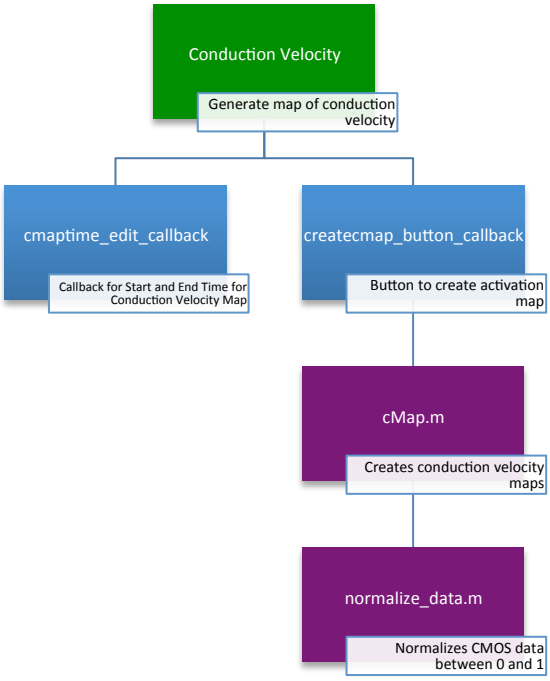

Movie Screen
Display the voltage gradient of the heart over time
drawFrame
Draw movie screen
movieslider_callback
Movie slider functionality
drawFrame
Draw movie screen

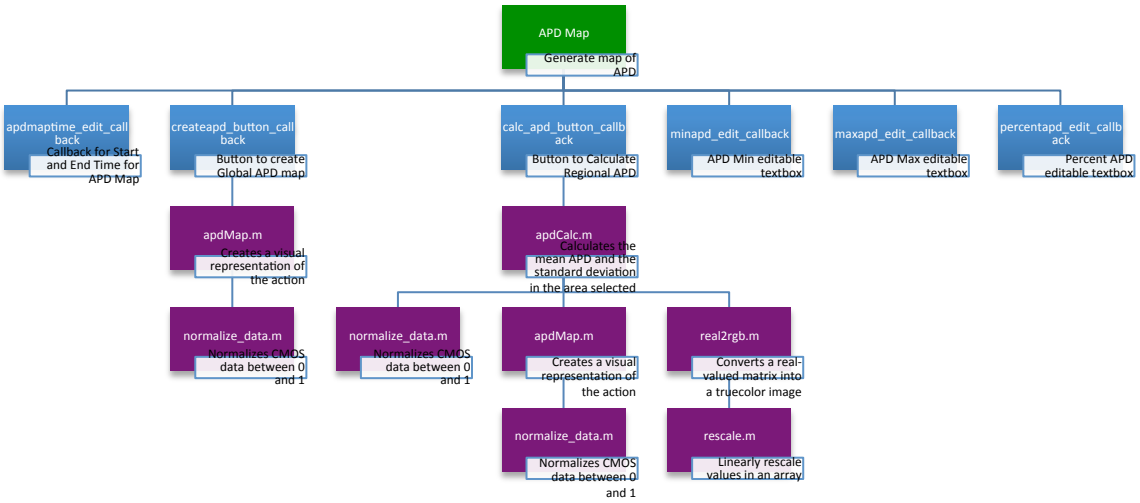

Display Control
Controls the functionality of the Movie Screen
play_button_callback
Play button functionality
stop_button_callback
Stop button functionality
dispwave_button_callback
Display Wave Button Functionality
expmov_button_callback
Export movie to .avi file
drawFrame
Draw movie screen
drawFrame
Draw movie screen
myginput.m
Graphical input from mouse with custum cursor pointer
real2rgb.m
Converts a real-valued matrix into a truecolor image
rescale.m
Linearly rescale values in an array

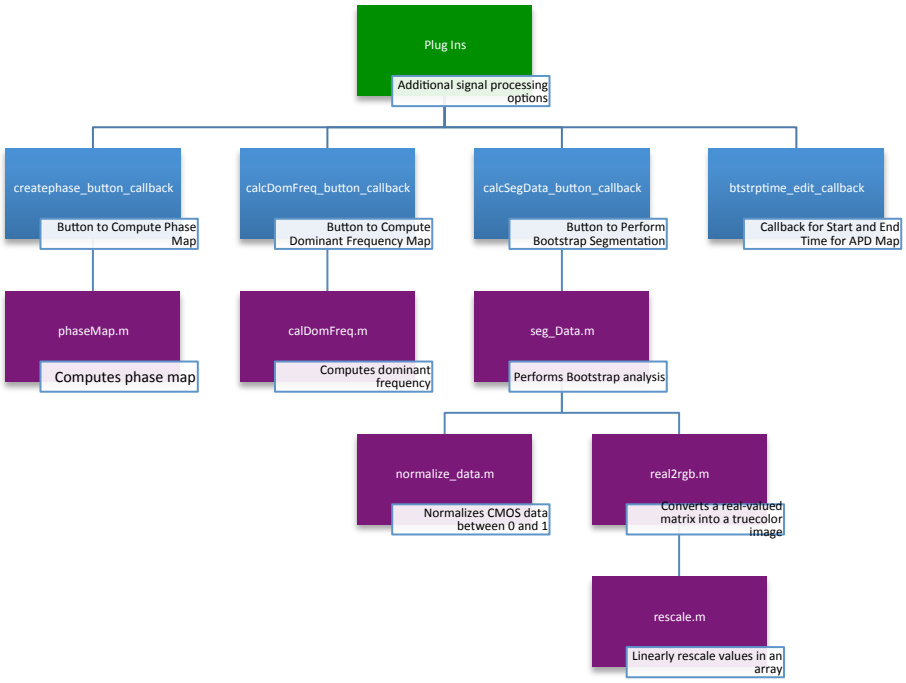

Signal Screens
Display the voltage over time of selected points on the heart
starttime_edit_callback
Start Time Editable Textbox for Signal Screens
endtime_edit_callback
End Time Editable Textbox for Signal Screens
expwave_button_callback
Export signal waves to new screen


Condition Signals
Process original data
cond_sig_selcbk
Condition Signals upon Selection Change
drawFrame
Draw movie screen
remove_BKGRD.m
Remove background
binning.m
Smooth out data
filter_data.m
Apply finite impulse response filter
remove_Drift.m
Signal drift removal
normalize_data.m
Normalize data between 0 and 1
denoise_data.m
Removes noisy pixels based on spectral power


Activation Map
Generate map of activation rates
amaptime_edit_callback
Callback for Start and End Time for Activation Map
createmap_button_callback
Button to create activation map


Conduction Velocity
Generate map of conduction velocity
cmaptime_edit_callback
Callback for Start and End Time for Conduction Velocity Map
createcmap_button_callback
Button to create activation map
cMap.m
Creates conduction velocity maps
normalize_data.m
Normalizes CMOS data between 0 and 1


APD Map
Generate map of APD
apdmaptime_edit_callback
Callback for Start and End Time for APD Map
createapd_button_callback
Button to create Global APD map
calc_apd_button_callback
Button to Calculate Regional APD
minapd_edit_callback
APD Min editable textbox
maxapd_edit_callback
APD Max editable textbox
percentapd_edit_callback
Percent APD editable textbox
apdMap.m
Creates a visual representation of the action
apdCalc.m
Calculates the mean APD and the standard deviation in the area selected
normalize_data.m
Normalizes CMOS data between 0 and 1
normalize_data.m
Normalizes CMOS data between 0 and 1
apdMap.m
Creates a visual representation of the action
real2rgb.m
Converts a real-valued matrix into a truecolor image
normalize_data.m
Normalizes CMOS data between 0 and 1
rescale.m
Linearly rescale values in an array


Plug Ins
Additional signal processing options
createphase_button_callback
Button to Compute Phase Map
calcDomFreq_button_callback
Button to Compute Dominant Frequency Map
calcSegData_button_callback
Button to Perform Bootstrap Segmentation
btstrptime_edit_callback
Callback for Start and End Time for APD Map
phaseMap.m
Computes phase map
calDomFreq.m
Computes dominant frequency
seg_Data.m
Performs Bootstrap analysis
normalize_data.m
Normalizes CMOS data between 0 and 1
real2rgb.m
Converts a real-valued matrix into a truecolor image
rescale.m
Linearly rescale values in an array