‘Zeng’ Data Analysis Package

Documentation

Contents

[1 Getting Started 2](#_Toc88220483)

[2 Converting Raw data Files to Zeng Readable Format 4](#_Toc88220484)

[2.1 The BATCH window 4](#_Toc88220485)

[3 Data Display and Manipulation 6](#_Toc88220486)

[3.1 The LOG IN window 6](#_Toc88220487)

[3.2 STRIPCHART WINDOW 8](#_Toc88220488)

[3.3 DATA INFORMATION 10](#_Toc88220489)

[3.4 WAVEFORMS WINDOW 11](#_Toc88220490)

[4 Working With Annotations 14](#_Toc88220491)

[4.1 ANNOTATION WINDOW 15](#_Toc88220492)

[5 Data Analysis 17](#_Toc88220493)

[5.1 ACTIVATION TIME ANALYSIS 17](#_Toc88220494)

[5.2 CONDUCTION VELOCITY ANALYSIS 19](#_Toc88220495)

[5.3 REPOLARIZATION TIME ANALYSIS 24](#_Toc88220496)

[5.4 PERCENT REPOLARIZATION TIME ANALYSIS 25](#_Toc88220497)

[5.5 SIGNAL AVERAGING ALGORITHM 27](#_Toc88220498)

[5.5.1 SIGNAL AVERAGING WINDOW 29](#_Toc88220499)

[5.6 ALTERNANS ALGORITHM 31](#_Toc88220500)

[5.6.1 ALTERNANS WINDOW 33](#_Toc88220501)

[5.7 AREA QUANTIFICATION WINDOW 37](#_Toc88220502)

[5.8 CUSTOM ALGORITHMS 38](#_Toc88220503)

[6 Graphical Representation of Data 39](#_Toc88220504)

[6.1 CONTOUR WINDOW 39](#_Toc88220505)

[6.1.1 VECTOR PARAMETER WINDOW 44](#_Toc88220506)

[6.1.2 AUTOCALCULATION OF VECTORS WINDOW 46](#_Toc88220507)

[6.2 MOVIE WINDOW 48](#_Toc88220508)

[6.2.1 IMAGE PROCESS LOG WINDOW 50](#_Toc88220509)

[6.2.2 DIGITAL FILTERS WINDOW 51](#_Toc88220510)

[6.2.3 DIGITAL THRESHOLDING WINDOW 52](#_Toc88220511)

[APPENDIX A. THE SKELETON.m File 53](#_Toc88220512)

[APPENDIX B. TABLE OF INTERPIXEL RESOLUTION FOR GIVEN CAMERAS/OPTICS 59](#_Toc88220513)

[APPENDIX C. CONVERTING SCIMEDIA, DUAL CAMERA DATAFILES INTO ZENG FORMAT 60](#_Toc88220514)

# 

# 1 Getting Started

This chapter tells you how to get started with the *‘Zeng’ data analysis package.*

**What you need**

To use the *‘Zeng’ data analysis package*, you should be familiar with the basic Windows functions, such as working with documents, windows, menus, dialogs and the mouse.

Microsoft Windows System Requirements:

The minimum requirements is a 5860based PC compatible computer, 64 MB of RAM, 1 MB hard disk space, Color VGA Graphics card and monitor, mouse, and Windows 95 or higher.

This program is very memory intensive. In order to adequately display and manipulate 30 seconds of 256 channel data sampled at 1 KHz, it is recommended the computer have at least 256 MB of system RAM.

The *‘Zeng’ data analysis package* requires the software package MATLAB R11 (or better) by The Mathworks Inc. Refer to the MATLAB manual for installation of this software.

**Installing *‘Zeng’ data analysis***

Drag and drop the Zeng folder from the provided CD to the C drive of your computer. The root location of Zeng will be C:\Zeng

Start MATLAB and type at the command prompt:

>> cd c:

>> cd zeng

>> zeng

This will put you in the Zeng root directory. As long as Matlab has not closed, *‘Zeng’ data analysis* can be re-launched by simply typing “zeng” at the command prompt.

Getting Started

**Installation Alternative**

Instead of typing the above lines every time you want to run the *‘Zeng’ data analysis* program, follow the steps below to create a small program to automatically launch the *‘Zeng’ data analysis package*.

Open the MATLAB editor by clicking on File -> New -> M-File

In the new window, type:

cd c:

cd zeng

zeng

Save this file as zeng.m in the root MATLAB directory.

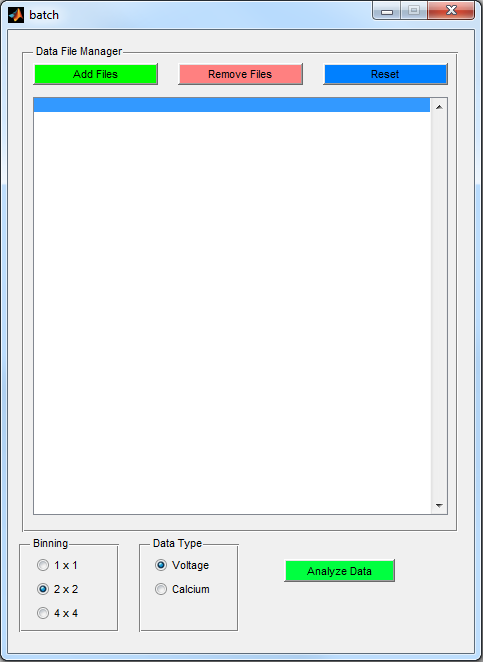
Now, the *‘Zeng’ data analysis package*  can be launched by typing “zeng” at the MATLAB command line when MATLAB is first launched, or when you simply want to restart *‘Zeng’ data analysis.*

# 2 Converting Raw data Files to Zeng Readable Format

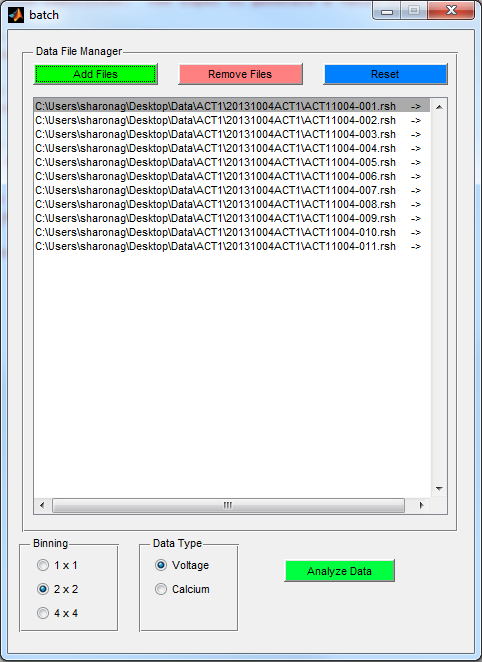
This chapter explains how to convert the raw data files from the Ultima L CMOS camera to a format that can be read by the Zeng Analysis Software. The data files required to perform this process and their locations are described here. Once converted these files will be available to be selected using the LOG IN Window as described in the “Data Display and Manipulation” chapter.

### 2.1 The BATCH window

After adding the “Zeng Analysis” folder to the current MATLAB path, type the command “batch” into the command window and press Enter. The following window will open. This window allows the user to select all the raw data files to be converted to Zeng readable format.



The “Add Files” button navigates the user to a second window where the user can browse to the location of the raw data files and select them. The default setting displays all the header files in the current location. The header files are those files that are in the RSH File format and contains information like the Gain, Sampling Rate, Stimulus parameters and any comments included during recording. The user can select the header files of all the data files that are required by clicking on each header file while holding down the control button. NOTE: All the raw data files will have to be in the same folder as the selected Header files to perform the conversion. Then click the “Open” button. This will navigate the user to another window where the user can select a location to save the converted files. Once the location has been selected click on the “Save” button. This will populate the Batch window with the selected header files as shown below.



The “Remove Files” button allows to user to remove selected files from the list of files to be converted and the “Reset” button allows the user to remove all files that are currently in the list shown in the window above.

The next step would be to determine the binning in the converted files. Select “1 X 1” if no binning is required. Select “2 X 2” if the pixels are required to be averaged using adjacent pixels that form a 2-by-2 square called a bin and select “4 X 4” ” if the pixels are required to be averaged using adjacent pixels that form a 4-by-4 bin. Binning the data reduces some noise in the recorded signal.

The final step before conversion is to select the Data Type. Select “Voltage” if the recorded optical signals are a representative of the membrane voltage obtained by using a voltage sensitive dye like Di-4-ANEPPS and select “Calcium” in case of optical signals of intracellular calcium using a dye such as Indo-1. Selecting the “Calcium” data type includes a mirror shift during conversion that is required for further “Calcium” processing.

When all the above steps are completed, click on the “Analyze Data” button to perform the conversion. A process bar appears to indicate the stage of the conversion process. At completion, the process bar disappears and the files will be ready to be selected using the LOG IN window as described in the next chapter.

REQUIREMENTS:

1. All raw data files should be in the same folder as the header files.
2. The programs batch.m and vconvert .m and the figure batch.fig should be saved in the Zeng Analysis folder which should be in the current Matlab path.

# 3 Data Display and Manipulation

This chapter explains how to open, view, and manipulate data. The data manipulation described in this chapter does not result in actual changes to the raw data file. Instead, the user will learn how look at the raw data tracings with different time resolutions and open alternative windows for viewing data. In addition, many algorithms provided in this software package produce annotations. Annotations are fiducial marks that corresponds to interesting events in time such as an action potential’s depolarization. This chapter will discuss the basics of annotations. For a more detailed discussion of annotation manipulation, refer to Chapter 4, “Working With Annotations”.

### 3.1 The LOG IN window

When the *‘Zeng’ data analysis package* is launched after typing “zeng” at the Matlab command prompt, this is the first window to open. This window will show the user all the files currently available for analysis. It will also allow the user to read important information about each data file. This extra information is created during data acquisition and saved in a header file (ie datafile.h). The header file includes information such as the pacing rate, look up table, and the comments that are actually displayed in the LOG IN window.

In order for a datafile to be recognized by the *‘Zeng’ data analysis package,* Its corresponding header file must be present in the same directory.

For example, if the user wishes to open the file:

06210100.

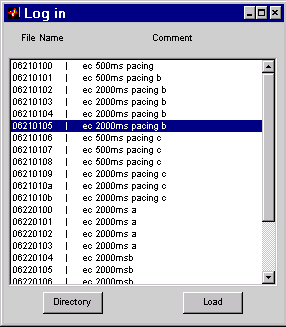
The file:

06210100.h

must be in the same directory as 06210100.

The filenames of all the files found in a directory will appear under the File Name heading. The comments extracted from the header file will appear under the Comment heading. The files can be selected by clicking on them once. Double clicking on a file will NOT open the file. A file can only be opened by clicking on the Load button after the file is highlighted in blue.

Data Display and Manipulation 3.1 LOG IN Window



**DIRECTORY BUTTON:** Allows the user to select files to be analyzed. This will open a “Select Open File” dialog box. When any file is selected the LOG IN window reads in and will display all files compatible with the Zeng format within that directory. All of these files will be displayed in the LOG IN window.

**LOAD BUTTON:** Highlight a file by clicking on it once with the mouse cursor. Click the LOAD button after highlighting a file in order to view the data. The STRIPCHART Window will open a few seconds later after clicking the Load Button. Larger data files will take longer to load. If an error occurs and the file cannot be opened for some reason, an error will be returned to the main MATLAB window.

For more details on the STRIPCHART Window, see Chapter 3.2.

The LOG IN Window will remain open throughout the data analysis session.

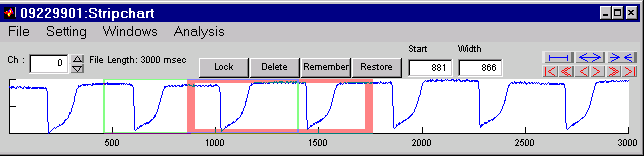
If the LOG In Window is closed, the entire *‘Zeng’ data analysis package* will terminate. Therefore, this window must remain open for the entire duration of data analysis. However, it is okay to minimize this window.

Data Display and Manipulation 3.2 STRIPCHART Window

### 3.2 STRIPCHART WINDOW

Almost every aspect of data manipulation, analysis and display is accessed through this window. In addition, this window specifies which segment of data is currently active.

Upon loading a file from the LOG IN window, this window opens up automatically. The user designates what part of the data is active by dragging a window along the signals. Multiple segments of data (ie data windows) can be selected as shown below. The green box indicates a previously highlighted section of data. The bolder red box indicates the currently active window of data. The *‘Zeng’ data analysis* routines will only operate on the data currently active in the STRIPCHART window. Data analysis display, or graphics will be shown in windows whose color corresponds to the color of the data window from which it was derived. This allows the user to analyze multiple segments of the same datafile simultatneously.



**FILE Pull Down Menu:**

*Load Analysis:* option to retrieve previously saved annotations.

*Save Analysis:* option to save annotations for later use.

*Save Analysis as:* option to save annotations under a different name for later use

*Save Raw as:* option to save the specific channel indicated in the ch: box, and the data in the active window in ASCII format for use in Excel or DeltaGraph.

*Print:* option to print the entire Stripchart window

**SETTING Pull Down Menu:**

*Unit►*

*msec:* Displays the horizontal axis in milliseconds. This is the default.

*samples:* Displays the horizontal axis in samples.

*Analysis File Saving►*

*Include Segments:* Saves the start and stop time of the segments. The start and stop time are critical for determining the location of analysis with respect to the beginning of the file. ‘*Zeng’ data analysis package* treats the first annotation as a time equal to zero. In order to preserve the activation time with respect to the beginning of the data set, this information should be saved.

*Include Information:* Appends a small header to the analysis file with information on the time and optical mapping system. This information will help the user determine which analysis corresponds to which datafile.

**WINDOWS Pull Down Menu:**

*Log:* Opens the log window. See Log Window

*Annotation:* Opens the annotation window. See Annotation Window

*Data Information:* Opens the data information window. See Data Information Window

*Movies:* Opens the movie window. See Movie Window

*Waveforms:* Opens the waveform window. See Waveform Window.

*Contour:* Opens the contour window. See Contour Window

Data Display and Manipulation 3.2 STRIPCHART Window

**ANALYSIS Pull Down Menu:**

*Activation Time:* Calls the activation time analysis algorithm.

*Repolarization Time:* Calls the repolarization time algorithm.

*Percent Repolarization:* Calls the percent repolarization algorithm.

*Action Potential Duration:* Calls the activation time and repolarization time algorithms.

*Signal Average:* Calls the signal averaging algorithm.

*Alternans:* Calls the alternans algorithm.

*Other:* Allows the user to select any M-file analysis routine compatible with the *‘Zeng’ data analysis package.*

**Ch:** This shows which channel is currently being shown in the signal averaging window. The user may scroll through the channels by using the up and down arrows, or by entering the channel number manually in the box to the immediate right of the “Ch:” label.

**FILE LENGTH:** This indicates the duration of the recording in milliseconds.

**LOCK:** This command locks the active colored window (bold line) so that it may not be moved or deleted.

**DELETE:** Delete the active window. This closes all analysis associated with the data formerly highlighted in that window. The annotations will still exist even if the active window is deleted.

**REMEMBER:** When this is selected, the window will fill in with a solid color indicating that it has been remembered. This window cannot be moved or deleted. Use this if the data highlighted is important, and you want to use the same window to look around at other parts of the data set.

**RESTORE:** Clicking this button will return the window that was Remembered to the originally remembered location and size.

**START:** This number indicates the start of the active data window in the units selected under *SETTING -> UNIT*.

**WIDTH:**  This number indicates the width of the active data window in units selected under *SETTING -> UNIT*.

**The following buttons control display of data in the STRIPCHART window.**

 : Displays the entire dataset in the Stripchart window.

 : Zooms in on the data set within the Stripchart window.

 : Zooms out on the data set within the Stripchart window.

 : Moves the zoomed to the beginning of the entire data set.

 : Scrolls left through the data set equal to the total length of the zoomed in data.

 : Incrementally scrolls left through the data set.

 : Incrementally scrolls right through the data set.

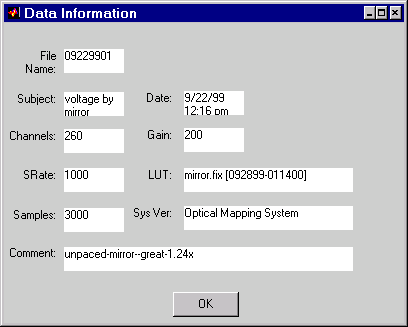
 : Scrolls right through the data set equal to the total length of the zoomed in data.

 : Moves to the end of the entire data set.

Data Display and Manipulation 3.3 DATA INFORMATION Window

### 3.3 DATA INFORMATION

The DATA INFORMATION window shows important information about datafiles. It is accessed from STRIPCHART Window by selecting Windows -> Data Inforamation.



**File Name:** Name of the file, automatically generated by the data acquisition computer.

**Subject:** Manually entered by the user at the time of data collection.

**Date:** Date of data collection automatically determined by the data collection computer at the time of acquisition.

**Channels:** Automatically generated number of channels recorded from during an experiment.

**Gain:** Amplification factor of the signal. Could be set to a required number in the data acquisition program.

**SRate:** Sampling rate (in Hz) set manually by the user during the time of data acquisition.

**LUT:** Standard Look Up Table, automatically loaded by the data acquisition program.

**Samples:** Number of samples that was saved in a given period of time during data acquisition; manually put in by the user.

**Sys Ver:** System Version—automatically generated by data acquisition program.

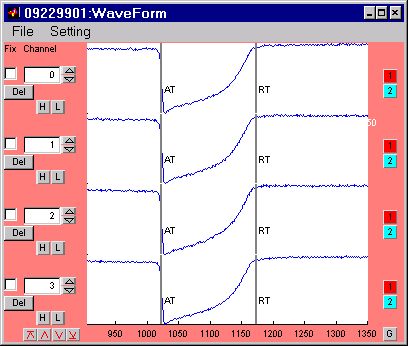
**Comment:** To help specify the data acquired at the time of acquisition, the user can add any comment in this field.

Data Display and Manipulation 3.4 WAVEFORMS Window

### 3.4 WAVEFORMS WINDOW

This window is called from the STRIPCHART Window under the Window pull down menu. This window is useful for viewing multiple channels of data at one time. Select “No. of Waveforms” under the SETTING pull down menu to select how many channels of data are displayed simultaneously.

It is also useful for viewing, manipulating annotations and making measurements with the electronic calipers. The user can left click the mouse on any annotation in this window. As long as the left mouse button is held down, the user can drag the annotation to any site of interest. This is a useful feature for correcting automatically detected annotations. Users should look at all the annotations in this window after an analysis program has run to ensure annotations were correctly assigned.



**FILE Pull Down Menu:**

*Save Waveforms:* Option to save waveforms as an ASCII delimited file to be read by Excel or DeltaGraph.

*Print:* Option to print the entire Waveform Window

*Print Setup:* Option to set printer attributes before printing the entire Waveform window.

*Export►*

*jpg:*Exports the Waveform window as a jpeg image.

*pict:* Exports the Waveform window as a pict image.

**SETTING Pull Down Menu:**

*Filters:*Option to set the high and low pass filter characteristics. These values are used when clicking on the H or L buttons below each channel

*Inverting YAxis:* Option to invert the data along the y-axis. Optical action potentials appear upside down. Select this option to have the signals appear right side up.

*No.* *of Waveforms:* Option to select how many data channels appear in the Waveform window.

*Fonts:* Option to select how text is formatted in the Waveform window.

**FIX:** Select this option to “lock” the channel. When scrolling up and down through the channels the position of all “locked” channels in the WAVEFORMS window will not change.

Data Display and Manipulation 3.4 WAVEFORMS Window

**CHANNEL:**  This indicates the current channel being shown to the right of this option. The user may manually enter a channel or scroll through them with the up and down arrows to the right of the edit box.

**H:** When this option is selected, the signal in that channel will be high pass filtered. The filter specifications are set in the Setting pull down menu under Filters.

**L:** When this option is selected, the signal in that channel will be low pass filtered. The filter specifications are set in the Setting pull down menu under Filters.

**DEL:** Selecting this option removes the current channel from display in the Waveforms window.

 : This button scrolls to the last channels in the data set. For the example window above, clicking on this button will display the signals for channels 256, 256, 258 and 259.

 : This button scrolls to the next set of channels. For the example window avobe, clicking on this button will display the signals for channels 4, 5, 6, and 7.

 : This button scrolls to the previous set of channels.

 : This button scrolls to the first channels in the data set.

**Electronic Calipers**

The buttons below have been provided to measure intervals between events as well as amplitudes of signals. These can be used to quickly ascertain an event of interest without running data analysis on the entire datafile, or to validate the results of previous data analysis.

 : Places a red line in the corresponding channel. In addition, the x and y values of that line are displayed at the bottom of the signal.

 Places a blue line in the corresponding channel. In addition, the x and y values of that line are displayed at the bottom of the signal.

In the event that both the 1 and 2 button are depressed, the dx and dy values are displayed, where dx corresponds to the difference in the x values of annotation 1 and 2. The dy value corresponds to the difference in the y values of annotations 1 and 2.

: Places a large black line in every displayed channel. When this line is moved, the change is reflected globally and moves for every channel. When the 1 and 2 button are depressed, a dx1 and dx2 will appear indicating the distance from the black line.

# 4 Working With Annotations

The *‘Zeng’ data analysis package* uses annotations to mark the timing of events of interest. Annotations can be generated by automatic algorithms, created manually, deleted, or modified and edited. This chapter will discuss how the *‘Zeng’ data analysis package* utilizes annotations, and how the user can edit annotations. Annotations can be automatically with algorithms such as ACTIVATION TIME , REPOLARIZATION TIME, and PERCENT REPOLARIZATION analysis, Chapters 5.1, 5.2 and 5.3.

Annotations can be directly manipulated using the WAVEFORMS window. The user can manually correct annotations in the WAVEFORMS window by left clicking with the mouse on an annotation and drag it a new time point.

**Definitions:**

Global Annotation: An annotation that applies to every channel in a data set

Local Annotation: An annotation that applies to data in one channel.

A global annotation creates the indenticale annotation in every channel of the data set. When a global annotation is modified in one channel, it is modified similarly in all channels. The converse is not true. Local annotations can be manipulated without affecting global annotations.

**Annotation Practice**

Open a data file using the *‘Zeng’ data analysis package.* In the STRIPCHART window, select the pull down menu:

Analysis -> Activation Time. Hit the “run” button. This analysis window will be explained in further detail in Chapter 5.1, ACTIVATION TIME analysis. For now, it is not important how it assigns activation time.

The ACTIVATION TIME analysis runs and a red status bar indicates the progress of the algorithm. When it is done, the status bar disappears. Annotations cannot be seen in the STRIPCHART window.

Select Windows -> Wave forms from the STRIPCHART pull down menus. This will open the WAVEFORMS window as described in Chapter 3.4.

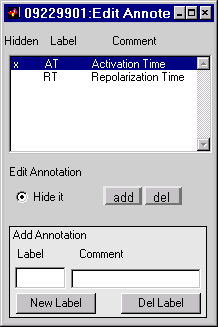
If the data set is sufficiently free of noise, the user should see gray vertical lines at the sharpest vertical deflections in the signals. The letters ‘AT’ should appear next to the gray line. Look at the window provided in Chapter 3.4. This window has both AT and RT lines corresponding to activation and repolarization respectively.

Try clicking on the gray lines within the WAVEFORMS window and dragging them around to different parts of the signals. When the left mouse button is released, notice that the gray line remains where it was last moved. You have just manually edited your first annotation.

Working With Annotations 4.1 ANNOTATION Window

### 4.1 ANNOTATION WINDOW

Whenever an annotation is created or loaded, the active annotations are displayed in this window. This window is for organizing and manipulating annotations. The changes made here are reflected in the WAVEFORMS window (Chapter 3.4).



**HIDDEN:** A hidden annotation is denoted with an “x” in this field. The annotation exists but will not be shown in the WAVEFORMS window.

**LABEL:** Annotations in the WAVEFORMS window have labels corresponding to this field in the ANNOTATIONS window.

**COMMENT:** This field offers a more detailed explanation of the annotations and labels appearing in the WAVEFORMS window.

**HIDE IT:** Selecting this options hides the currently highlighted annotation. These annotations will not appear in the WAVEFORMS window. The annotations still exist and can be recalled by de-selecting this option.

**ADD:** When this is selected, the user may click in the WAVEFORMS window in order to add the currently highlighted annotation.

**DEL:**  When this is selected, the user may click on annotations in the WAVEFORMS window to delete them.

**ADD ANNOTATION:** This option allows the user to create and define their own annotations.

**LABEL:** This is the label the user-defined annotation will have.

**COMMENT:** This is the comment the user-defined annotation will have.

**NEW LABEL:** This option creates the annotations defined in the ADD ANNOTATION, LABEL, and COMMENT fields.

**DEL LABEL:** This option deletes the currently highlighted annotations from the Waveform window and the Annotation window.

# 5 Data Analysis

Up to now, much time has been spent discussing the manual manipulation of annotations. Several analysis programs were developed which generate the most annotations. These programs were developed to be used with many different paradigms. As such, they all require the user to tweak a set of parameters. This user interaction allows the programs to perform their specified tasks on varying data sets.

The default set of parameters in all the following windows were considered to be the most robust for guinea pig and canine ventricular action potentials. With a user input, each algorithm can be used with data from human ECG to chick embryonic action potentials. The power of these tool lies in the ability of the user to create a unique set of variables that can be saved and re-used in subsequent data analysis sessions.

Each analysis program is accessed from the STRIPCHART window using the *Analysis* pull down menu. The algorithms discussed below are included in the *Analysis* pull down menu since they are the most commonly used tools. However, the user may create their own analysis programs and access them through the *Other* option in the *Analysis* pull down menu.

Each analysis algorithm returns a channel number, time of annotation, annotation label, and an annotation name to the ANNOTATION window (Chapter 4.1)

### 5.1 ACTIVATION TIME ANALYSIS

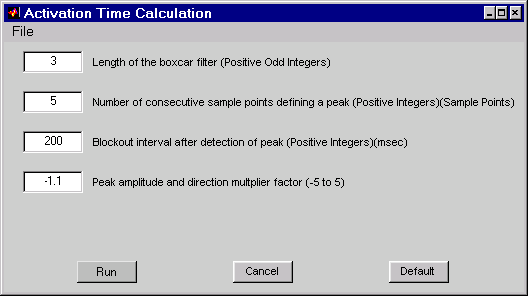
**Purpose:** This program is used to determine the Activation Times (AT) from cardiac action potentials in the WAVEFORMS window.

**Algorithm:** The program takes the first derivative of signals (Figure A). The optical action potential is depicted in red, and the first derivative is in blue. The first derivative is smoothed with a boxcar filter in order to remove high frequency noise. The program then searches for points that exceed a given threshold (dotted black line) to determine a peak. Finally, if the number of consecutive points in the peak is greater than or equal to a specified number, the maximum of that peak is considered a true peak. The program returns all the true peaks for the signal and considers that an activation time (solid black line).

Data Analysis 5.1 ACTIVATION TIME Analysis



Figure A: The optical action potential is red. The first derivative is blue. The black dotted line represents a threshold defined by the PEAK AMPLITUDE AND DIRECTION MULTIPLYING FACTOR.



RUN BUTTON: Runs the algorithm to generate Activation Times.

CANCEL BUTTON: Cancels the command and closes the analysis.

DEFAULT BUTTON: Reverts all the numerical choices in the window to their default settings.

FILE Pull Down Menu: *Save Parameters*: option to save customized parameters for later use.

*Load Parameters*: option to retrieve previously saved parameters.

The following four parameters are referred to as Peak Detection Parameters in the rest of the manual.

**Length of the boxcar filter (Positive Odd Integers):** This is a variable that specifies the length of the boxcar filter. It is a convolution of boxcar waveform with the signal, i.e., smoothing of the signal. Noisy data may require larger boxcar filtering numbers in order to determine the peak. Make sure to use odd positive integers (otherwise, the signal might get distorted).

**Number of consecutive sample points defining a peak (Positive Integers):** A peak should exceed some threshold for the specified number of consecutive points to be considered a valid peak. Higher numbers for this parameter make the peak detection algorithm less sensitive but more specific. Therefore, fewer peaks will be returned with high numbers. This parameter is used to reduce the effects of outlier points such as noise.

**Blockout interval after detection of peak (Positive Integers):** This is a variable that specifies a time interval (in milliseconds) for the algorithm to start looking for another peak after the detection of the first peak. The blockout interval must be large enough to identify only one peak per cycle. However, the blockout window must be shorter than the cardiac cycle length, or the second peak in the cycle will be missed.

**Peak amplitude and direction multiplier factor (-5 to 5):** The sign of the integer entered determines the polarity of the peak. Inverted signals require negative thresholds. The program thus looks for points below the threshold for prospective peaks. Upright signals require positive thresholds. In this case, the program looks for points above the threshold for prospective peaks. Points that exceed the standard deviation multiplied by this parameter are considered eligible peaks. Numbers closer to zero are less specific but more sensitive. Noisy data will require higher numbers.

Data Analysis 5.2 CONDUCTION VELOCITY Analysis

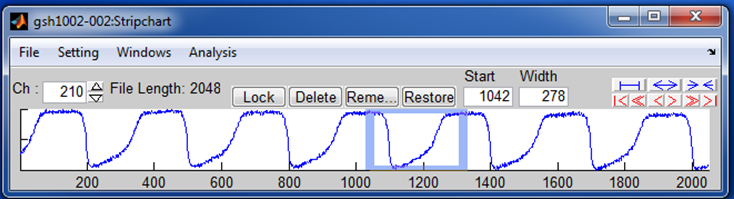
### 5.2 CONDUCTION VELOCITY ANALYSIS

This document describes the standardized procedure for analyzing conduction velocity from maps of optical action potentials. All members of the Poelzing lab should follow these procedures such that conduction velocity is analyzed and results are reported in a consistent fashion amongst the various lab members.

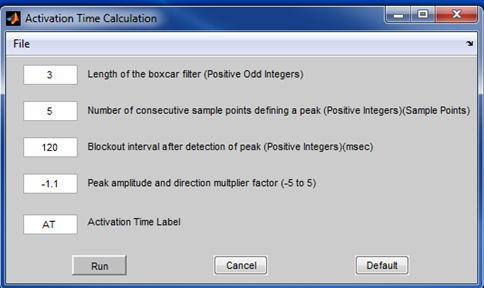
Before starting to analyze conduction velocity for the first time, new users should run the analysis on the conduction velocity training files located on the L:Drive/Data/Training Files. Conduction velocity values should be in line with historical values for these files before the new user proceeds with analyzing novel data.

**Steps**

1. In the Zeng stripchart create a window spanning a single action potential.



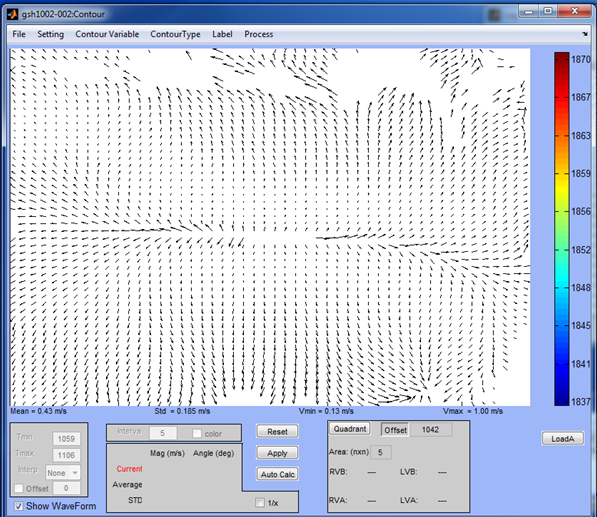
1. Under the “Analysis” tab, run the “Activation Time” analysis .m file using the following parameter settings:



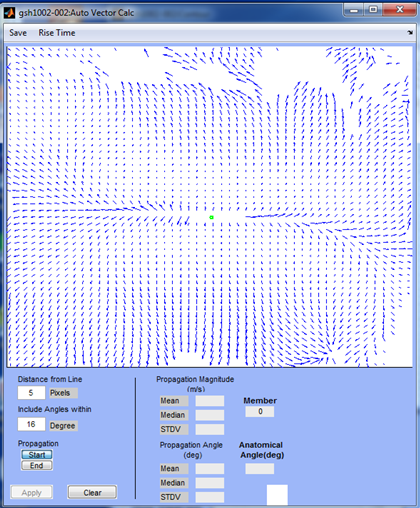
1. Under the “Windows” tab, open a “Contour” window.
2. Under “Contour Variable,” select “Activation Time” and deselect “Channels” under the “Label” tab.
3. Under “Contour Type” select “Vectors.”
4. Go to “Vector parameters” under the “Setting” tab and enter the appropriate distance between mapping sites (i.e. pixel resolution) for the given camera system, magnification and binning according to the table below (use default settings for the remaining parameters) and then “Apply” and then “OK.”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | | **MiCAM Ultima** | **MiCAMHS02-CMOS** |
| Sensor size (mm2) | | | 10 x 10 | 5.76x4.8 |
| Full array size (pixels) | | | 100 x 100 | 96 x 80 (<1.2ms/frame) |
|  | | |  | 192 x 160 (≥1.2ms/frame) |
| Active array size (pixels) | | | 100 x 100 | 92 x 80 (<1.2ms/frame) |
|  | | |  | 188 x 160 (≥1.2ms/frame) |
| Pixel size (µm2) | | | 100 x 100 | 60 x 60 (for 92x80) |
|  | | |  | 30 x 30 (for 188x160) |
| ***PIXEL RESOLUTION (mm)*** | | | | |
| Magnif (front/back lens) | | | 1.0X (1.0/1.0) | 1.25X (=2.0/1.6) |
| FOV (mm2) | | | 10 x 10 | 4.42 x 3.84 |
|  | Binning | 1x1 | 0.100 | 0.048 |
|  |  | 2x2 | 0.200 | 0.096 |
|  |  | 3x3 | 0.300 | 0.144 |
| Magnif (front/back lens) | | | 0.63X (0.63/1.0) | 0.63X (1.0/1.6) |
| FOV (mm2) | | | 15.9 x 15.9 | 8.83 x 7.68 |
|  | Binning | 1x1 | 0.159 | 0.096 |
|  |  | 2x2 | 0.318 | 0.192 |
|  |  | 3x3 | 0.477 | 0.288 |

1. At the bottom of the Contour Window, press the “AutoCalc” button.

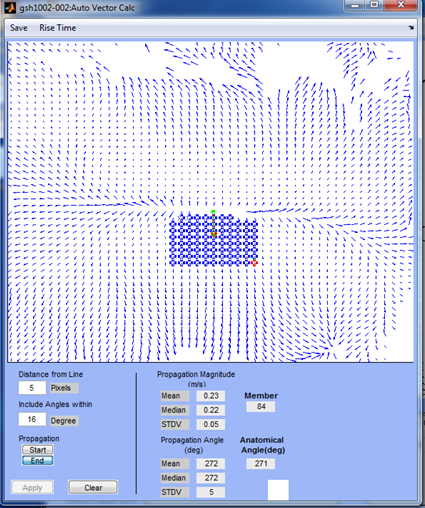


1. In the new “Auto Vector Calc” window, input/verify the following parameters:
   1. Distance from Line = 5 Pixels
   2. Include Angles within = 16 Degrees
2. Choose the direction of propagation on which you want to measure conduction velocity, for epicardial mapping of Langendorff perfused guinea pig hearts typically:
   1. Transverse = up and down
   2. Longitudinal = left and right
3. Click the Propagation “Start” button and then click on the center of the pacing site to select the starting point for the user defined line denoting the direction/path of propagation/conduction.



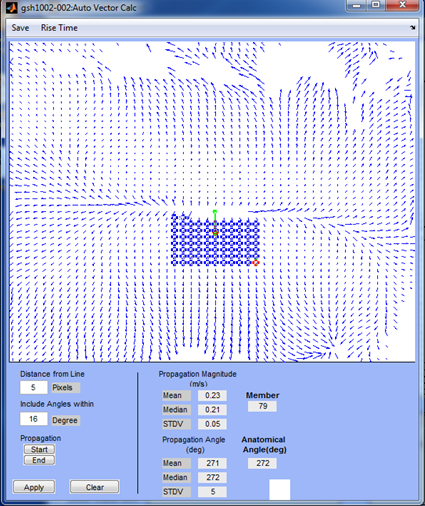
1. Once the starting point is properly positioned, click the Propagation “End” button. Click on the contour map, placing the end point such that the start and end point form a line in the direction of propagation in which the user would like to measure conduction velocity. Limit the selected vectors to the first X “layers” of vectors, corresponding to a distance of ~0.25 cm from the pacing site. See the table below for the number of vector layers (X) for the given camera system, magnification and binning:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | | **MiCAM Ultima** | | **MiCAMHS02-CMOS** | |
| ***# of layers of vectors to include in CV measurements*** | | | | | |
| **Magnif (front/back lens)** | | **1.0X (1.0/1.0)** | | **1.25X (=2.0/1.6)** | |
|  | | **Trans** | **Long** | **Trans** | **Long** |
| **Binning** | **1x1** | 25 | 50 | Use full array | |
| **2x2** | 13 | 26 |
| **3x3** | 8 | 17 |
| **Magnif (front/back lens)** | | **0.63X (0.63/1.0)** | | **0.63X (1.0/1.6)** | |
|  | | **Trans** | **Long** | **Trans** | **Long** |
| **Binning** | **1x1** | 16 | 32 | 26 | 52 |
| **2x2** | 8 | 16 | 13 | 26 |
| **3x3** | 5 | 10 | 9 | 17 |



**8 layers**

1. To remove the pacing artifact, exclude the first “layer” of vectors adjacent to the pacing site by deselecting the “End” button and individually clicking on each vector so that they disappear.



**First layer removed**

1. Mean, median, standard deviation, number of members, propagation angle, and anatomical angle are displayed and output to the command window.
2. To repeat the analysis in other directions (e.g. up, down, left, right), click the “Clear” button to reset the window and repeat steps 9-13 in the direction(s) of interest.
3. If the measurement yields too few “member” vectors, the results may be unreliable. For this reason, the minimum number of vectors for which a measurements should be used is:
   1. Transverse = 25
   2. Longitudinal = 5
4. For additional details on vector parameters/analysis, see sections 6.1.1 and 6.1.2 in the Zeng Analysis Software Documentation.

Data Analysis 5.3 REPOLARIZATION TIME Analysis

### 5.3 REPOLARIZATION TIME ANALYSIS

**Purpose:** This program is used to determine Repolarization Times (RT) on the signals in the WAVEFORMS window.

**Algorithm:** The program takes the second derivative of signals (Figure A). The optical signal is in red, and the second derivative is in blue. The minimum of the second derivative, after activation time is determined, is considered the Repolarization Time (solid black line). NOTE: if the algorithm is unable to find AT, it will not be able to find RT.

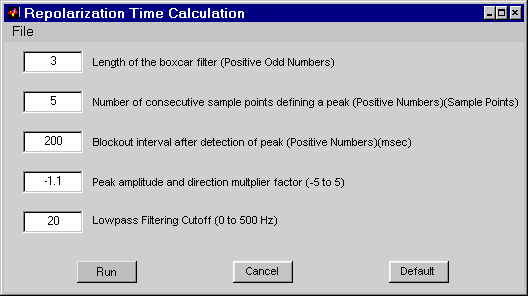


Figure A: The optical action potential is red. The second derivative is blue.

RUN BUTTON: Runs the algorithm to generate Repolarization Times.

CANCEL BUTTON: Cancels the command and closes the analysis.

DEFAULT BUTTON: Reverts all the numerical choices in the window to their default settings.

FILE Pull Down Menu: *Save parameters:* option to save customized parameters for later use.

*Load parameters:* option to retrieve previously saved parameters.

The first four parameters are Peak Detection Parameters and are identical to those of Activation Time analysis. Refer to the section IV. Activation Time Analysis for more details.

**Low Pass Filtering Cutoff (0 to 500 Hz):** Since detecting of RT requires additional filtering, this filter passes all frequencies up to the specified one. Frequency of 500 Hz is equivalent to no filtering at all, and frequency of 0 Hz would return a straight line. Nosier data requires lower cutoffs to eliminate their frequency components from the signal.

Data Analysis 5.4 PERCENT REPOLARIZATION TIME Analysis

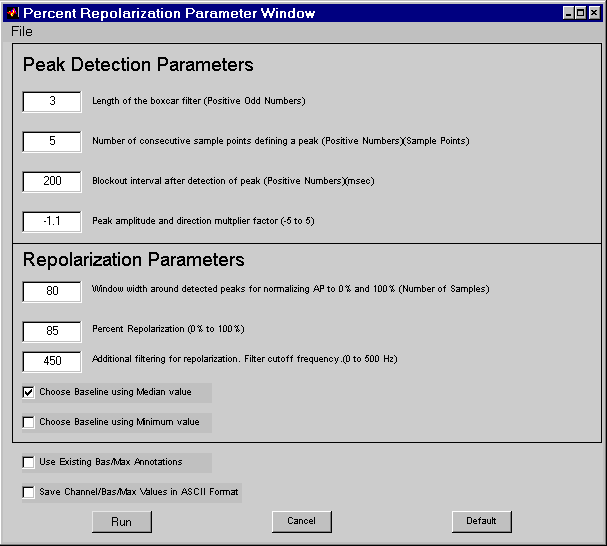
### 5.4 PERCENT REPOLARIZATION TIME ANALYSIS

**Purpose:** This calculation is used to determine Repolarization Times (RT) based on a percent of the signals amplitude in the WAVEFORMS Window.



**Algorithm:** The program finds the activation time of the signal and centers a window of a user specified width centered on the activation time (Black box on the red Action Potential). The maximum of the signal is found in the right hand side of the box and the baseline of the signal is either determined by the median or minimum value in the left hand side of the window. Full repolarization (100% repolarization) is when the signal has reached the baseline.

Figure A: The optical action potential is red. The black box indicates the window which the program looks for the maximum and baseline values.



Data Analysis 5.4 PERCENT REPOLARIZATION TIME Analysis

RUN BUTTON: Runs the algorithm to generate Repolarization Times.

CANCEL BUTTON: Cancels the command and closes the analysis.

DEFAULT BUTTON: Reverts all the numerical choices in the window to their default settings.

FILE Pull Down Menu: *Save parameters:* option to save customized parameters for later use.

*Load parameters:* option to retrieve previously saved parameters.

The first four parameters are Peak Detection Parameters and are identical to those of Activation Time analysis. Refer to the section IV. Activation Time Analysis for more details.

**Window Width around detected peaks for normalizing APD to 0% and 100% (Number of Sample):** The program must first determine what is the amplitude of the action potential in order to find a percent repolarization based on that amplitude. This parameter specifies the width of a window, centered on a detected activation time, which is used to find the maximum and minimum of the signal. The maximum is detected as simply the maximum value in the right hand side of the window. The minimum can either be the median or minimum value in the left side of the window. Rapidly paced data should use smaller window values (<50).

**Percent Repolarization (0% to 100%):** Once the program determines the amplitude of the signal, the program begins looking for data points that have an amplitude less than the amplitude of the signal multiplied by the percent specified in this parameter. The program considers 100% repolarization to be the baseline of the signal, and 0% repolarization to be the maximum value of the signal.

**Additional filtering for repolarization. Filter cutoff frequency. (0 to 500 Hz):** Since detection of repolarization time requires additional filtering, this filter passes all frequencies up to the specified one. Frequency of 500 Hz is equivalent to no filtering at all, and frequency of 0 Hz would return a straight line. Nosier data requires lower cutoffs to eliminate the higher frequency noise from the signal.

**Choose Baseline using Median Value:** Select this option to select the baseline value based on the median value in the left side of the window around detected peaks. Select this option for noisier data.

**Choose Baseline using Minimum Value:** Select this option to select the baseline value based on the minimum value in the left side of the window around detected peaks. Select this option for cleaner data, and if you want to know where the absolute minimum of a signal lies.

**Use Existing Bas/Max Annotations:** When the program is run, a Bas and Max annotation are returned to the waveform windows indicating where the program set the baseline and maximum value to be respectively. These annotations can be physically corrected. Select this option if the Bas/Max annotations have been corrected. The program will not try to determine the baseline of maximum values automatically. Instead it will use the existing annotations.

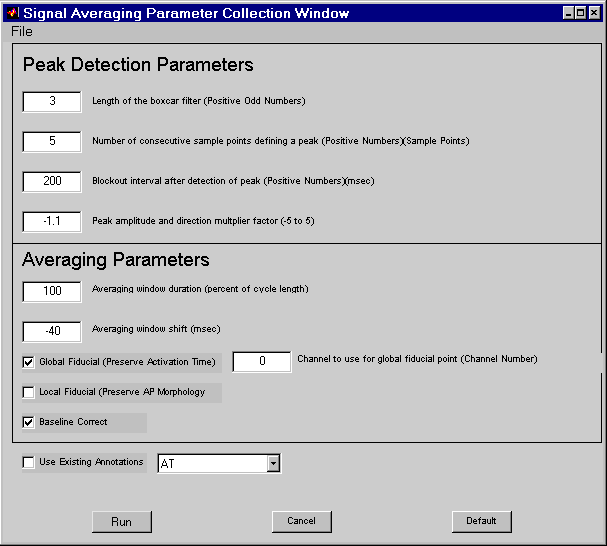
**Save Channel/Bas/Max Values in ASCII Format:** Select this option to save the channel number, baseline annotation time, and maximum annotation time in an ASCII file. This data can then be read in a program such as EXCEL in order to determine the amplitude of each channel’s signal.

Data Analysis 5.5 SIGNAL AVERAGING ALGORITHM

### 5.5 SIGNAL AVERAGING ALGORITHM

**Purpose:** This program will take channels with multiple action potentials and average them into single action potentials.

**Algorithm:** The program finds the activation time of all the optical signals within a channel. All the signals are lined up at a common timeQ, based on the activation time, and summed. The final signal is divided by the number of signals found in that channel. The averaged signals are returned to the AVERAGED DATA window.



**RUN:** Runs the algorithm to generate Repolarization Times.

**CANCEL:** Cancels the command and closes the analysis.

**DEFAULT:** Reverts all the numerical choices in the window to their default settings.

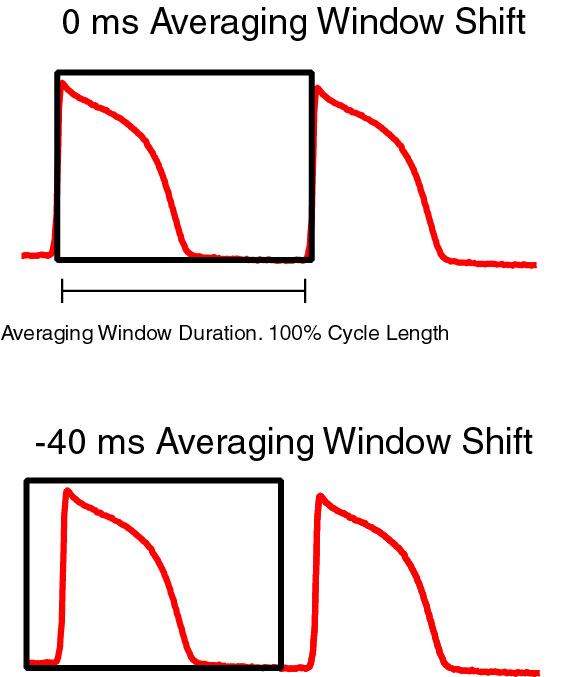
Data Analysis 5.5 SIGNAL AVERAGING ALGORITHM

**FILE Pull Down Menu:** *Save parameters:* option to save customized parameters for later use.

*Load parameters:* option to retrieve previously saved parameters.

The first four parameters are Peak Detection Parameters and are identical to those of Activation Time analysis. Refer to the section IV. Activation Time Analysis for more details.

**Averaging Window Duration (percent of cycle length):** The program finds multiple activation times within channels. It uses the average difference between activation time of all the channels to determine the cycle length. This parameter dictates how much of the signal (in ms) is returned to the AVERAGED DATA window. It also serves the purpose of determining whether or not to use the last activation time in the average. For example, if the last action potential only has a duration of 80% of the cycle length, and the Averaging Window Duration is set at 100, then the last action potential will not be used in the average. If the Averaging Window Duration is set to 80% or lower, then the final action potential will be used in the average.

**Averaging Window Shift (ms):** All signals are aligned to the activation time which is considered time 0 ms. If the averaging window shift is set to 0, then the returned signal will have the first data point as the activation time of the signal (See figure). If the Averaging window shift is set to –40 ms, then the first returned data point will be 40 ms prior to the activation time.

**Global Fiducial (Preserve Activation Time):** When this option is selected, the user may specify a single channel to assign activation times to. Each subsequent channel has an activation time assigned to it that corresponds to the exact same times as the activation times in the chosen channel. Each signal will therefore have it’s original activation time with respect to the global fiducial.

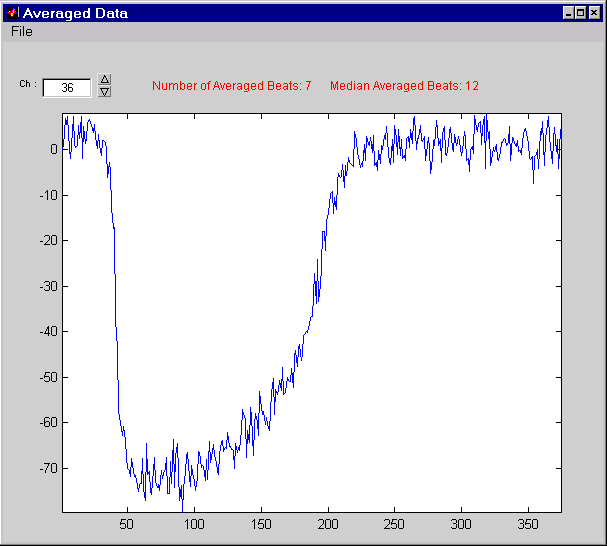
**Local Fiducial (Preserve Action Potential Morphology):** When this option is selected, each channel is assigned an activation time based on the peak detection algorithm. Each signal within a channel will be aligned at a time zero and summed. Therefore, every activation time will accurately match every other activation time within a channel. However, the activation times across channels will be lost.

**Use Existing Annotations:** Physically corrected annotations can be used to define the activation times of signals. Select this option if annotations have been corrected. The program will NOT try to determine the activation times automatically. Instead it will use the existing annotations.

Data Analysis 5.5.1 SIGNAL AVERAGING Window

### 5.5.1 SIGNAL AVERAGING WINDOW

**Purpose:** This program will show a single action potential for every channel based on the Signal Averaging Algorithm.



**FILE Pull Down Menu:** *Save parameters:* option to save customized parameters for later use.

*Load parameters:* option to retrieve previously saved parameters.

**Ch:** This shows which channel is currently being shown in the signal averaging window. The user may scroll through the channels by using the up and down arrows, or by entering the channel number manually in the box to the immediate right of the “Ch:” label.

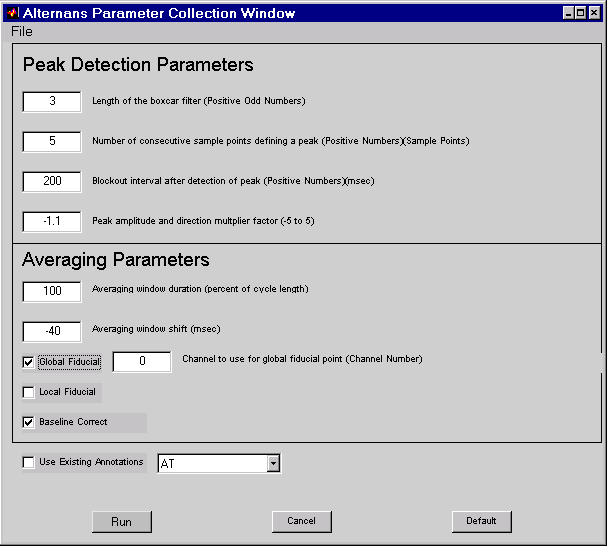
**Number of Averaged Beats:** This indicates how many beats were found for that specific channel. If the number of beats found in that channel does not equal the Median Averaged Beats, the text appears in red. Otherwise, the text is black.

**Median Averaged Beats:** The program records how many beats are averaged from each channel. It finds the median of the number of beats and reports it in this field. If the user highlighted 12 beats from the Zeng Stripchart window, then the Median Averaged Beats should be 12.Data Analysis 5.6 ALTERNANS ALGORITHM

### 5.6 ALTERNANS ALGORITHM

**Purpose:** This program will take channels with multiple action potentials and average them into two action potentials, odd and even beats. The user may use the output to determine the alternans in a signal.

**Algorithm:** The program finds the activation time all the optical signals within a channel. Even and odd signals are lined up at common times, based on the activation time, and summed. The final signal is divided by the number of signals found in that channel. The averaged signals are returned to the ALTERNANS DATA window.



**RUN:** Runs the algorithm to generate Repolarization Times.

**CANCEL:** Cancels the command and closes the analysis.

**DEFAULT:** Reverts all the numerical choices in the window to their default settings.

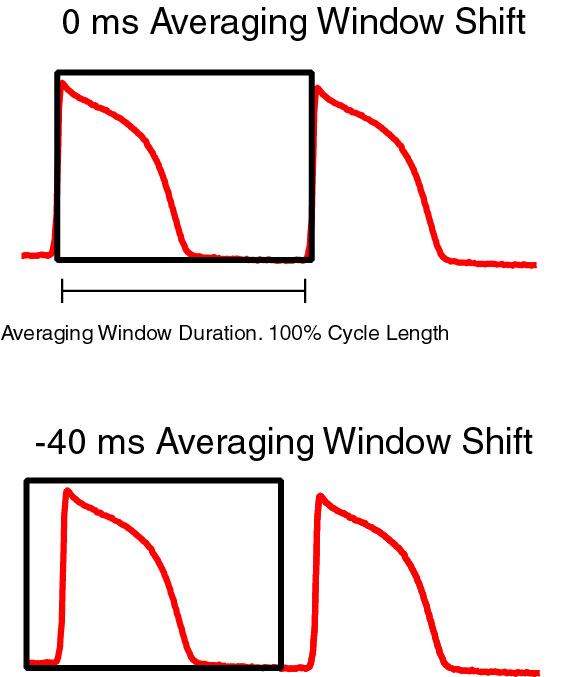
**FILE Pull Down Menu:** *Save parameters:* option to save customized parameters for later use.

*Load parameters:* option to retrieve previously saved parameters.

Data Analysis 5.6 ALTERNANS ALGORITHM

The first four parameters are Peak Detection Parameters and are identical to those of Activation Time analysis. Refer to the section IV. Activation Time Analysis for more details.

**Averaging Window Duration (percent of cycle length):** The program finds multiple activation times within channels. It uses the average difference between activation time of all the channels to determine the cycle length. This parameter dictates how much of the signal (in ms) is returned to the AVERAGED DATA window. It also serves the purpose of determining whether or not to use the last activation time in the average. For example, if the last action potential only has a duration of 80% of the cycle length, and the Averaging Window Duration is set at 100, then the last action potential will not be used in the average. If the Averaging Window Duration is set to 80% or lower, then the final action potential will be used in the average.

**Averaging Window Shift (ms):** All signals are aligned to the activation time which is considered time 0 ms. If the averaging window shift is set to 0, then the returned signal will have the first data point as the activation time of the signal (See figure). If the Averaging window shift is set to –40 ms, then the first returned data point will be 40 ms prior to the activation time.

**Global Fiducial (Preserve Activation Time):** When this option is selected, the user may specify a single channel to assign activation times to. Each subsequent channel has an activation time assigned to it that corresponds to the exact same times as the activation times in the chosen channel. Each signal will therefore have it’s original activation time with respect to the global fiducial.

**Local Fiducial (Preserve Action Potential Morphology):** When this option is selected, each channel is assigned an activation time based on the peak detection algorithm. Each signal within a channel will be aligned at a time zero and summed. Therefore, every activation time will accurately match every other activation time within a channel. However, the activation times across channels will be lost.

**Use Existing Annotations:** Physically corrected annotations can be used to define the activation times of signals. Select this option if annotations have been corrected. The program will NOT try to determine the activation times automatically. Instead it will use the existing annotations.

Data Analysis 5.6.1 ALTERNANS Window

### 5.6.1 ALTERNANS WINDOW

**Purpose:** This program will show two action potential for every channel based on the Alternans Algorithm. In addition, the green trace shows the difference between the two signals. All the data contained within the borders of the pink box are summarized in the gray boxes at the top of the window.

Every channel must undergo two tests. The first test is a *Cycle Length Stability* test. The program goes through every channel and assigns activation times to the action potentials it can find. The cycle length is calculated as the difference in time between two subsequent beats. If the cycle length of any beat falls outside of one standard deviation of all the cycle lengths within a channel, the channel will have failed the *cycle length stability* test.

The second test is the *Beat Count* test. The number of beats found within a channel is compared to the average number of beats for every channel. If the number of beats found in a single channel is greater than or less than the average number of beats in the entire data set, the channel will fail the *Beat Count* test.

The results of *Cycle Length Stability* test and the *Beat Count* test are returned to the MALTAB window shortly before the ALTERNANS window opens.

**Example:**

============================================

Channel CL stability Beat Count

--------------------------------------------------------------------------

21 GOOD BAD

36 BAD BAD

37 GOOD BAD

38 GOOD BAD

39 BAD BAD

40 GOOD BAD

42 BAD BAD

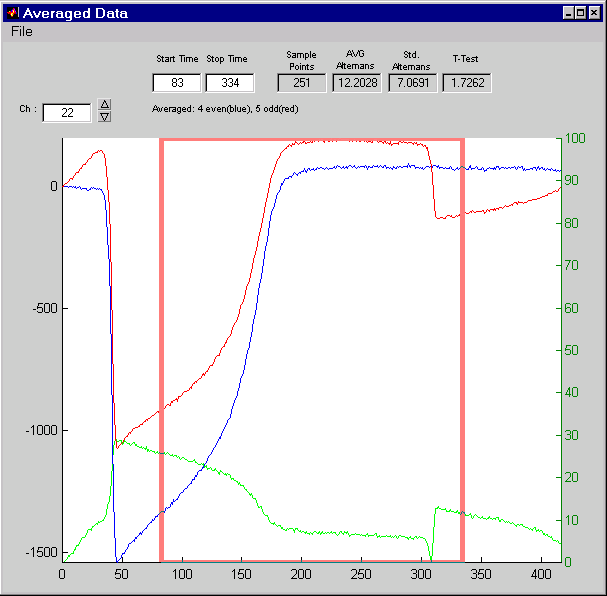
44 GOOD BAD

50 GOOD BAD

51 GOOD BAD

This means that channel 21, for example, had a cycle length that fell within one standard deviation of the cycle lengths within that channel. However, it had fewer beats than the average number of beats in the entire data set. This output is useful to identify channels that had possible annotation errors in them. The user can use the WAVEFORMS window to view and correct the annotations manually, or re-run the ALTERNANS algorithm using different parameters.

Data Analysis 5.6.1 ALTERNANS Window



**FILE Pull Down Menu:** *Save Data:* Option to concatenate the odd and even beat and save them in a file recognizable by Zeng’s program.

*Export Time,Avg Odd, Avg Even:* Option to save the averaged odd and even beats, along with time, to an ASCII file.

*Export Alternans Quantification:* Option to save all the alternan values appearing in upper part of the window in an ASCII format.

*Exit:* Closes the Alternans window

**Ch:** This shows which channel is currently being shown in the signal averaging window. The user may scroll through the channels by using the up and down arrows, or by entering the channel number manually in the box to the immediate right of the “Ch:” label.

Data Analysis 5.6.1 ALTERNANS Window

**Averaged:** This indicates how many beats were found for that specific channel. The number of odd beats as well as the number of evened beats are returned here.

**Sample Points:** This value indicates how many points of a single trace are contained within the pink box.

**AVG Alternans:** The green trace represents the difference between the red and blue lines. The average alternans is simple the mean value of that trace within the pink box.

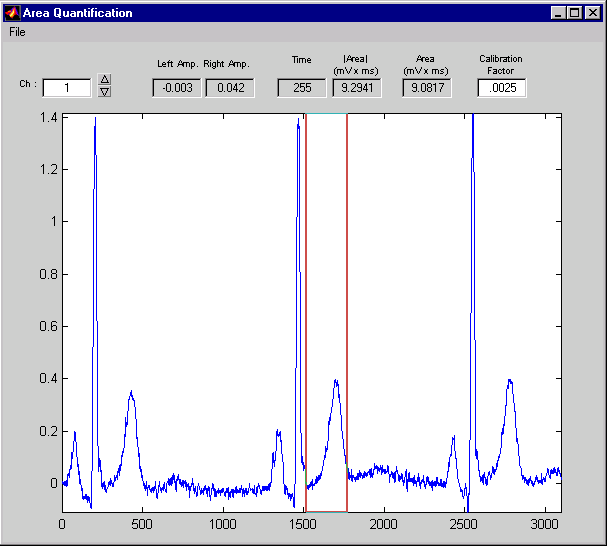
**Std. Alternans:** The green trace represents the difference between the red and blue lines. The value returned in this box is the standard deviation of the green line within the pink box.

**T-Test:** The program performs a T-test on the red and blue lines. The value of the T-test are returned in this box. The T-Test is calculated by taking the mean value of the differences between the even and odd trace and dividing that quantity by the standard deviation of that difference. The higher the number indicates the larger variation between the even and odd beats. This translates to a larger T-Test value is equivalent to larger differences in that part of the signal.

Data Analysis 5.7 Area Quantification Window

### 5.7 AREA QUANTIFICATION WINDOW

This window may be accessed from the STRIPCHART Window from the analysis pull down menu by selecting Other -> AreaQuantification. A segment of the ECG must already be highlighted for the window to appear. This window may be used to calculate signal durations, as well as the area under a curve using electronic calipers.



**FILE Pull Down Menu:**

*Export Single Channel:* Exports the waveform in an ASCII format.

*Exit:*  Closes the AreaQuantification window.

**Ch:** This shows which channel is currently being shown in the AreaQuantification window. The user may switch the channel by entering the channel number manually in the box to the immediate right of the “Ch:” label, or by scrolling through the channels using the up and down arrows.

**Left Amp:** This number displays the amplitude in mV of the waveform intersecting the electronic caliper on the left.

**Right Amp:** This number displays the amplitude in mV of the waveform intersecting the electronic caliper on the left.

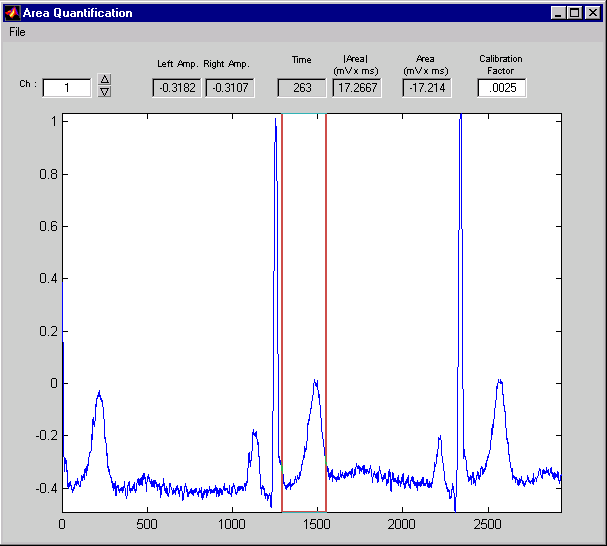
**Time:**  The time in milliseconds between the two electronic calipers.

**|Area| (mV x ms):** displays the absolute area, calculated by the integral of the curve between the two calipers. In other words, it adds the area below the zero line to the area above the zero line.

**Area (mV x ms):** displays the area calculated by the integral of the curve between the two calipers by subtracting the area below the zero line from the area above the zero line.

Data Analysis 5.7 Area Quantification Window

>>Note of Caution: The zero line always appears at the beginning of the segment selected from STRIPCHART, therefore the calculations of area are dependent upon the selection of the zero line. For example, in the above window, the Area calculated is 9.2941, whereas, if the segment selected were modified to begin a few milliseconds later, like so:



the Area calculated is displayed as 17.2667, which is quite different from the previous measurement. Therefore, the user must exercise caution when selecting the original segment for which AreaQuantification window is called.

**Calibration Factor:** automatically displays the mV/(analog to digital) ratio, and the user may manually modify the calibration factor by entering the desired number in the box immediately below the “Calibration Factor:” label.

### 5.8 CUSTOM ALGORITHMS

Users can create their own algorithms using a template. The template file is called SKELETON.m and can be found in the M directory of Zeng. The SKELETON.m file includes directions and comments imbedded in the code to guide the user in creating their own algorithms.

Programmers should be familiar with writing MATLAB functions before proceeding to write a zeng M-file.

The first part of the SKELETON.m file includes code that must be present in order for the file to be compatible with the *‘Zeng’ data analysis package.* The second part of the file is empty except for a few lines that say “ALL YOUR CODE GOES HERE.” This is where the user can write their own code.

There are a few lines at the bottom of the file, which are necessary for returning the annotations to the *‘Zeng’ data analysis package*. Only the variable names should be changed to match the users code.

The entire SKELETON.m code has been included as an appendix to this document.

# 6 Graphical Representation of Data

A series of programs were developed to view everything from movies of the data to conduction velocity vectors. Each graphical representation is an attempt to display data in a sensible way. For example, if someone is studying conduction velocity, isochrones of conduction velocity would be appropriate. However, if someone wanted to show an interesting meandering wavefront, it may be best to represent it as a movie.

This chapter discusses the various windows for displaying and manipulating data. Almost every window has the ability to export the generated graphics to various programs for publication quality editing. Each program contained here is extremely powerful in its own right. However, the programs in this chapter are primarily for organizing the data and putting it in an easy to understand form. MATLAB figures however do not make for good publication material. As such, the data can always be exported to more sophisticated graphics programs such as Adobe Illustrator and Microsoft Excel in order to get it into final publication form.

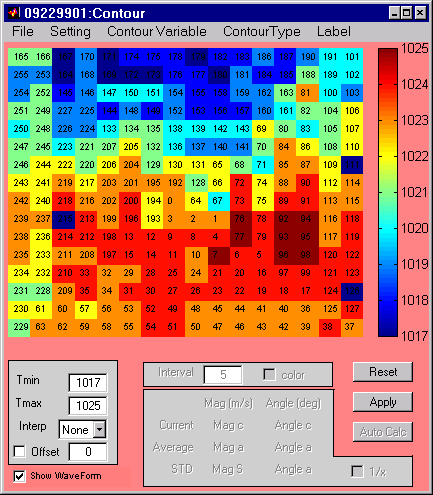
### 6.1 CONTOUR WINDOW

This multifunction window is used for visualizing important events associated with recorded data sets. Isochrones of activation or repolarization can be generated from this window. In addition, any annotation can be transformed into an isochrone map.

An additional program is bundled into the CONTOUR window allowing the user to measure conduction velocities. The data can be exported to any more powerful graphics packages such as Excel or DeltaGraph for journal quality finishing.

If the Waveforms window is open, a user may click on any pixel and have the data channel appear in the WAVEFORMS window. This is especially useful for annotation times that do not make sense spatially. For example, a single annotation with a time value significantly higher or lower than any other annotation time in a data set will reduce the number of isochrones generated. A uniformly colored map will be returned with only one pixel having a color significantly different from the rest. This obviously lends no additional information to the investigator. The user can click on these outliers and observe if an annotation time was selected incorrectly in the WAVEFORMS window.

Graphical Representation of Data 6.1 CONTOUR Window



**FILE Pull Down Menu:** *Print:* Option to print the entire Contour Window

*Print Setup:* Option to set printer attributes before printing the entire Contour window.

*Export►*

*Adobe Illustrator:*Exports the Contour window as a valid Adobe Illustrator format.

*jpg:* Exports the Contour window as a jpg image

*pict:* Exports the Waveform window as a pict image.

*xyt:*Exports the data into an ASCII file. Column one contains the x location of each pixel. Column 2 contains the y position of each pixel, and the 3rd column is the time associated with the annotation being viewed. Array positions with no value are not exported. Note: Interpolated values are also exported.

*xyt (full array):*Exports the data into an ASCII file. Column one contains the x location of each pixel. Column 2 contains the y position of each pixel, and the 3rd column is the time associated with the annotation being viewed. All array positions are exported and positions without a value are exported as "NaN". Note: Interpolated values are also exported. (added KRL 12/11/01)

**SETTING Pull Down Menu:**  *Change LUT:* Select this option in order to change the lookup table for this file. The lookup table is a file which tells a program how each channel is arranged spatially.

***Vector Parameters:*** This option opens the Vector Parameter Window. This window is explained in more detail under the VII.1. Vector Parameter Window.

***Colormap:*** In order to display the contours in different colors, select this option. There are six colormaps to choose from.

Graphical Representation of Data 6.1 CONTOUR Window

**CONTOUR VARIABLE Pull Down Menu:**

***Mapping Array:*** Displays only the channels in their spatial locations defined by the lookup table.

***Difference:*** Displays the spatial difference between the first and second enabled annotations listed directly below this option. The array fills in with colors corresponding to the different times. This only corresponds to annotations existing in the currently selected data in the Stripchart window.

***Activation Time:*** Displays the spatial activation time of the data selected in the Stripchart window.

***Repolarization Time:*** Displays the repolarization times spatially of the data selected in the Stripchart window.

**CONTOUR TYPE Pull Down Menus:**

***Raw Data:*** When this option is selected, each pixel displays a specific color that corresponds to the point in time of a selected annotation (from the CONTOUR VARIABLE Pull Down Menu).

***Isochrone:*** This option will generate a contour map of the currently selected annotation. This is useful for seeing the average spatial distribution of selected parameter.

***Vectors:*** This option creates vectors for each pixel that correspond to the gradient of the currently selected annotation. The gradient has a magnitude as well as a direction.

**LABEL Pull Down Menus:** *Channels:* This option causes the channel numbers to be superimposed on the data set.

*Time:* This option causes the time of the annotation to be superimposed on the data set.

*Channel + Time:* Select this option to see both the channel number and time of the annotation displayed in the same channel.

**TMIN:** This option sets the minimum time value that will be displayed in the axis above. The user may use this option in order to make all data sets begin with the same time value.

**TMAX:** This option sets the maximum time value that will be displayed in the axis above. The user may use this option in order to make all data sets end with the same time value.

**INTERP:** This value dictates the spatial interpolation. An interpolation of 8 for example creates 8 pixels out of one pixel. This option makes the spatial map appear less pixilated.

**OFFSET:** When this is selected, each pixel will have this number subtracted from it. This is very useful for having the datasets begin at a predefined time point such as zero.

**SHOW WAVEFORM:** When this checkbox is selected, clicking on a pixel will cause its corresponding channel data to appear in the Waveforms window. When this is unselected, nothing happens when clicking on a pixel.

**INTERVAL:** This option can only be used when displaying isochrones. This value specifies how many time points are considered a single color. Decreasing this number will create more isochrones.

**COLOR:** When this checkbox is selected, isochrone colors are displayed in the axis above. When the box is not selected, only the lines corresponding to the boundaries of the isochrone appear.

**RESET:** Whenever the user makes a change to an annotation in the Waveforms window, the Contour window will not reflect those changes until this button is pressed.

Graphical Representation of Data 6.1 CONTOUR Window

**APPLY:** Whenever a change is made in any of the edit boxes in the Contour window, the user may press apply to effect those changes. However, editing a value in any of the input boxes followed by hitting the <enter> key will also apply these changes.

**AUTOCALC:** This option is only activated after vectors have been assigned. The vectors are passed to the Auto Vector Calc window discussed under VII.2 Auto Vector Calc window.

**Parameters inside the gray box**

The parameters in the gray box are active only when vectors are selected from the CONTOUR TYPE Pull Down Menu. When a vector is selected, the tail of the vector will become blue or red. Red denotes the currently selected vector.

**CURRENT:** These two values correspond to the currently selected vector (Red Tail). The magnitude and angle of the vector are returned here.

**AVERAGE:** These two values correspond to all the vectors with red or blue tails. The average magnitude and angle of the selected vectors are returned here.

**STD:** These two values correspond to the standard deviation of all the selected vectors (Red or Blue). The standard deviation of the magnitude and average are returned here.

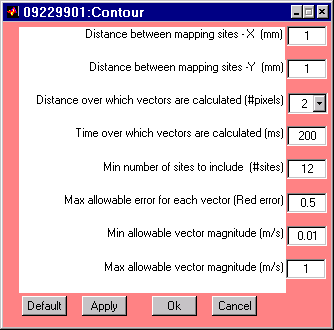
**1/x:** This option is used to invert the units of the vectors. The default is unchecked. Data appears normally as m/s.

Graphical Representation of Data 6.1.1 VECTOR PARAMETER Window

### 6.1.1 VECTOR PARAMETER WINDOW

This window is activated from the Contour window under Settings -> Vector Parameters.These values govern how Phil Bayly’s algorithm determines vectors for a given dataset. The user should be familiar with Phil Bayly’s algorithm before using this function. Refer to:

Bayly PV, Ken Knight BH, Rogers JM, Hillsley RE, Ideker, RE, Smith WM. *Estimation of conduction velocity vector fields from epicardial mapping data.* IEEE Trans Biomed Eng 1998;45:563-571

****

**DISTANCE BETWEEN MAPPING SITES-X (mm):** This parameter indicates how far apart each pixel is from its neighbor in the horizontal.

**DISTANCE BETWEEN MAPPING SITES-Y(mm):** This parameter indicates how far apart each pixel is from its neighbor in the vertical direction.

**DISTANCE OVER WHICH VECTORS ARE CALCULATED:** The gradient for a particular pixel is determined by the values of that pixel’s neighbors. This parameter instructs how many neighbors in every direction to include in determining the vector. For example, a value of 4 tells the program to look at 4 pixels in every direction to determine the gradient.

**TIME OVER WHICH VECTORS ARE CALCULATED (ms):** This parameter indicates the range of time from which vectors can be selected. Setting a low number eliminates annotations that fall outside of that range from the vector calculation. This program will not use any annotations that have a time value greater than this number.

**MIN NUMBER OF SITES TO INCLUDE (#sites):** This value checks for how many sites need to have an activation time assigned. Example: if “Min Number of Sites To Include” is 12 and “Distance over which vectors are calculated” is 2, the program will search the surrounding 5x5 pixels (25 pixels in total) and return a conduction vector if 12 of 25 pixels have activation times.

**MAX ALLOWABLE ERROR FOR EACH VECTOR (red error):** This parameter is not used

**MIN ALLOWABLE VECTOR MAGNITUDE (m/s) :** All vectors that are found which have a magnitude below this number are considered invalid and will not be returned.

Graphical Representation of Data 6.1.1 VECTOR PARAMETER Window

**MAX ALLOWABLE VECTOR MAGNITUDE (m/s):** All vectors that are found to have a magnitude greater than this value are considered invalid and will not be returned.

**DEFAULT:** This option will restore the default values to this window.

**APPLY:** This option applies the defined values and calculates the vectors.

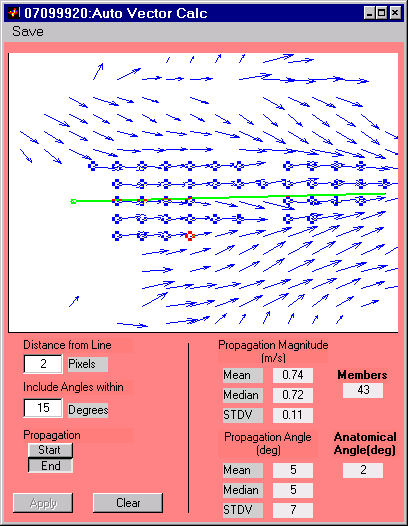
**OK:** This option sets the parameters in memory but does not force the program to calculate the vectors immediately.

**CANCEL:** This option ignores all changes made to this window and closes it.

Graphical Representation of Data 6.1.2 AUTOCALCULATION OF VECTORS Window

### 6.1.2 AUTOCALCULATION OF VECTORS WINDOW

The Contour window has a rudimentary function to manually select vectors of interest and record the averages and standard deviations of those vectors. This window can be called from the Contour window. The programs developed in this window provide more robust and unbiased analysis of vectors based on anatomical features and statistical parameters.



**SAVE Menu:** This option allows the user to specify a filename. All data generated while the Auto Vector Calc window will be subsequently saved to this file every time this option is selected.

**DISTANCE FROM LINE:** The line shown in green is defined by the user. The user specifies how many pixels away from the line the program should consider for analysis.

**INCLUDE ANGLES WITHIN \_\_\_\_ DEGREES:**  When the green line is created, the angle that line makes with the x-axis is automatically calculated. All vectors that are analyzed must fall within +/- the number of degrees specified here to be used in the average. Use a larger number if the gradient vectors vary largely.

Graphical Representation of Data 6.1.2 AUTOCALCULATION OF VECTORS Window

**PROPAGATION:**

**START:** Select this option and then click anywhere in the above axis. This places a green dot which signifies the beginning of the user defined green line.

**END:** Select this option and click anywhere in the above axis. This indicates the end location of the green line. A green line automatically appears between the location specified by the Start and End values. The line should be in the direction of propagation. The angle of this line is instrumental in automatically selecting vectors. Once the end has been established, the program automatically begins looking for vectors that comply with the “Distance From Line” and “Include Angles Within parameters.”

**APPLY:** The user may manually select a number of vectors in the axis above. These vectors are then treated as a line. Every vector that is within the distance specified in “Distance From Line,” will be considered for inclusion for automatic calculation. Every vector whose angle falls within the average angle of the selected pixels +/- the value specified in “Include Angles Within,” will be considered for inclusion for automatic calculation.

**PROPAGATION MAGINTUDE (m/s):**

**MEAN:** The mean propagation magnitude of all the selected vectors appear here.

**MEDIAN:** The median propagation magnitude of all the selected vectors are returned here.

**STDV:** The standard deviation of the magnitude of all the selected vectors are returned here.

**PROPAGATION ANGLE (deg):**

**MEAN:** The mean propagation angle in degrees of all the selected vectors appear here.

**MEDIAN:** The median propagation angle in degrees of all the selected vectors are returned here.

**STDV:** The standard deviation of all selected vector’s angles are returned here.

**MEMBERS:** The total number of selected vectors are displayed in this area.

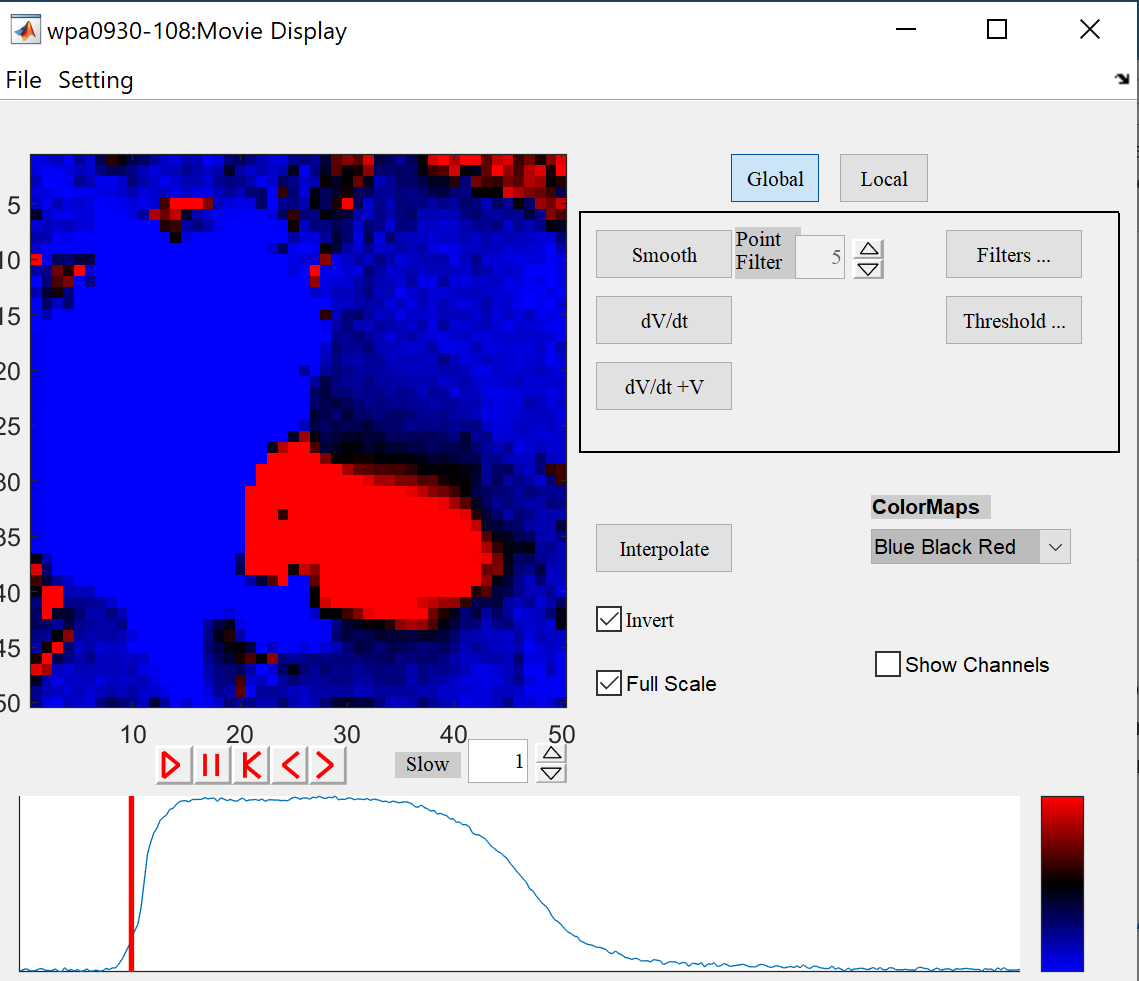
**ANATOMICAL ANGLE:** The angle specified by the user defined green line appear here. The closer the Anatomical Angle is to the Mean Propagation Angle, the more accurate the automatic calculation of all the vectors will be. Graphical Representation of Data 6.2 MOVIE Window

### 6.2 MOVIE WINDOW

This window is an interface for displaying movies of data sets as well as for manipulating the data set for movie output. The movie appears in the upper left corner axis. The waveform tracing at the bottom of the window shows data for the active channel. A red line marches across the bottom window indicating where in time the movie is. The user may skip around in the movie by clicking anywhere in the bottom axis.

Of all the applications in the *‘Zeng’ data analysis package,* the MOVIE program is the only program that physically changes the data set. The changed data is not returned to the main set of windows such as the STRIPCHART or WAVEFORMS window. All changes remain local to the MOVIE window. In order to see the changes the data must first be saved and then re-opened in the *‘Zeng’ data analysis package.*

It is important to note that this program was intended to be used to clean up data for graphical representation. As such, the experimenter should use the functions provided in this program very cautiously IF they intend to infer results from the manipulated data set.



**FILE Pull Down Menu:** *Save Data:* option to save manipulated data in a format Zeng’s program can read.

*Make Movie:* option to create a Windows Media Player compatible movie.

**SETTING Pull Down Menu:**  *Expand Decimated Data:*option to fill in empty data channels with interpolated data.

*Change LUT:* option to select a new look up table.

Graphical Representation of Data 6.2 MOVIE Window

**GLOBAL:** All functions in the box below this option are applied to the entire data set when this box is pushed.

**LOCAL:** All functions in the box below this option are applied to the currently highlighted channel.

**SMOOTH:** Applies a boxcar filter specified in the POINT FILTER box to the data set. Use a higher number for a smoother signal.

**dV/dt:** This function takes the derivative of the signal and returns the output to the movie window and the waveform trace below.

**dV/dt:** This function adds the derivative of the signal to the original signal. Use this function when trying to emphasize depolarization in a movie.

**FILTERS:** This opens the Filter Window. These filters are applied to the signals in the movie program only.

**THRESHOLD:** This opens the Threshold Window. Thresholding is useful for making a noisy baseline not flicker in a movie.

**INTERPOLATE:** This button applies only to the currently selected channel. The value of the channel is replaced by an average of the eight surrounding channels.

**COLORMAPS:** Use this pull down menu to specify what colors the movie will use.

**INVERT:** This inverts an optical action potential. When this checkbox is selected (default), the optical action potentials appear upright.

**FULL SCALE:** Optical action potentials often have different amplitudes based on light intensity and dye concentrations. Full scale normalizes the scale of all signals to zero and 100. Fullscale the data to remove heterogeneities due to light and dye concentration.

**SHOW CHANNELS:** Select this option to superimpose all the channel numbers on the movie array.

 : Plays the movie

 : Pause the movie on the current frame.

 : Instructs the program to start the movie at the beginning of the data set.

 : Go one frame back on the movie.

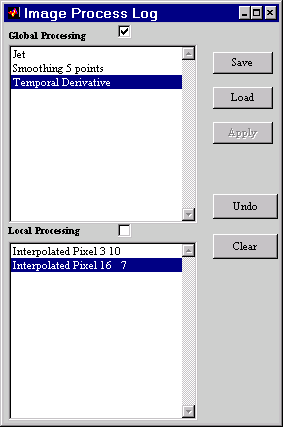
 : Go one frame forward on the movie.

**SKIP FRAME:**  This parameter specifies how many frames are skipped in a movie. A skip frame parameter of 2 will show every second frame. Use a higher skip frame value to make the movie play faster.

Graphical Representation of Data 6.2.1 IMAGE PROCESS LOG Window

### 6.2.1 IMAGE PROCESS LOG WINDOW

This window automatically opens when the MOVIE program is launched. All actions performed on the data set in the MOVIE window are recorded here. These actions can be saved for use at a later time.



**GLOBAL PROCESSING:** All functions performed on the entire data set are recorded here. This can be made the active window for undoing functions.

**LOCAL PROCESSING:** All functions performed on specific channels are recorded here. This can be made the active window for undoing functions.

**SAVE:** This option allows the user to save all the functions appearing in both windows for use at a later time.

**LOAD:** This option allows the user to retrieve previously saved functions.

**APPLY:** When saved functions are loaded, the user has the option of applying the Global or Local functions to the data set depending on which processing boxes are checked. For example, if the user whishes to only apply local processing to a movie, the user would select the Local Processing check box and click Apply.

**UNDO:** This option will undo the currently highlighted function in the currently selected window.

**CLEAR:** This option will revert the data set back to its original form. All operations will be cleared from the Image Process Log.

Graphical Representation of Data 6.2.2 DIGITAL FILTERS Window

### 6.2.2 DIGITAL FILTERS WINDOW

This window is called by the Filter button in the MOVIE program. Different filters and their characteristics can be chosen graphically in this window. The filter is then applied to either the entire data set or a specific channel depending on the selection of the Global and Local options in the MOVIE program.



**FILTER NAME:** The algorithmic name of a filter can be chosen here.

**FILTER TYPE:** Option to specify whether the filter is Low Pass, High Pass, Band Pass, or Band Stop.

**ORDER:** Specifies the filter order. Higher order filters have sharper cutoffs but worse frequency domain characteristics.

**LOW CUT-OFF:** Option to specify the lower frequency for the chosen filter.

**HIGH CUT-OFF:** Option to specify the higher frequency for the chosen filter.

**PASS RIPPLE:** Option to specify the size of the ripple in the pass band. This value is in decibles. The program will attempt to get as close to this value as possible.

**STOP RIPPLE:** Option to specify the size of the ripple in the stop band. This value is in decibles. The program will attempt to get as close to this value as possible.

**PREVIEW:** Clicking on this button will temporarily apply the filter to the data in the Movie Window. This allows the user to see the effect of the filter before applying it.

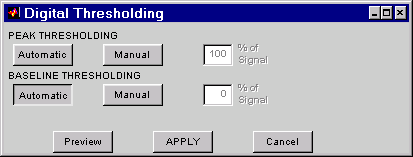
**APPLY:** Clicking on this button will apply the filter to the data in the Movie Window.

**CANCEL:** This button closes the window without applying any filters to the data in the Movie Window.

Graphical Representation of Data 6.2.3 DIGITAL THRESHOLDING Window

### 6.2.3 DIGITAL THRESHOLDING WINDOW

This window is called by the Threshold button in the MOVIE program. When making a movie, it is sometimes desirable to remove the flickering noise from the peaks and baselines of all the signals. This window can be used to create a movie with very clear activation and repolarization patterns.



**PEAK THRESHOLDING**

**AUTOMATIC:** Select this option to automatically remove all signals above 80% of the amplitude of a signal.

**MANUAL:** Select this option to set manually select in the Movie window what percent of the signal will be thresholded. All values above this threshold will be set to the maximum of the signal.

**BASELINE THRESHOLDING**

**AUTOMATIC:** Select this option to automatically remove all signals below 20% of the amplitude of a signal.

**MANUAL:** Select this option to set manually select in the Movie window what percent of the signal will be thresholded. All values below this threshold will be set to zero.

**PREVIEW:** Clicking on this button will temporarily apply the filter to the data in the Movie Window. This allows the user to see the effect of the filter before applying it.

**APPLY:** Clicking on this button will apply the filter to the data in the Movie Window.

**CANCEL:** This button closes the window without applying any filters to the data in the Movie Window.

# APPENDIX A. THE SKELETON.m File

% AUTHOR: Your name here

% DATE LAST MODIFIED: 00/00/00

% This is a Skeleton program for use with Zeng's Analysis program.

% You will enter your own description here.

% The first line in the program is the function that is going to be called.

% The left hand arguments between the [ ] are what Zeng's program expects to be returned

% The right hand arguments between the ( ) are what Zeng's program sends to the SKELETON

% function

function [Label,Comment,Annote\_Out]=SKELETON(Z,X1X2,ChLabel,Annote)

%The function name is SKELETON, and the filename is SKELETON.M

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% OUTPUTS OUTPUTS OUTPUTS OUTPUTS OUTPUTS OUTPUTS %

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% LABEL: Zeng's program expects a label back after the program is executed.

% The label must be a 1x3 character array. Only one label can be returned per

% function

Label='SKE';

% In this case, the program returns the label "SKE"

% COMMENT: Zeng's program expects a Comment back after the program is executed.

% The comment must be a 1x(any size) character array. This is the comment

% that is viewed in Zeng's Annotation window.

Comment='SKELETON PROGRAM';

% In this case, the program will return the comment "SKELETON PROGRAM"

% ANNOTE\_OUT: Zeng's program expects a variable called Annote\_out which contains:

% a channel number in the first column

% a time in msec in the second column

% a label in the third column

%

% Since the label is a character array, the entire output Annote\_Out

% is by default a character array. Zeng's stripchart program decodes

% Annote\_Out and parses it into valid numerical and character arrays.

Annote\_Out=[];

% In this case, the program returns an empty set in Annote\_Out which means that

% no information was generated

APPENDIX A SKELETON.m

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% INPUTS INPUTS INPUTS INPUTS INPUTS INPUTS INPUTS %

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Z: Zeng's program automatically sends a value of 2 with Z automatically

% Technically, Z specifies the order of a butterworth lowpass filter

% You may ignore this variable.

% X1X2: Zeng's program sends the start time and stop time of the window in msec.

% X1X2(1) is the start time

% X1X2(2) is the stop time

% IMPORTANT. Remeber, that after you have determined a time to return in Annote out

% to add X1X2(1) to the column 2 of Annote\_Out

% This can be done with the following line of code inserted at the end of the function.

%

% Annote\_Out(:,2)=Annote\_Out(:,2)+X1X2(1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% START OF PROGRAM START OF PROGRAM START OF PROGRAM %

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Zeng's program does not send the raw data to these functions. The variable Data

% is a global variable which can be seen by all programs if they have the following

% statement directly beneath the function call.

global Data

global Fig

% In order to use the data, you must have this line in your code.

NewData=Data{Z};

% NEVER change the contents of the variable Data since Zeng's program relies on its

% never changing integrity to function.

% NewData is ROW oriented. Each channel is a single row. In order to extract

% the fourth channel for example you would use the code:

FOURTHCHANNEL=NewData(4,:);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% GRAPHICAL USER INTERFACE SPECIFICATIONS %

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% ENTER DESCRIPTION FOR PARAMTERS

Parameter1Description=('Length of the boxcar filter (Positive Odd Numbers)');

Parameter2Description=('Number of Consecutive pixels defining a peak (Positive Numbers)(Pixels)');

Parameter3Description=('Number of ms to blockout after a detected peak (Positive Numbers)(msec)');

Parameter4Description=('A scaling factor to determine whether a peak is detected');

Parameter5Description=(' ');

Parameter6Description=(' ');

Parameter7Description=(' ');

APPENDIX A SKELETON.m

% ENTER PARAMETER DEFAULTS

P1Default=3;

P2Default=5;

P3Default=200;

P4Default=-1.1;

P5Default=0;

P6Default=0;

P7Default=0;

% The 3 Line Below should NEVER be edited.

% These lines send information to the Graphical User Interface.

PDefaults=[P1Default;P2Default;P3Default;P4Default;P5Default;P6Default;P7Default];

continue=Interface(Parameter1Description,Parameter2Description,Parameter3Description,Parameter4Description,Parameter5Description,Parameter6Description,Parameter7Description,PDefaults);

if continue==1

%%%%%%%%%%% ERROR CHEKCING OF GRAPHICAL USER INTERFACE VARIABLES

%%%%%%%%%%% GOES IN THE LINES BELOW

% Parameter 1

Log.Defaults.Parameter1=str2num(get(Fig.Parameter1,'String'));

if isempty(Log.Defaults.Parameter1)

close(Fig.Figure)

warndlg('VALUES WERE INCORRECT. PROGRAM ABORTED')

break

end

% Parameter 2

Log.Defaults.Parameter2=str2num(get(Fig.Parameter2,'String'));

if isempty(Log.Defaults.Parameter2)

close(Fig.Figure)

warndlg('VALUES WERE INCORRECT. PROGRAM ABORTED')

break

end

% Parameter 3

Log.Defaults.Parameter3=str2num(get(Fig.Parameter3,'String'));

if isempty(Log.Defaults.Parameter3)

close(Fig.Figure)

warndlg('VALUES WERE INCORRECT. PROGRAM ABORTED')

break

end

% Parameter 4

Log.Defaults.Parameter4=str2num(get(Fig.Parameter4,'String'));

if isempty(Log.Defaults.Parameter4)

close(Fig.Figure)

warndlg('VALUES WERE INCORRECT. PROGRAM ABORTED')

break

end

APPENDIX A SKELETON.m

% Parameter 5

Log.Defaults.Parameter5=str2num(get(Fig.Parameter5,'String'));

if isempty(Log.Defaults.Parameter5)

close(Fig.Figure)

warndlg('VALUES WERE INCORRECT. PROGRAM ABORTED')

break

end

% Parameter 6

Log.Defaults.Parameter6=str2num(get(Fig.Parameter6,'String'));

if isempty(Log.Defaults.Parameter6)

close(Fig.Figure)

warndlg('VALUES WERE INCORRECT. PROGRAM ABORTED')

break

end

% Parameter 7

Log.Defaults.Parameter7=str2num(get(Fig.Parameter7,'String'));

if isempty(Log.Defaults.Parameter7)

close(Fig.Figure)

warndlg('VALUES WERE INCORRECT. PROGRAM ABORTED')

break

end

%%%%%%%%%%%%%%%%%%%% END ERROR CHECKING

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

% Parameter Variable Names are:

% Log.Defaults.Parameter1

% Log.Defaults.Parameter2

% Log.Defaults.Parameter3

% Log.Defaults.Parameter4

% Log.Defaults.Parameter5

% Log.Defaults.Parameter6

% Log.Defaults.Parameter7

% ALL YOUR CODE GOES HERE. % ALL YOUR CODE GOES HERE.

% ALL YOUR CODE GOES HERE. % ALL YOUR CODE GOES HERE.

% ALL YOUR CODE GOES HERE. % ALL YOUR CODE GOES HERE.

% ALL YOUR CODE GOES HERE. % ALL YOUR CODE GOES HERE.

% ALL YOUR CODE GOES HERE. % ALL YOUR CODE GOES HERE.

% ALL YOUR CODE GOES HERE. % ALL YOUR CODE GOES HERE.

% ALL YOUR CODE GOES HERE. % ALL YOUR CODE GOES HERE.

% ALL YOUR CODE GOES HERE. % ALL YOUR CODE GOES HERE.

% ALL YOUR CODE GOES HERE. % ALL YOUR CODE GOES HERE.

% This is the code responsible for returning annotations

% in Zeng's format.

% The first variable is all the annotations already created. DO NOT CHANGE THE FIRST

% VARIABLE'S NAME

% The second variable is the channel currently being organized

% The third variable is the time at which the annotation will be recorded

% The fourth variable is the three character variable label

APPENDIX A SKELETON.m

Annote\_Out=[Annote\_Out;CHANNELNUMBER TIME Label];

end

Annote\_Out(:,2)=Annote\_Out(:,2)+X1X2(1);

Annote\_Out=[Annote;Annote\_Out];

# APPENDIX B. TABLE OF INTERPIXEL RESOLUTION FOR GIVEN CAMERAS/OPTICS

\*\*created by Greg Hoeker on 01/06/15

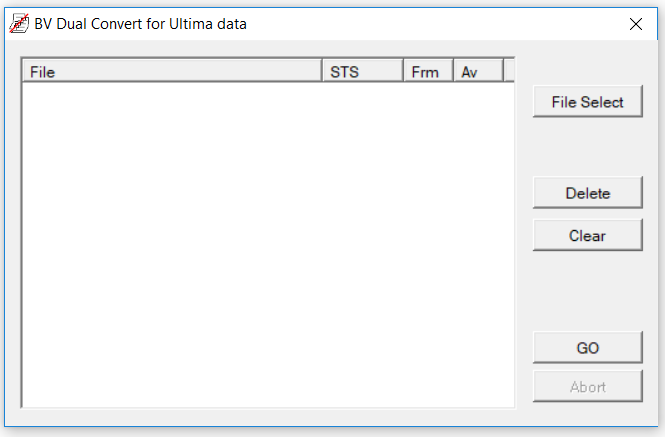
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | | **MiCAM Ultima** | **MiCAM02-CMOS** |
| Sensor size (mm2) | | | 10 x 10 | 5.76x4.8 |
| Full array size (pixels) | | | 100 x 100 | 96 x 80 (<1.2ms/frame) |
|  | | |  | 192 x 160 (≥1.2ms/frame) |
| Active array size (pixels) | | | 100 x 100 | 92 x 80 (<1.2ms/frame) |
|  | | |  | 188 x 160 (≥1.2ms/frame) |
| Pixel size (µm2) | | | 100 x 100 | 60 x 60 (for 92x80) |
|  | | |  | 30 x 30 (for 188x160) |
| ***PIXEL RESOLUTION (mm)*** | | | | |
| Magnif (front/back lens) | | | 1.0X (1.0/1.0) | 1.25X (=2.0/1.6) |
| FOV (mm2) | | | 10 x 10 | 4.42 x 3.84 |
|  | Binning | 1x1 | 0.100 | 0.048 |
|  |  | 2x2 | 0.200 | 0.096 |
|  |  | 3x3 | 0.300 | 0.144 |
| Magnif (front/back lens) | | | 0.63X (0.63/1.0) | 0.63X (1.0/1.6) |
| FOV (mm2) | | | 15.9 x 15.9 | 8.83 x 7.68 |
|  | Binning | 0.096 | 0.159 | 0.096 |
|  |  | 0.192 | 0.318 | 0.192 |
|  |  | 0.288 | 0.477 | 0.288 |
| Magnif (front/back lens) | | |  | 0.394X (0.63/1.6) |
| FOV (mm2) | | |  | 14.02 x 12.182 |
|  | Binning | 0.096 |  | 0.152 –kcc 2/22/19 |
|  |  | 0.192 |  | 0.3046 – kcc 2/22/19 |
|  |  | 0.288 |  | 0.457 – kcc 2/22/19 |

# APPENDIX C. CONVERTING SCIMEDIA, DUAL CAMERA DATAFILES INTO ZENG FORMAT

\*\*created by Xiaobo Wu and Ryan King 5/4/21

* Download BV\_Tool (Ver.0808) from <https://www.scimedia.com/fis/support/download/ultima/> and install it on your computer.
* To separate data from camera 1 and camera 2, it is necessary to use ‘BV\_Tool\_0808’ > ‘BV\_DualConv’
* BV\_tools dualconv produces filenames in the format xxx-xxx\_A.rsh and xxx-xxx\_B.rsh containing information from camera 1 and 2, respectively. Only the ‘.rsh’ file contains an ‘\_’; in order to ‘batch’ these files it is necessary to delete the ‘\_’, resulting in a final file name xxx-xxx(A/B).rsh.
* In vconvert.m
  + Line 461 – if fname(end-4)==’A’
    - Produces error “Undefined operator ‘==’ for input arguemnts of type ‘cell’
  + Change to: if fname{end-4}==’A’ | fname{end-4}==’B’

Initial camera output is stored as “rking0130-001.rsh” which is a file containing data from both cameras A and B. In order to process these data in Matlab it is necessary to separate the cameras into independent header files. To do so, access BV\_DualCnv within the BV\_Tools0808 folder.

Open BV\_DualCnv (see left). Begin by clicking “file select” and then navigating to the folder containing your combined header files. Select the combined header files for input and click “GO”. The output will return files labeled “rking0130-001\_A.rsh” and “rking0130-001\_B.rsh.” These correspond to cameras 1 and 2, respectively. At this stage it is worthwhile to use BV\_ANA to open the header files and ensure that the cameras were properly separated.

If files were properly separated it is necessary, that you eliminate the underscore from the file name for proper import into the labs batch file. To facilitate this process, I wrote a script “dualcnv\_rename.m” which will batch process this step (attached at end of document).

dualcnv\_rename.m

function [] = dualcnv\_rename()

% Collect inputs

if exist('dualcnvpath.mat', 'file')

load ('dualcnvpath.mat')

end

if ~exist('lookpath', 'var')

lookpath = pwd;

elseif isempty(lookpath) || ~ischar(lookpath)

lookpath = pwd;

end

[fnit, pi] = uigetfile([lookpath '\*\\*\_\*.rsh'], 'Select files for renaming');

lookpath = pi;

save('dualcnvpath.mat', 'lookpath');

charloc = find(fnit == '\_',1, 'last'); %locate the character '\_'

flist = dir([pi '\\*\_\*.rsh']); %read folder for total number of files to rename

for temp = 1:length(flist) %determine number of files in folder

fni = flist(temp).name;

newfnibeg = fni(1:charloc-1); %filename characters preeceding charloc

newfniend = fni(charloc+1:end); %filename characters following charloc

newfni = [newfnibeg newfniend]; %new filename, absent the '\_'

if ~exist([pi '\renamedoutput'], 'dir') %create output directory

mkdir(pi, 'renamedoutput')

end

save([pi '\renamedoutput\' newfni]); %save renamed files

end

end