Documentation for PFEIFER:
Preprocessing Framework for Electrograms Intermittently Fiducialized from Experimental Recordings

Anton Rodenhauser, Wilson Good, Brian Zenger, Rob MacLeod

January 22, 2018

Contents

Chapter 1

Statement of Need

Preprocessing Framework for Electrograms Intermittently Fiducialized from Experimental Recordings (PFEIFER) is a MATLAB Graphical User Interface tool designed to import, pre-process, and semi-automatically fiducialize cardiac electrogram signals obtained experimentally. The PFEIFER toolkit does this in one, easy to use framework implementable across many experimental platforms. Typical experimental signals require a substantial amount of processing before they can be used for diagnostic or analytical purposes. However, these processing steps used by cardiac researchers are rarely discussed in literature. Discrepancies in how these steps are performed change the resultant data and complicate the interpretation of results. PFEIFER allows for a common framework from which these processing steps can be performed while allowing the user to manually fiducialize representative beats and extend those fiducials across several seconds of continuous recording.

PFEIFER takes unprocessed cardiac electrical signals and performs a preprocessing routine, allows the user to isolate and fiducialize representative beats, and propagate those manual fiducials to subsequent beats in an automated manner. During experiments whose data we expect to be processed in PFEIFER , tens of thousands of individual heart beats may be recorded across each measurement electrode in the experimental system making the task of manually processing each beat impossible. To combat this large amount of data, other techniques for automatically processing signals and fiducials have been employed but their limited accuracy and inability to operate over multiple heart beats limits the functional range of these tools. PFEIFER leverages the nature that the time signals were recorded simultaneously and that heart beat electrical morphology does not change significantly on a second time scale.

Fiducial based analysis of electrograms makes tracking specific regions of the signal simple and increases the reproducibility of the analysis because the fiducials were seeded on a case-by-case basis by the user. Unfortunately, fiducial based analysis of cardiac electrograms is typically considered too user intensive and is not performed. Using the autofidicucializing function built into PFEIFER the user can apply fiducials from one beat to a subsequent 20-50 beats providing a continuous stream of fiducialized heart beats in a matter of seconds. A combination of this custom algorithm, user set hot keys, and smart fiducial cycling reduce the processing time for these experimental recordings to a few seconds per 'Run' .

PFEIFER was originally created to process the signals captured at the Cardiovascular Research and Training Institute (CVRTI) and therefore expects a very specific file format in order to operate. In order to accommodate for data not acquired at CVRTI we have implemented a File Conversion feature in PFEIFER to convert non-standard file formats to compatible file structure. The only requirements for each data set is a MATLAB matrix of Electrodes X Time.

Chapter 2

System Requirements

2.1 System Configuration

- CPU: Core Duo or higher, recommended i5 or i7
- Memory: 4 GB or higher, recommended 8

2.2 MATLAB Requirements

- Versions Tested: MATLAB r2016b
- Toolboxes required: None

Chapter 3

Installation Instructions

3.1 Add to Path

PFEIFER is designed as a collection of .m scripts and functions usable within an active installation of MATLAB. To run PFEIFER the user must navigate to the directory that contains the PFEIFER source code and add this folder and subfolders to the MATLAB path. Once added to the MATLAB path the user can access the PFEIFER GUI in any directory by calling "PFEIFER " from the command window of MATLAB.

Chapter 4

Glossary of Terms

Below is a list of terms that are used throughout the PFEIFER documentation that may be useful for users to understand the technical language.

- Autofiducialize: The act of finding fiducials in the subsequent beats in the run using the manually fiducialized beat.
- Fiducial: A time instant of particular importance. (e.g. The Start and End of the QRS are common fiducials)
- Fiducialize: The act of manually marking what temporal instances correspond to temporal events of note.
- Group: A set of electrodes belonging to disparate geometries. (e.g. Sock Electrodes vs Torso Electrodes)
- Helper Files: Files that store the PFEIFER settings and store the processing information for the experiment for posterity. Relevant settings are also stored in the output files but these are intended to stay with the experiment.
- Run: A time-aligned recording of potentials across multiple electrodes and electrode types.
- RunGroup: A grouping of Runs over which the groups do not change.
- Splitting: The act of breaking a long run file into lengths that show minute physiological change. Our current practice is run lengths of 15 seconds of acute myocardial ischemia.

Chapter 5

Input Data Overview

PFEIFER is designed work on time-aligned and continuously recorded cardiac signals across several measurement systems. As of the latest release PFEIFER will only calibrate and map Standardized Utah data. Considering this, data not in this standard format will have to be calibrated and mapped before being processed in this system. Similarly, if the data on each channel is not time aligned, this would have to be done by the user prior to being imported into PFEIFER .

If the user's data was not acquired by the Utah CEG group ensure that the data is:

1. **Data must be Time-aligned across all electrodes in a 'Run'**
2. **Data must be mapped such that the first channel in the 'Run' corresponds to electrode #1**
3. **Data must be calibrated such that the potentials in the 'Run' correspond to the actual amplitude of the signal**

5.1 The TS structure

The TS structure is the basic structural element that is needed in order to run .mat files in PFEIFER without having to first run them through the 'File Converter.' The TS structure has a few fields that are required by PFEIFER but the user may add additional fields to the TS structure which will be inherited by all subsequent processing in PFEIFER . The TS structure should be stored within the 'Run' file in the following format: User_Filename.ts.

The necessary fields within the TS structure are:

1. **potvals** - A matrix [m x n] where m is the number of electrode channels and n the number of potential values over time.
2. **numleads** - The number of channels contained within the potval matrix
3. **numframes** - The number of time instances in the potval structure
4. **filename** - The name of the file where the ts structure is saved.
5. **unit** - The units of the data contained within potvals (*e.g.*,mV)
6. **audit** - This is a note stored as a string

7. **label** - This is the intended label that classifies this ‘Run’
8. **leadinfo** - This is a [m x 1] array dictating which electrodes are considered bad (=1) of good (=0). If all electrodes are good each channel should be assigned a 0.

To generate the TS variable with the correct structure, assuming you have assigned your matrix of potential values the variable name ‘potvals’ and the intended filename stored as ‘newFilename’, can be done as follows:

1. **ts.potvals = potvals;**
2. **ts.numleads = size(potvals,1);**
3. **ts.numframes = size(potvals,2);**
4. **ts.filename = newFilename;**
5. **ts.unit = ‘’;**
6. **ts.audit = ‘’;**
7. **ts.label = filename;**
8. **ts.leadinfo = zeros(size(ts.potvals,1),1)**

5.2 Non-Standard Data

PFEIFER has been extended to work with .mat files as long as the potential values are stored in a matrix within the file structure of the .mat file. For PFEIFER to work on these files they must first be converted using the built in ‘File Converter’ which will append the ‘-pf’ tag onto the end of your file name indicating it can be used with PFEIFER . For a more detailed discussion of converting non-standard data see the ‘File Converter’ section. Users may skip the converting of their files if they put their data into the TS structure detailed above.

5.3 Utah-Standard Data

Experimental data acquired by the Computational Electrocardiography Group (CEG) group in Salt Lake City, Utah is stored in a specific .ac2 file format. These are raw, unmapped, and uncalibrated signals captured during an experiment. PFEIFER is easily able to convert .ac2s to .mat files. These .ac2 files also compile metadata along with the potential values within the TS structure and are stored on ‘Run’ by ‘Run’ basis.

It is not necessary to include additional fields beyond what is listed in ‘The TS structure’ section. However, an example of the TS structure in use is the TS structure of the Utah data before it has been processed. The fields typically present before processing include:

1. **ts.filename = ‘Run0001.mat’**
2. **ts.label = 1x79 char**
3. **ts.potvals = 1024x3008 double**
4. **ts.gain = 1024 x 1 uint16**

5. **ts.numleads = 1024**
6. **ts.numframes = 3008**
7. **ts.leadinfo = ‘’**
8. **ts.unit = ‘’;**
9. **ts.audit = ‘’;**
10. **ts.time = ’10:09:53 AM’**
11. **ts.samplefrequency = 1000**
12. **ts.origin = ’Run0001.ac2’**

5.4 Splitting Data

Splitting data is an optional preprocessing step that sections a continuous time signal recording into segments of a user specified length. The purpose of splitting these files is to limit the change in beat morphology within any given ‘Run’ . For acute myocardial ischemia we typically split the files into 15 second segments. File splitting is intended to reduce the amount of manual labor involved in fiducializing by leveraging the built in autofiducialize algorithm. More details on this algorithm can be seen in in the ‘Autofiducializer’ section.

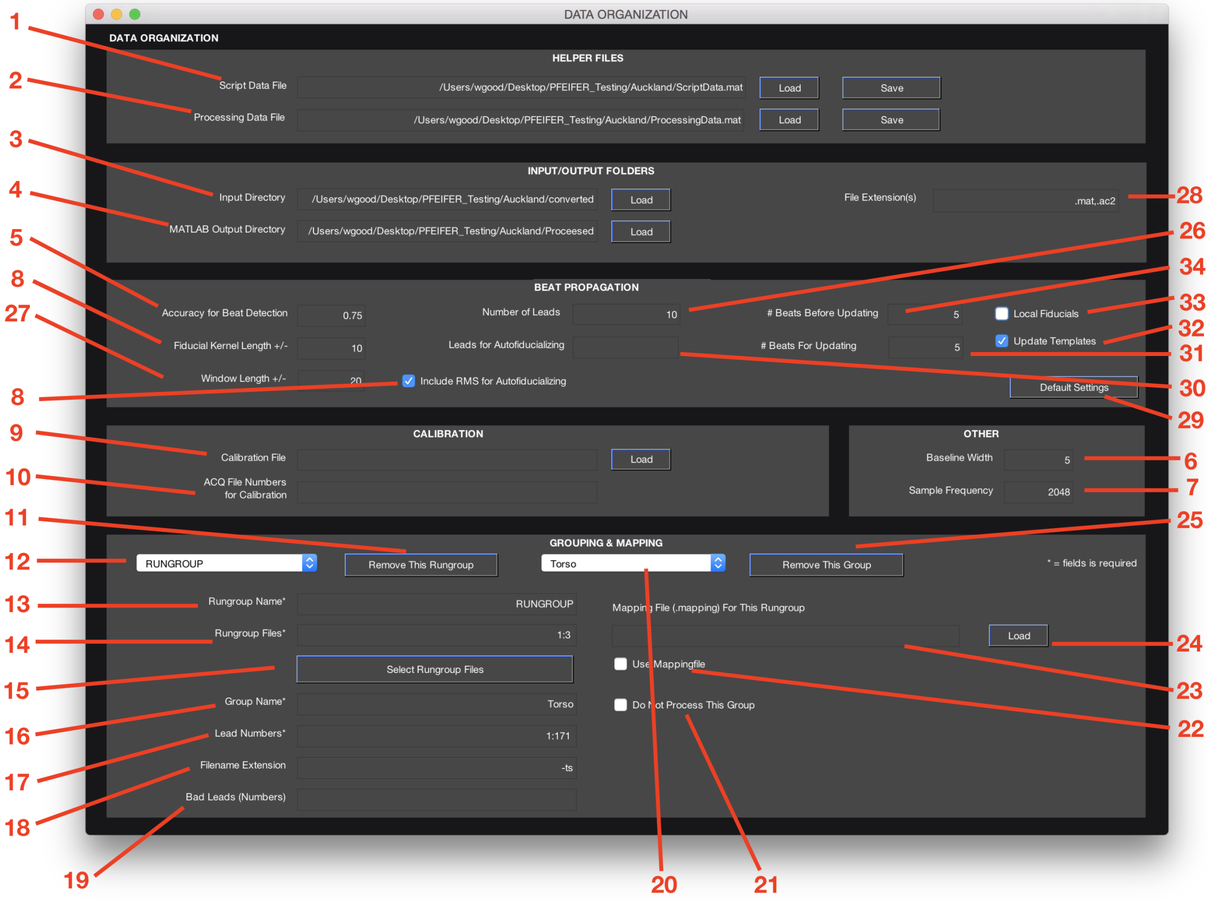
Briefly, the autofiducialize function takes the user placed fiducials and uses a cross correlation algorithm to find the fiducials in all of the subsequent beats with the ‘Run’ file. This is why it is important to constrain the length of each ‘Run’ file, if the beats change too much the autofiducializing algorithm with not be able to accurately fiducialize the signals any further. Split files will maintain the TS file information of the parent file from which it was split.

Chapter 6

API Overview

6.1 Starting PFEIFER

Once PFEIFER has been added to the MATLAB path the user can start PFEIFER by typing "PFEIFER " into the command window of MATLAB. This will launch two windows, the Data Organization window and the Workbench window. These initial windows launch with each initialization of the PFEIFER



DATA ORGANIZATION

HELPER FILES

Script Data File: /Users/wgood/Desktop/PFEIFER_Testing/Auckland/ScriptData.mat

Processing Data File: /Users/wgood/Desktop/PFEIFER_Testing/Auckland/ProcessingData.mat

INPUT/OUTPUT FOLDERS

Input Directory: /Users/wgood/Desktop/PFEIFER_Testing/Auckland/converted

MATLAB Output Directory: /Users/wgood/Desktop/PFEIFER_Testing/Auckland/Proceesed

File Extension(s): .mat,.ac2

BEAT PROPAGATION

Accuracy for Beat Detection: 0.75

Fiducial Kernel Length +/-: 10

Window Length +/-: 20

Number of Leads: 10

Leads for Autofiducializing:

☑ Include RMS for Autofiducializing

Beats Before Updating: 5

Beats For Updating: 5

☐ Local Fiducials

☑ Update Templates

CALIBRATION

Calibration File:

ACQ File Numbers for Calibration:

OTHER

Baseline Width: 5

Sample Frequency: 2048

GROUPING & MAPPING

RUNGROUP

Torso

* = fields is required

Rungroup Name*: RUNGROUP

Rungroup Files*: 1:3

Group Name*: Torso

Lead Numbers*: 1:171

Filename Extension: -ts

Bad Leads (Numbers):

Mapping File (.mapping) For This Rungroup:

☐ Use Mappingfile

☐ Do Not Process This Group

Figure 6.1: Data Organization Window

The data organization window is dedicated to storing relevant data i/o settings, ‘Run’ groupings, electrode groupings, and processing settings required for processing data within PFEIFER . Each portion of the window is described in detail below.

6.2.1 PFEIFER Helper Files

PFEIFER uses two files to store the metadata and fiducial information on an experimental basis. We suggest keeping these two files with your experimental files and having PFEIFER create new ones within the directory of each new experiment. This prevents vital experiment information from being overwritten. While the relevant fiducial information is stored in the output files, PFEIFER looks to the helper files to glean the GUI parameters that drive the fiducial selections. The Script Data File contains all PFEIFER settings. The Processing Data File contains file specific data, e.g. the user selected beat bounds or any fiducials. The helper files help to speed up the processing and rendering of the signals in PFEIFER . For example, rather than setting all PFEIFER settings by hand every time you start PFEIFER , the user can load a Script Data File that has stored the desired settings. PFEIFER will then apply the settings from that loaded Script Data File to the data organization window. When processing an experiment for the first time the user will have to use PFEIFER to generate these files. Note: the user will only do this if the user has not used PFEIFER on the specific data set previously or they would like to alter the settings used to process the data from a prior experiment (figure 6.1-1,2).

6.2.2 Input and Output Directory Settings

As shown in figure 6.1-3,4,28 PFEIFER requires a specified location of user input files and output files. These can be selected by manually entering the path to the directories desired or browsing to them through a system file window. File extensions of files to be opened as input data can be specified in the file extension(s) field. This specifies which files within the directory will be loaded into PFEIFER by a given type. Current supported file types are: .ac2 and .mat.

6.2.3 Input Data Baseline Width and Sample Frequency

The user can select the baseline width and sampling frequency of the acquired data (figure 6.1-6,7). Baseline width is the width of the average used during isoelectric beat selection in order to perform a baseline correction on the input signal. Sampling frequency is the rate at which the input data was sampled during acquisition.

6.2.4 Calibration and Mapping File

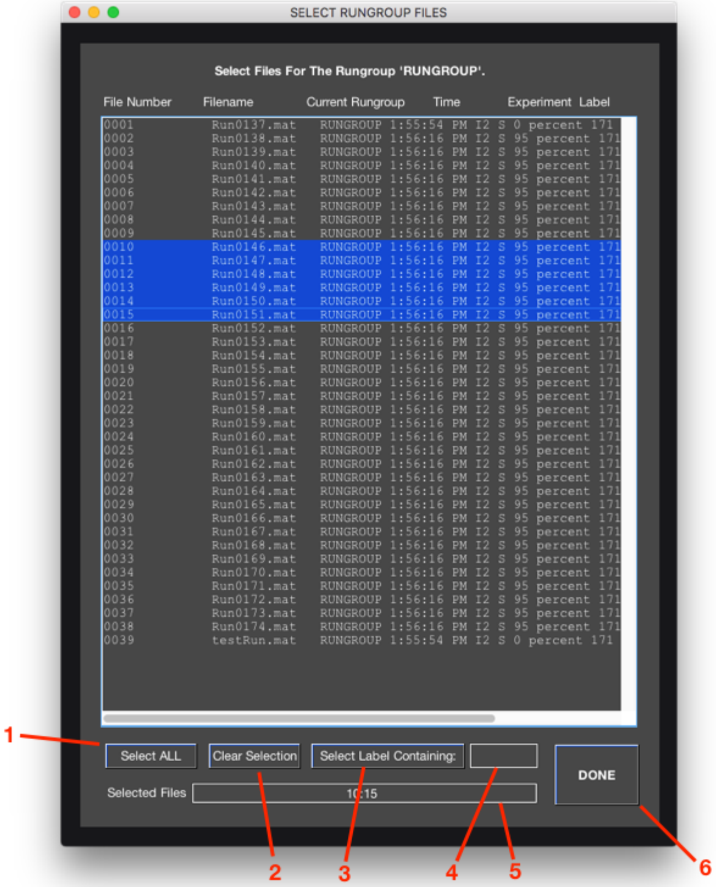
Note: Currently calibration and mapping of the data in PFEIFER is only supported for Utah data. Non-Utah data needs to be calibrated and mapped outside of PFEIFER.

PFEIFER has two options with calibration files. First, it can import specific data points that are user predefined calibration signals.

PFEIFER can also use predefined calibration files. These are stored files created from a previous calibration created by PFEIFER . This file location can be navigated to via the browsing pop-up window. (figure 6.1-9,10)

An important feature of PFEIFER is the ability to map raw data files into user interpretable channel numbers. This is done via the mapping file. The structure of the mapping file is a text file that indicates which lead number should be applied to each channel from the raw signal input (figure 6.1-23,24).

6.2.5 Rungroup and Group Selection



SELECT RUNGROUP FILES

Select Files For The Rungroup 'RUNGROUP'.

```
File Number     Filename       Current Rungroup    Time          Experiment  Label
0001            Run0137.mat    RUNGROUP 1:55:54 PM I2 S 0 percent 171
0002            Run0138.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0003            Run0139.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0004            Run0140.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0005            Run0141.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0006            Run0142.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0007            Run0143.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0008            Run0144.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0009            Run0145.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0010            Run0146.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0011            Run0147.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0012            Run0148.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0013            Run0149.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0014            Run0150.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0015            Run0151.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0016            Run0152.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0017            Run0153.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0018            Run0154.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0019            Run0155.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0020            Run0156.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0021            Run0157.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0022            Run0158.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0023            Run0159.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0024            Run0160.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0025            Run0161.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0026            Run0162.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0027            Run0163.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0028            Run0164.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0029            Run0165.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0030            Run0166.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0031            Run0167.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0032            Run0168.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0033            Run0169.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0034            Run0170.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0035            Run0171.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0036            Run0172.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0037            Run0173.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0038            Run0174.mat    RUNGROUP 1:56:16 PM I2 S 95 percent 171
0039            testRun.mat    RUNGROUP 1:55:54 PM I2 S 0 percent 171
```

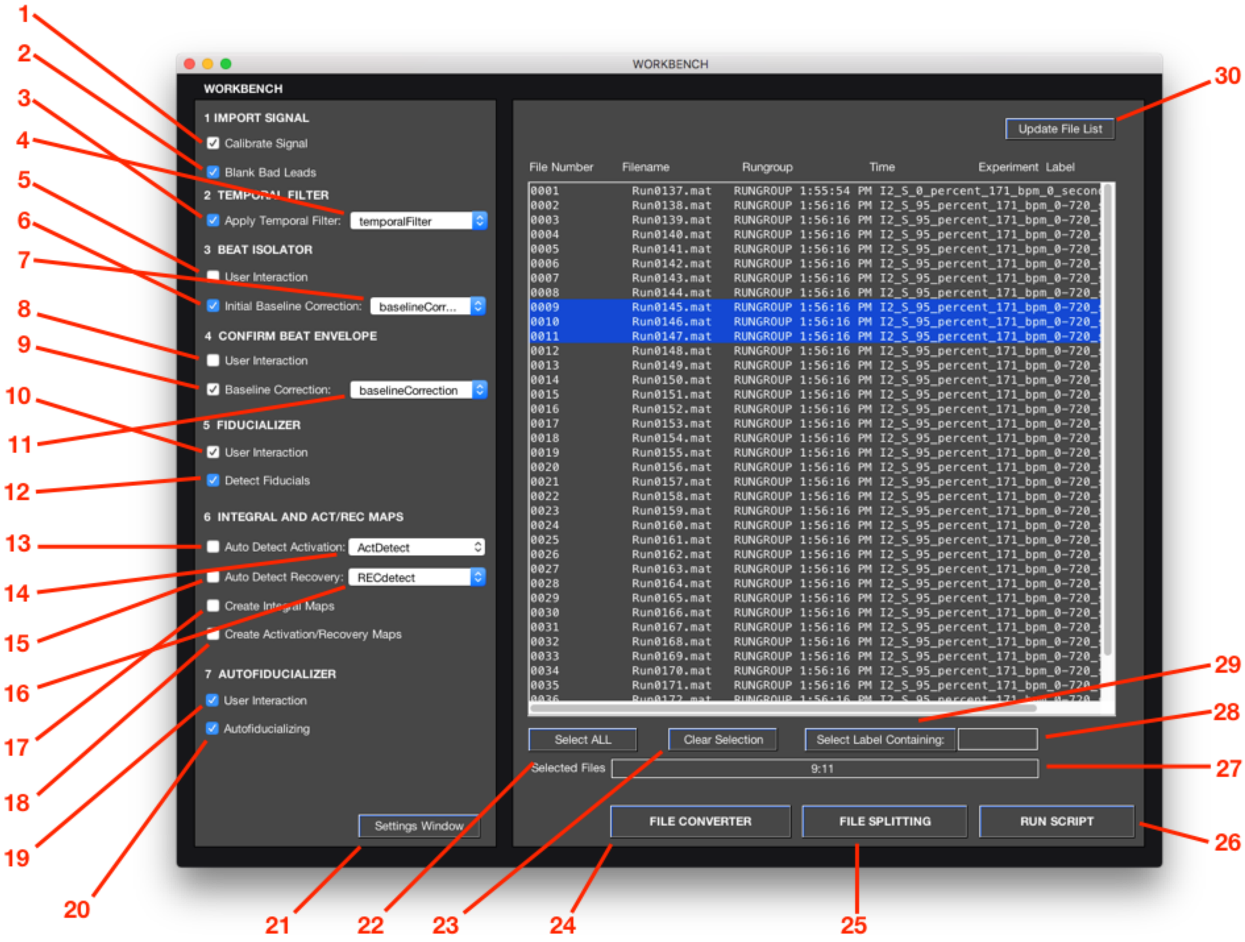
Selected Files: 10:15

Figure 6.2: Select Rungroup File Window

This portion of the window shown in figure 6.2 allows the user to set a specific range of run files to be used with specific channel group settings. This allows the user to specify a range of run files for a specific "rungroup" setting.

Groups are defined by entering the name of the group, the lead or channel numbers corresponding to each measurement device (*e.g.*,sock electrodes), the file extension name, and any predetermined bad lead or channel numbers. This structure allows the user to save out specific groups of leads into predefined file types across the previously set runs in the group. Adding groups is done by selecting "add new group" from the group drop down menu (figure 6.1-12).

6.3 Workbench Window



WORKBENCH

1 IMPORT SIGNAL
☑ Calibrate Signal
☑ Blank Bad Leads

2 TEMPORAL FILTER
☑ Apply Temporal Filter: temporalFilter

3 BEAT ISOLATOR
□ User Interaction
☑ Initial Baseline Correction: baselineCorr...

4 CONFIRM BEAT ENVELOPE
□ User Interaction
☑ Baseline Correction: baselineCorrection

5 FIDUCIALIZER
☑ User Interaction
☑ Detect Fiducials

6 INTEGRAL AND ACT/REC MAPS
□ Auto Detect Activation: ActDetect
□ Auto Detect Recovery: RECdetect
□ Create Integral Maps
□ Create Activation/Recovery Maps

7 AUTOFIDUCIALIZER
☑ User Interaction
☑ Autofiducializing

```
File Number  Filename      Rungroup  Time        Experiment  Label
0001         Run0137.mat   RUNGROUP  1:55:54 PM  I2_S_0_percent_171_bpm_0_secon
0002         Run0138.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0003         Run0139.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0004         Run0140.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0005         Run0141.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0006         Run0142.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0007         Run0143.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0008         Run0144.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0009         Run0145.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0010         Run0146.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0011         Run0147.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0012         Run0148.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0013         Run0149.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0014         Run0150.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0015         Run0151.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0016         Run0152.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0017         Run0153.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0018         Run0154.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0019         Run0155.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0020         Run0156.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0021         Run0157.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0022         Run0158.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0023         Run0159.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0024         Run0160.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0025         Run0161.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0026         Run0162.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0027         Run0163.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0028         Run0164.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0029         Run0165.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0030         Run0166.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0031         Run0167.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0032         Run0168.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0033         Run0169.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0034         Run0170.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0035         Run0171.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720_
0036         Run0172.mat   RUNGROUP  1:56:16 PM  I2_S_95_percent_171_bpm_0-720
```

Selected Files: 9:11

Figure 6.3: Workbench Window

The workbench window (figure 6.3) is a user interaction window that allows the user to select which steps within the data processing they would like to perform on each selected run file. Processing steps are labeled with a check box that can be selected in order to "perform this action" and deselected to "not perform this action". This workbench window also allows the user to interact with the preprocessing steps described in the sections 5.4 and 5.2.

6.3.1 The PFEIFER Toolbox

The PFEIFER toolbox is a component of the workbench window in PFEIFER that allows the user to control which tools to use in processing experimental signals. PFEIFER also supports users importing their own functions to be used in the toolbox, for more details on how to do this please see section 7.2.

Temporal Filter

This section controls the application of the temporal filter on run data. See the section on temporal filter for a more detailed explanation. The user can select between multiple temporal filters via the drop down menu adjacent to the apply temporal filter check box. (figure 6.3-3,4)

Beat Isolator

The slice average signal step allows the user to manually select a region of data from a single run file as a beat to be analyzed and baseline corrected. This process allows the user to select this region of data through another user window (figure 6.3-5-7). This filter also applies a baseline correction based on the first and last parts of the user selected beat range (described in more detail in section 7.3

Confirm Beat Envelop

The baseline correction allows the user to select regions within the previously selected beat to be used as isoelectric regions for the baseline correction. This is done through another user window that is described below.(figure 6.3-8,9,11)

Fiducializer

Detecting fiducials is performed initially via manual user interaction. In this step the user is able to select which parts of the baseline corrected beat correspond to specific elements within the beat. These points selected are saved into the output data file and can be used to analyze specific parameters of the signal over time. (figure 6.3-10,12)

Integral and ACT/REC Maps

Integral and Activation Recovery maps allow the user to toggle if integral and Activation recovery maps should be calculated and outputted as separate data files (figure 6.3-13-18). The user can also select to create integral and activation recovery maps from the specified run files and previously user selected fiducials.

Autofiducializing

This allows the user to have PFEIFER perform and view the output of autofiducializing of the previously user selected ficiducials applied to the entire signal within one run file (figure 6.3-19,20). This can be useful to process long run files with of many heart beats. Autofiducializing is performed using a cross correlation algorithm that identifies beat envelopes and fiducials using the user set fiducials of that run as a template.

6.3.2 Main File List

This list shows all the run files located in the input file directory. The user can select the files to process by mouse click or typing them into the field. The user can update the file list from initial loading if there were changes made to the input file directory outside of PFEIFER (figure 6.3-22-30).

6.3.3 File Converter

Pressing this button opens the File Converter window. This window is used to convert input files which are not in the right format to be used within PFEIFER (see 5.2) into a format that PFEIFER can work with.

6.3.4 File Splitter

Pressing this button opens the File Splitter window that can be used to split a run file into multiple smaller run files. For more information on splitting data, see 5.4.

6.4 The PFEIFER Run Script

Once the desired settings, processing steps, and run files have been selected the user can click "run script." This begins the processing of each individual run with desired predetermined user input. The processing iterates over each file until all selected files are completed based on user specified tasks.

6.5 User Interaction Window General Settings

Each user interaction window has several similar characteristics that allow the user to view the signal in specific and repeatable ways. Below is a description of the important features of these view windows and how the user can select a variety of ways to view the signal.

6.5.1 Navigation Bar

The navigation bar is located at the top of each user interaction pop up window. This allows the user to cycle through selected files from the previously selected file list. It also displays the file name, file label, time the file was created, and file number of the detected file. The 'Stop' button allows the user to terminate the run process of PFEIFER and return to the data organization and workbench windows to make desired modifications. The 'Prev' and the 'Next' buttons abort the processing of the current run and go back to the previous or the next run. The 'Back' button returns you to the first User Interaction Window of the current run (figure 6.4-1-3).

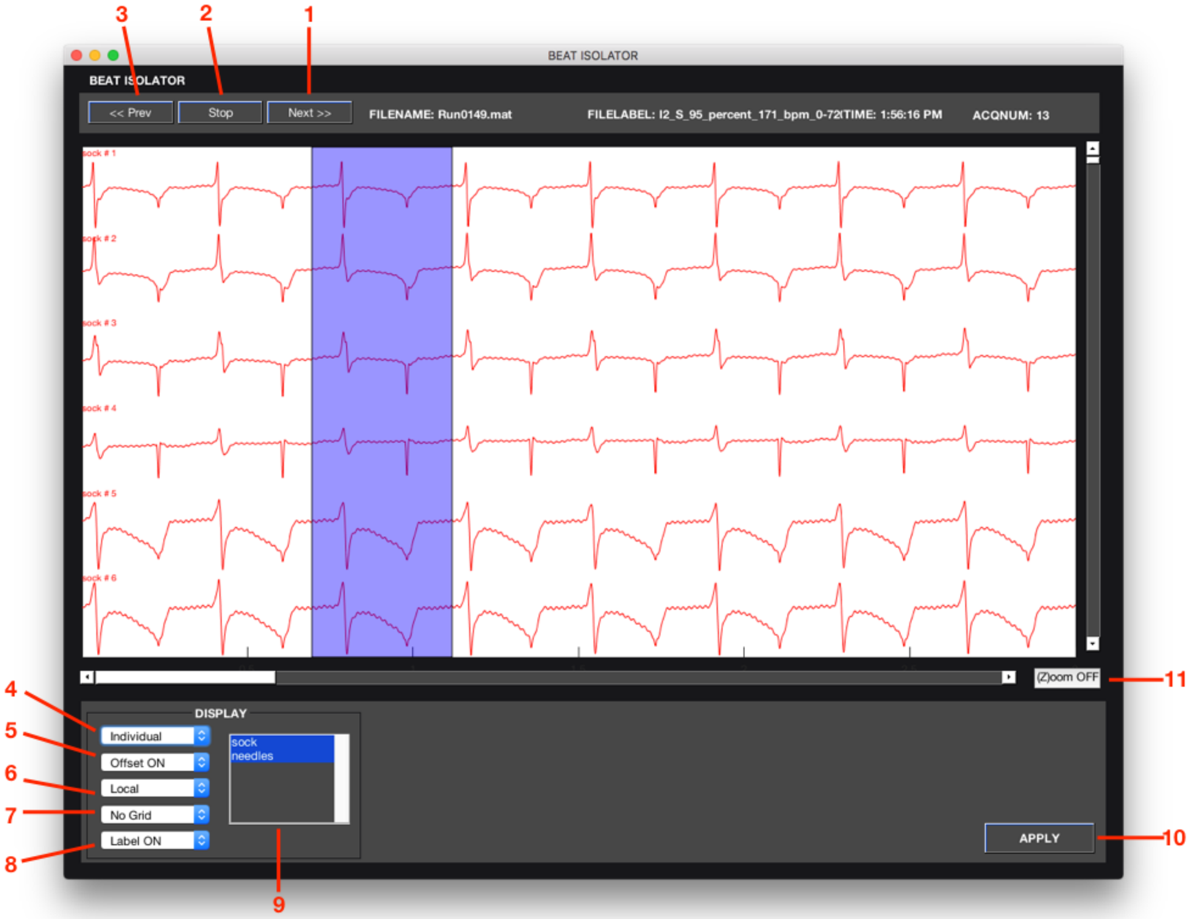
6.5.2 Display

Each user window also has the capability to view the signal in a variety of ways. The first drop down menu of the display section allows the user to select how to view the signal. The signal can be viewed as individual leads, group RMS, and global RMS signals. The user can view the signals with or without an offset and with multiple scaling options: 'Local' scaling scales the signal on a lead by lead basis, 'Group' scaling scales the signal according to the maximum potential value of a group and 'Global' scaling scales the signals to the maximum potential value of all leads. The user can also display signal labels and a predefined grid in the signal window. (figure 6.4-4-9).

6.5.3 Zoom

By pressing the Zoom ON/OFF button you can swap between the viewing and the zoom mode. In the view mode you can simply look at the signals and scroll up and down or to the left and right using the scroll bars. This mode is used to select fiducials. In the zoom mode the user can still scroll up and down or to the left and right but in addition, the user can now also zoom into and out of the signals. Zoom in by clicking and holding the left mouse button and draw a rectangle on the display. After the left mouse button is released the display will zoom into the drawn rectangle. To zoom out, simply press the right mouse button on the display. (figure 6.4-11).

6.6 Beat Isolation Window



BEAT ISOLATOR

FILENAME: Run0149.mat

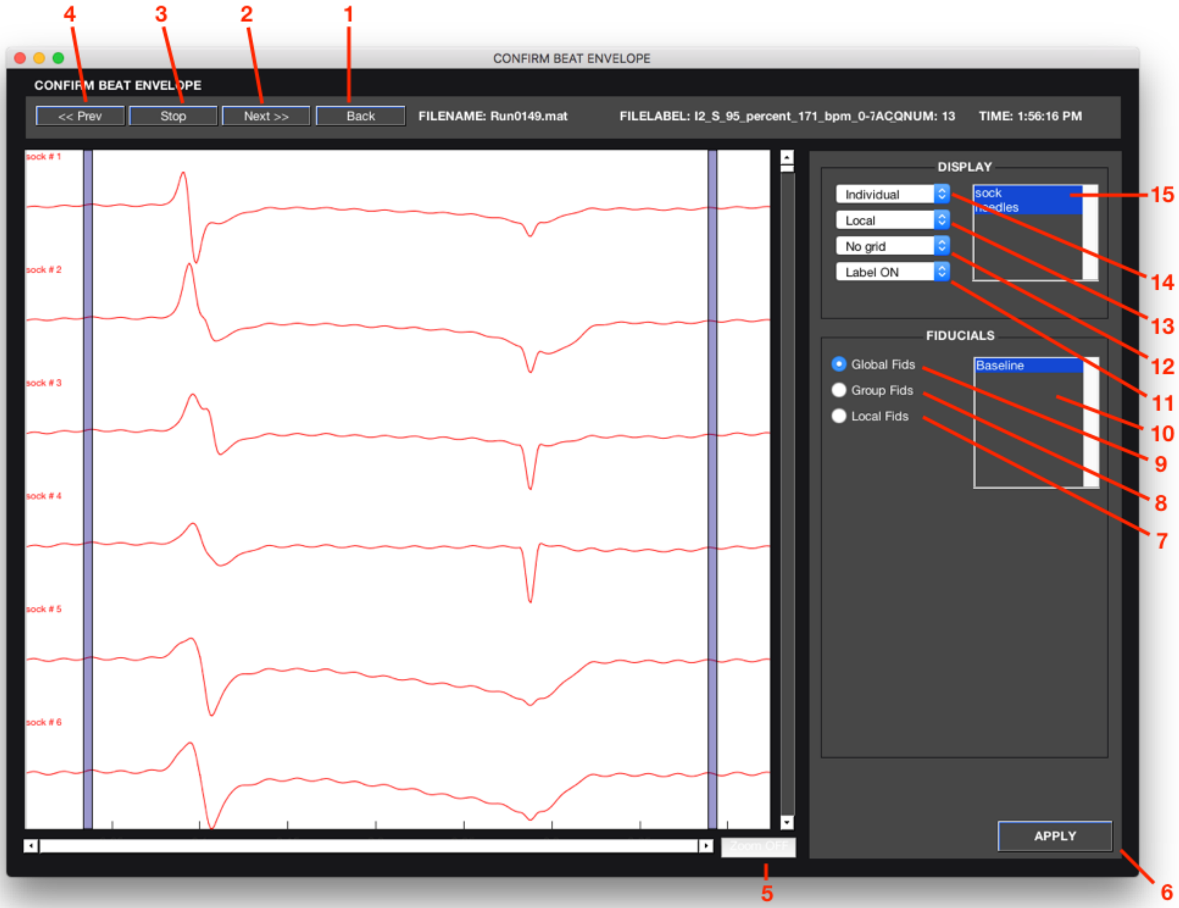
FILELABEL: I2_S_95_percent_171_bpm_0-72(TIME: 1:56:16 PM

ACQNUM: 13

Figure 6.4: Beat Isolation Window

If user interaction was checked for BEAT ISOLATOR a pop up window titled "beat isolator" will appear after pressing apply. This window allows the user to select which portion of the run file will be used as a beat. A left mouse click and drag from the desired time point in the signal will select the beat duration across all channels within the signal. This range can be modified by a left click nearest to the desired edge range for modification. Once the beat has been selected the user must click "apply" to progress to the next PFEIFER processing step. (figure 6.4).

6.7 Confirm Beat Envelope Window



FILENAME: Run0149.mat FILELABEL: I2_S_95_percent_171_bpm_0-7ACQNUM: 13 TIME: 1:56:16 PM

Individual
Local
No grid
Label ON

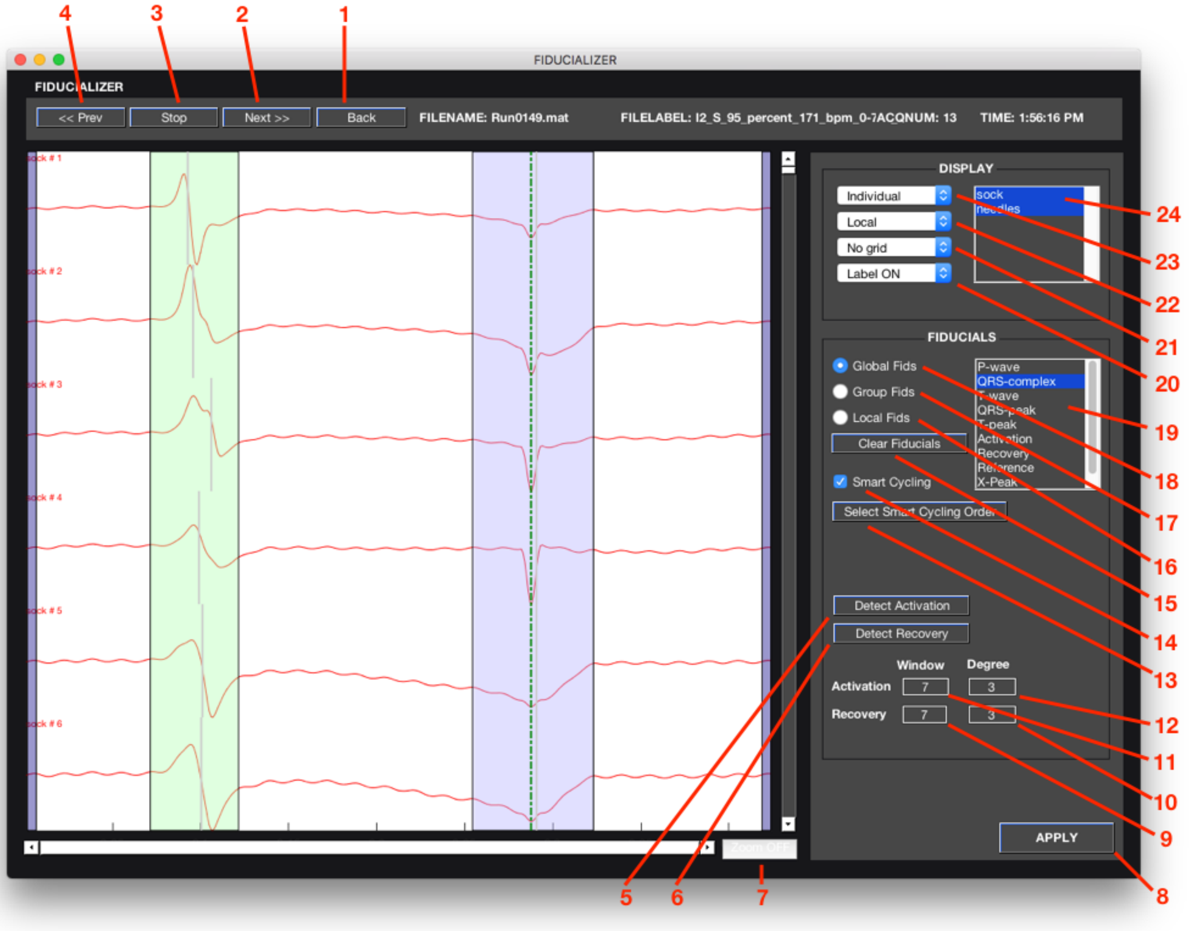
● Global Fids
○ Group Fids
○ Local Fids

Baseline

Figure 6.5: Confirm Beat Envelope Window

If user interaction is selected for the baseline correction a window titled "confirm beat envelope" will appear. This window allows the user to modify the start and end point of the beat and determine which parts of the signal should be interpreted in the baseline correction filter as isoelectric points. The default window has two columns in the display signal that correspond to the two regions of the signal that will be compared for baseline correction. The length of these columns can be adjusted in the processing script settings window under baseline width. Using the cursor the user can click and drag to move these columns to the desired location. If the Group Fids or Local Fids buttons to the right of the display are pressed, the user can select the start and end points for the baseline correction on a group by group or lead by lead basis. The user must be in the 'Individual Fids' or the 'Group Fids' display view in order to be able to chose individual or group fiducials. (figure 6.5).

6.8 Fiducializer



FILENAME: Run0149.mat FILELABEL: I2_S_95_percent_171_bpm_0-7ACQNUM: 13 TIME: 1:56:16 PM

	Window	Degree
Activation	7	3
Recovery	7	3

Figure 6.6: Fiducializer Window

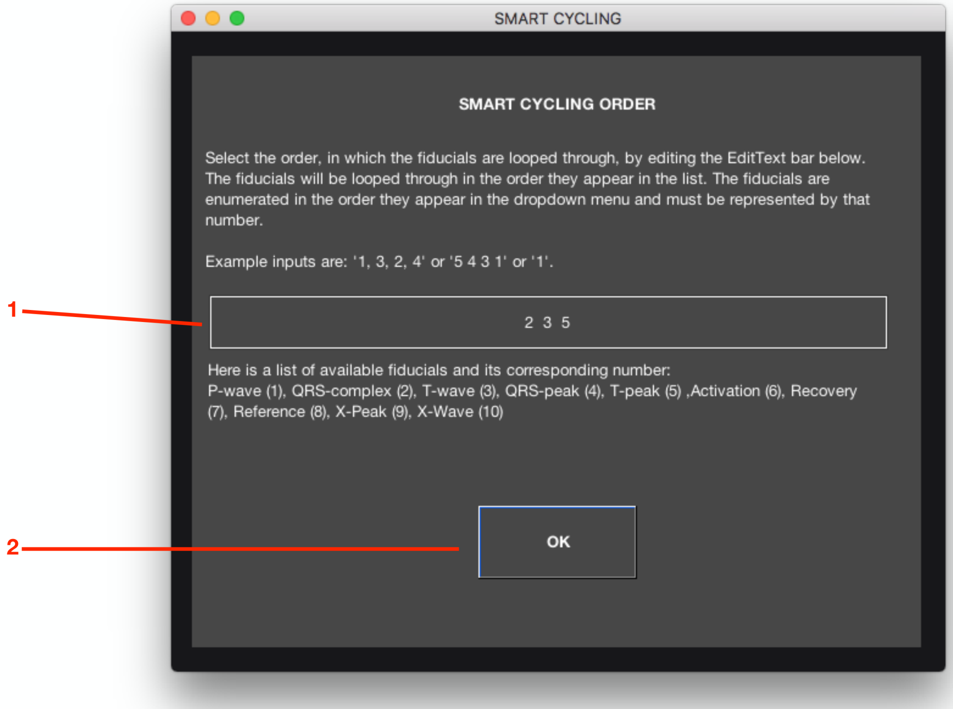
If user interaction is selected for 'Detect Fiducials' a window titled "fiducializer" will appear after pressing 'apply'. This window allows the user to select which regions of the processed signal correspond to specific fiducial points within the signal.

6.8.1 Fiducials

The user is able to select from a list of fiducials to be identified with the cursor. The possible fiducials to be selected from the signal are p-wave, QRS complex, t wave, QRS peak, t peak, activation, recovery, reference, and x peak. Any label of "peak" is a point selection and any label as "wave" is a range of values to be selected within the signal. If a user identifies a common set of signals for each run file and beat the "smart looping" function can be enabled. To do this, the user must select a fiducial cycling order by pressing the 'Select Smart Cycling Order' button and using the new window to make adjustments (figure 6.7. Once the order is set, the user can use the cursor to identify the first fiducial, the next desired fiducial in the smart cycling fiducial order will be automatically set for the next user cursor action. A right click creates a new fiducial point, a left click moves the nearest fiducial point and pressing the delete key deletes the last modified fiducial. (figure 6.6).

6.8.2 Smart Cycling Fiducials

After pressing the 'Select Smart Cycling Order (figure 6.7), the 'Select Smart Cycling Order' window pops up. Here you can enter the order, in which the fiducials are looped through, in a text field. Press 'OK' to confirm your input and return to the fiducializer window.



SMART CYCLING ORDER

Select the order, in which the fiducials are looped through, by editing the EditText bar below. The fiducials will be looped through in the order they appear in the list. The fiducials are enumerated in the order they appear in the dropdown menu and must be represented by that number.

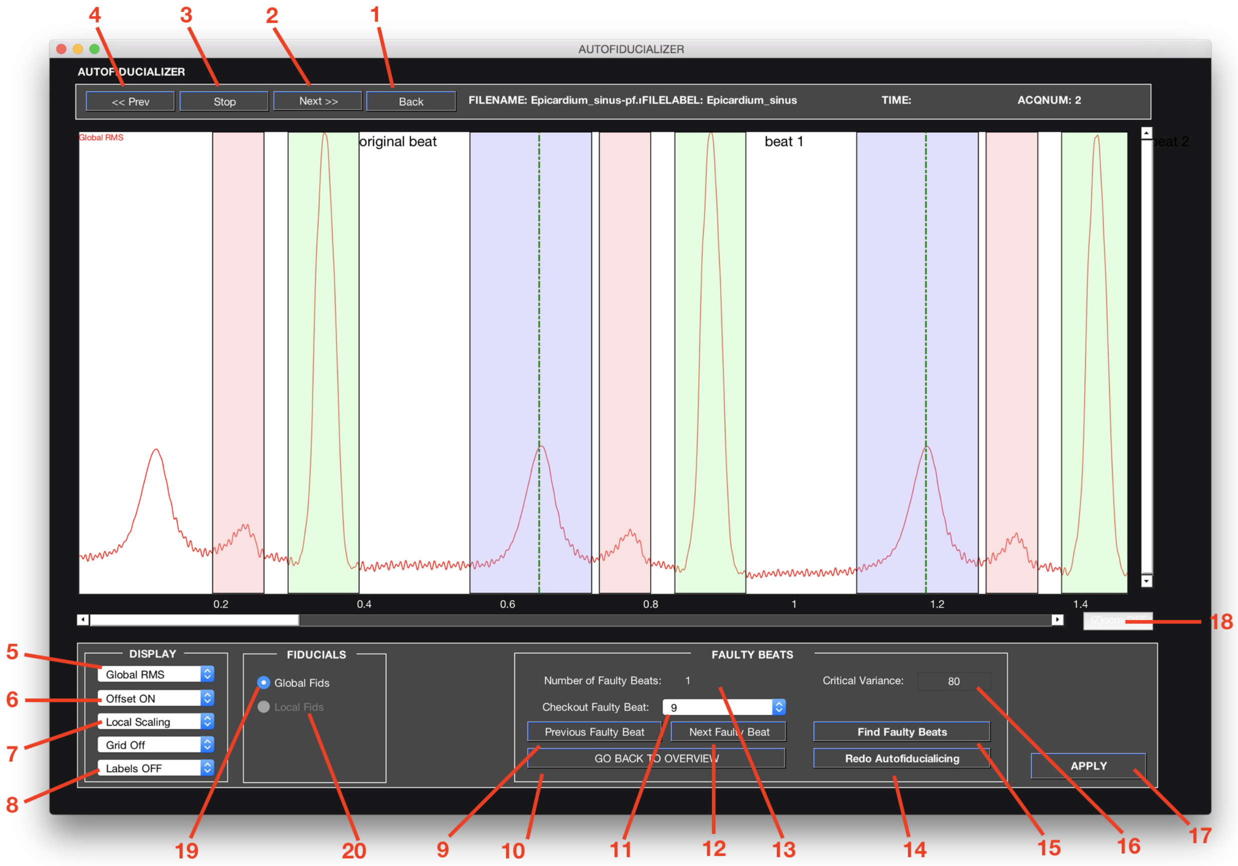
Example inputs are: '1, 3, 2, 4' or '5 4 3 1' or '1'.

2 3 5

Here is a list of available fiducials and its corresponding number:
P-wave (1), QRS-complex (2), T-wave (3), QRS-peak (4), T-peak (5) ,Activation (6), Recovery (7), Reference (8), X-Peak (9), X-Wave (10)

Figure 6.7: Smart Cycling Window

6.9 Autofiducializing



FILENAME: Epicardium_sinus-pf.ıFILELABEL: Epicardium_sinus

TIME:

ACQNUM: 2

Number of Faulty Beats: 1

Checkout Faulty Beat: 9

Critical Variance: 80

Figure 6.8: Autofiducializing Window

If the user has checked the autofiducializing functionality and the user interaction for autofiducializing a window will appear titled "autofiduializer." This window allows the user to review the beats autofiducialized by the autofiducializing algorithm and allows the user to change any fiducials that do not meet satisfactory automatic selection (figure 6.8).

6.9.1 Faulty Beat Review

In the output files, only the global RMS fiducials found by the autofiducializing algorithm will be saved. However, individual fiducials are still computed for each lead. The global fiducial is then computed as the mode of those individual fiducials. If the individual fiducials all have the same value that value is then applied for the global fiducial (figure 6.8-9-16).

6.9.2 Global vs. Local Autofiducials

There are several settings in bot the data organization and autofiducializer window that can be used to fine tune the algorithm to accommodate different signals. These settings, (Figure 6.1-15,8,26,27,29-34, 6.8-19,20) allow the user to change fiducial kernel length, window length, number of leads used for autoficializing, specific leads to be used, using the RMS signal, number of beats before the algorithm is re-run, and number of beats used for updating. All of these have stored best default we have found work for detecting our specific electrogram signals. Options of local fiducializing and updating templates can be checked as well.

Chapter 7

Workbench Tool Overview

7.1 Importing Your Own Tool

PFEIFER allows the user to import their own functions for to use for specific applications as long as they do not need additional functional inputs beyond what the default tools use. To import your own functions navigate to PFEIFER folder ${Local_PFEIFER_Copy}/SOURCE/TOOLS/ and save your function within the relevant tool folder for it to be present on the PFEIFER workbench when it is initialized next. Details on the default tool functionality can be found in the following sections as well as the inputs and outputs required by PFEIFER in order for the tools to function.

7.2 Temporal Filter

Temporal filters are typically employed to reduce noise and/or undesirable components from signals. The default temporal filter used in PFEIFER leverages MATLAB 's built in filter function and applies a rational transfer function with the following coefficients:

- A = 1;
- B = [0.03266412226059 0.06320942361376 0.09378788647083 0.10617422096837 0.09378788647083 0.06320942361376 0.03266412226059];

7.2.1 Required Inputs & Outputs

Input:

- **unfiltered_potvals:** a numLead x numTimeframes array containing the unfiltered time series

Output:

- **filtered_potvals:** a numLead x numTimeframes array containing the now filtered time series

7.3 Baseline Correction

Baseline correction is used to correct the wandering of the isoelectric point (what should be 0 mV) that commonly occurs in biological recordings. We can leverage the premise that in ECGs time instances prior to the Pwave and after the Twave should read 0 mV therefore we approximate the function dictating the drift and subtract it from the signal. This subtraction returns the signal to its appropriate morphology without the presence of drift. The default baseline correction algorithm fits a line to the two user selections of the isoelectric points that bound the beat. The values of the fitted function are then subtracted in order to return the signal to its appropriate baseline.

7.3.1 Required Inputs & Outputs

Input:

- **potvals:** a numLead x numTimeframes array containing the time series that you want to baseline correct
- **startframe:** the index of the timeframe where baseline correction begins. startframe can be in one of the following formats:
  - an integer between 1 and numTimeframes: in the case that all leads are isoelectric at the same time instant.
  - a numLeads x 1 integer array with values between 1 and numTimeframes. These are the individual startframes for each lead
- **endframe:** the index of the timeframe where baseline correction ends. endframe can be in one of the following formats:
  - an integer: in the case that all leads are isoelectric at the same time instant.
  - a numLeads x 1 integer array with individual endframes for each lead.
- **baselineWidth:** an integer indicating over how many timeframes the startframes and the endframes is averaged.

Output:

- **baseline_corrected_potvals:** a numLead x numTimeframes array containing the time series that are now baseline corrected

7.4 Activation Time Finder

Finding activation times is a routine processing step for cardiac electrograms or ECGs. The default algorithm employed by PFEIFER uses the fiducialized QRS bounds and finds the minimum dV/dt within the bounds.

7.4.1 Required Inputs & Outputs

Input:

- **potvals:** a 1 x numTimeframes double array containing the part of the time series of one lead where the activation/recovery is. Activation uses the QRS as defined by the user (or autofiducialization).
- **window:** controls the fitting of the polynomial.
- **degree:** sets the degree of the polynomial being fit.

Output:

- **X:** a double containing the activation time

7.5 Recovery Time Finder

Finding recovery times is a routine processing step for cardiac electrograms or ECGs. The default algorithm employed by PFEIFER uses the fiducialized Twave bounds and finds the maximum dV/dt within the bounds.

7.5.1 Required Inputs & Outputs

Input:

- **potvals:** a 1 x numTimeframes double array containing the part of the time series of one lead where the activation/recovery is. Recovery uses the Twave as defined by the user (or autofiducialization).
- **window:** controls the fitting of the polynomial.
- **degree:** sets the degree of the polynomial being fit.

Output:

- **X:** a double containing the recovery time

Chapter 8

Example Data Processing Pipeline

8.1 Initializing PFEIFER

Before loading signals into PFEIFER there are a number of mandatory benchmarks the data must meet as stated in 'Input Data Overview'. In addition there are further optional steps the user can take before loading data into PFEIFER . First, the user should concatenate all files that were recorded simultaneously. If files recorded on different electrode systems (*e.g.*,torso surface electrodes and a sock electrode system) are stored in separate files then they should be combined to be processed at once. The second step is to package your data in a TS file structure as indicated in TS Structure section. PFEIFER will create this structure for the user but if more metadata was required it would be best to do this ahead of loading PFEIFER .

1. Generating Helper Files
   - Once you have loaded PFEIFER for the first time in an experiment both the Data Organization window and the Workbench should be empty. The user should generate the two helper files as seen in figure 6.1-1,2. This is done by pressing save next to the respective data field and navigating to the directory in which the user would like to save the helper files. It is suggested that these are saved with the experiments as they store relevant information about how each 'Run' is processed.
2. Converting to a TS structure
   - The user may skip this step if you have structured your data into a TS structure.
   - Selecting the 'File Converter' button on the workbench window will open a new window. From this window you can select an input and output directory. Point the input towards your unconverted .mat files and the output towards where you want to store the raw converted files. Note: they will have a -pf appended on the end of the original filename. Select the files in which the user would like to convert and hit 'Convert', see figure 6.3-24. You may close the window.
3. Splitting 'Run's
   - We suggest splitting your files into lengths of time over which the signal does not significantly change. For acute myocardial ischemia we found 15 seconds is adequate in which the autofiducializing function works accurately.
   - To split PFEIFER compatible files the user must point the input directory in the Data Organization window at the directory containing the files you wish to split. After doing this the files should show up in the file list in the Workbench window, the user can now select

the 'File Splitter' option, also on the Workbench (figure 6.3-25). In this new screen select the folder to store the split files. Note: These split files will store the metadata of the parent files. Select all of the files you would like to split and hit 'Split'.

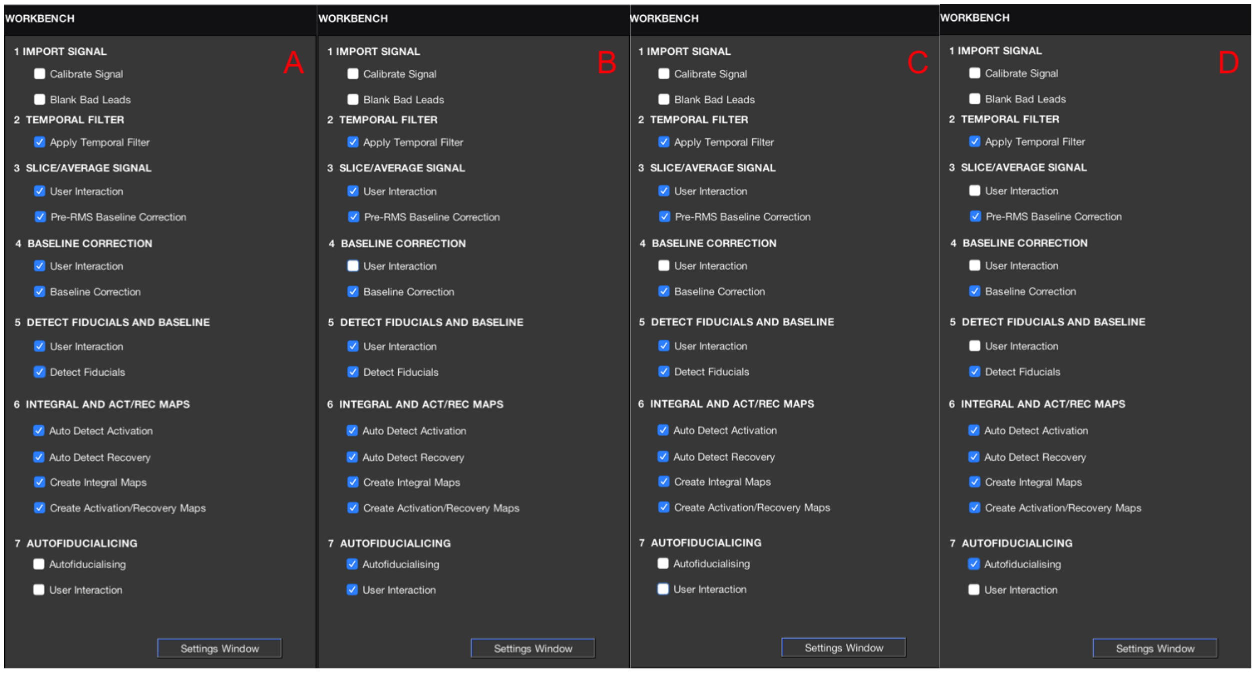
4. Data I/O
   - Input and Output Directories:
     - As can be seen in figure 6.1-3,4 there are fields that allow the user to specify what directory their unprocessed signals are stored and where they would like the output files to be stored. The user must specify the input directory before splitting their files.
   - Sample Frequency:
     - As can be seen in figure 6.1-7 PFEIFER allows the user to set the sampling frequency their data was acquired at.
   - Setting up Rungroups:
     - As can be seen in figure 6.1-11-15 PFEIFER allows the user to separate 'Run's captured using different measurement systems in the same experiment. For example, in Utah experiments we typically record some sock potentials before placing needle electrodes. Therefore all 'Run's that correspond to beats captured before the needle electrodes are included correspond to one Rungroup and the runs taken after the insertion of needles correspond to a second Rungroup. If the measurement system did not change throughout the experiment just assign all of the 'Run's to a single rungroup. These files can be selected using 'Select Rungroup Files' (see figure 6.1-15) or can be entered manually in the field seen in figure 6.1-14. New rungroups can be created by going to the dropdown menu that says 'NEW RUNGROUP' and selecting 'NEW RUNGROUP' as seen in figure 6.1-12.
   - Setting up Groups:
     - As can be seen in figure 6.1-16-25 PFEIFER allows the user to separate electrodes within the 'Run's belonging to a common rungroup. When the groups are saved they are stored separately with the tag specified by the user. The user can also specify the group tag using the field located in figure 6.1-18 'Filename Extension'. This extension corresponds to the group as named by the user in the 'Group Name' field in figure 6.1-16. The user may create new groups within an already generated rungroup by going to the dropdown menu as seen in figure 6.1-20 and selecting 'NEW GROUP'. The button seen in figure 6.1-25 removes the currently selected group. For example, in Utah torso tank experiments we typically have three groups [Torso Electrodes, Sock Electrodes, and Needle Electrodes] with the extensions [-ts,-cs,and -ns]. Since all of these electrodes are stored in a single potvals structure we must let PFEIFER know which electrodes correspond to each of these groupings. For a correctly mapped file these number will simply correspond to the row number within the matrix.

8.2 Running PFEIFER Using Basic User Settings

8.2.1 Choosing PFEIFER Workbench Tools

See Figure 8.1 showing what settings can and should be used for basic and expert users. Note: some settings have to be selected the first time the signal is processed but can be deselected after the first processing step.

- Basic User Settings



A

WORKBENCH

1 IMPORT SIGNAL
□ Calibrate Signal
□ Blank Bad Leads
2 TEMPORAL FILTER
☑ Apply Temporal Filter
3 SLICE/AVERAGE SIGNAL
☑ User Interaction
☑ Pre-RMS Baseline Correction
4 BASELINE CORRECTION
☑ User Interaction
☑ Baseline Correction
5 DETECT FIDUCIALS AND BASELINE
☑ User Interaction
☑ Detect Fiducials
6 INTEGRAL AND ACT/REC MAPS
☑ Auto Detect Activation
☑ Auto Detect Recovery
☑ Create Integral Maps
☑ Create Activation/Recovery Maps
7 AUTOFIDUCIALICING
□ Autofiducialising
□ User Interaction

Settings Window

B

WORKBENCH

1 IMPORT SIGNAL
□ Calibrate Signal
□ Blank Bad Leads
2 TEMPORAL FILTER
☑ Apply Temporal Filter
3 SLICE/AVERAGE SIGNAL
☑ User Interaction
☑ Pre-RMS Baseline Correction
4 BASELINE CORRECTION
□ User Interaction
☑ Baseline Correction
5 DETECT FIDUCIALS AND BASELINE
☑ User Interaction
☑ Detect Fiducials
6 INTEGRAL AND ACT/REC MAPS
☑ Auto Detect Activation
☑ Auto Detect Recovery
☑ Create Integral Maps
☑ Create Activation/Recovery Maps
7 AUTOFIDUCIALICING
☑ Autofiducialising
☑ User Interaction

Settings Window

C

WORKBENCH

1 IMPORT SIGNAL
□ Calibrate Signal
□ Blank Bad Leads
2 TEMPORAL FILTER
☑ Apply Temporal Filter
3 SLICE/AVERAGE SIGNAL
☑ User Interaction
☑ Pre-RMS Baseline Correction
4 BASELINE CORRECTION
□ User Interaction
☑ Baseline Correction
5 DETECT FIDUCIALS AND BASELINE
☑ User Interaction
☑ Detect Fiducials
6 INTEGRAL AND ACT/REC MAPS
☑ Auto Detect Activation
☑ Auto Detect Recovery
☑ Create Integral Maps
☑ Create Activation/Recovery Maps
7 AUTOFIDUCIALICING
□ Autofiducialising
□ User Interaction

Settings Window

D

WORKBENCH

1 IMPORT SIGNAL
□ Calibrate Signal
□ Blank Bad Leads
2 TEMPORAL FILTER
☑ Apply Temporal Filter
3 SLICE/AVERAGE SIGNAL
□ User Interaction
☑ Pre-RMS Baseline Correction
4 BASELINE CORRECTION
□ User Interaction
☑ Baseline Correction
5 DETECT FIDUCIALS AND BASELINE
□ User Interaction
☑ Detect Fiducials
6 INTEGRAL AND ACT/REC MAPS
☑ Auto Detect Activation
☑ Auto Detect Recovery
☑ Create Integral Maps
☑ Create Activation/Recovery Maps
7 AUTOFIDUCIALICING
☑ Autofiducialising
□ User Interaction

Settings Window

Figure 8.1: This figure shows suggested configurations for running PFEIFER . **A.)** Basic user settings. **B.)** Expert User settings for a small number of 'Run's. **C.)** Expert User configuration for a large number of 'Run's. This is for the first pass through the experiment. A second pass with the settings in D should then be used. **D.)** Expert Users who have manually fiducialized all of their 'Run's and would now like to autofiducialize the rest of their beats. Note: all user interaction has been turned off.

  - We suggest the user implement these settings if they do not have a lot of experience running PFEIFER . It prevents the user from using the autofiducialization function and asks the user to confirm the beat envelop.

- Expert User Settings (Small Amount of Files)

  - This is the configuration we suggest using if the user has some experience operating PFEIFER but has a small number of files they wish to process (less than 10 files). The reason for this distinction is that autofiducialization can take a few seconds per file, depending on the length, so the user may not want to sit idle at the end of processing each file. For a small amount of files it makes sense to do the autofiducialization after the manual fiducialization.

- Expert User Settings (Large Amount of Files)

  - This is the configuration we suggest using if the user has some experience operating PFEIFER but a large number of files they wish to process. If this is the case, we suggest manually fiducializing all of the files first and then switching got the configuration in D to do the autofiducialization.

- Autofiducializing Only

  - This is the configuration we suggest using if you have already manually fiducialaized one beat from each file and want to apply those detected fiducials to each subsequent beat within each run file. If user interaction is turned off in section 7 the program will automatically move through the files and save the autofidcuialized beats. Turning on 'User Interaction' will allow the user to review the autofiducialized beats after each file is processed.

8.2.2 Beat Isolation

- Selecting a Representative Beat
  - This is done in the 'Beat Isolator' window. This window under default settings shows the RMS signal created from all electrodes across all groups. The data being rendered in this window can be changed by going to the display settings dropdown menu (figure 6.4-4) field and selecting 'Group' or 'Individual' from the dropdown box. The window that defines a representative beat can be placed by the user by right clicking (and holding) on an isoelectric point before the beat and dragging the window to encompass the entire beat. These bounds can be edited in this window by left-clicking. A left click will snap the closest bound to the users cursor, so use sparingly. This RMS calculation uses all of the good leads in order to create the RMS signal. You can eliminate bad leads from the RMS calculation by including their electrode number in the 'Bad Leads' field on the Data Organization screen.

8.2.3 Confirm Beat Envelope

- Make Sure the Boundaries are on Isoelectric Points
  - This is the second window that pops up in the PFEIFER workflow. However, user interaction in this window is typically turned off as the only task being performed is a selection of the start and end of the representative beat, which was performed in the previous window. In this window the mouse controls are the same, left click edits an isoelectric bound and right click places a new bound. Pressing 'SPACE' or 'APPLY' will advance to the next window.

8.2.4 Fiducialize Beat

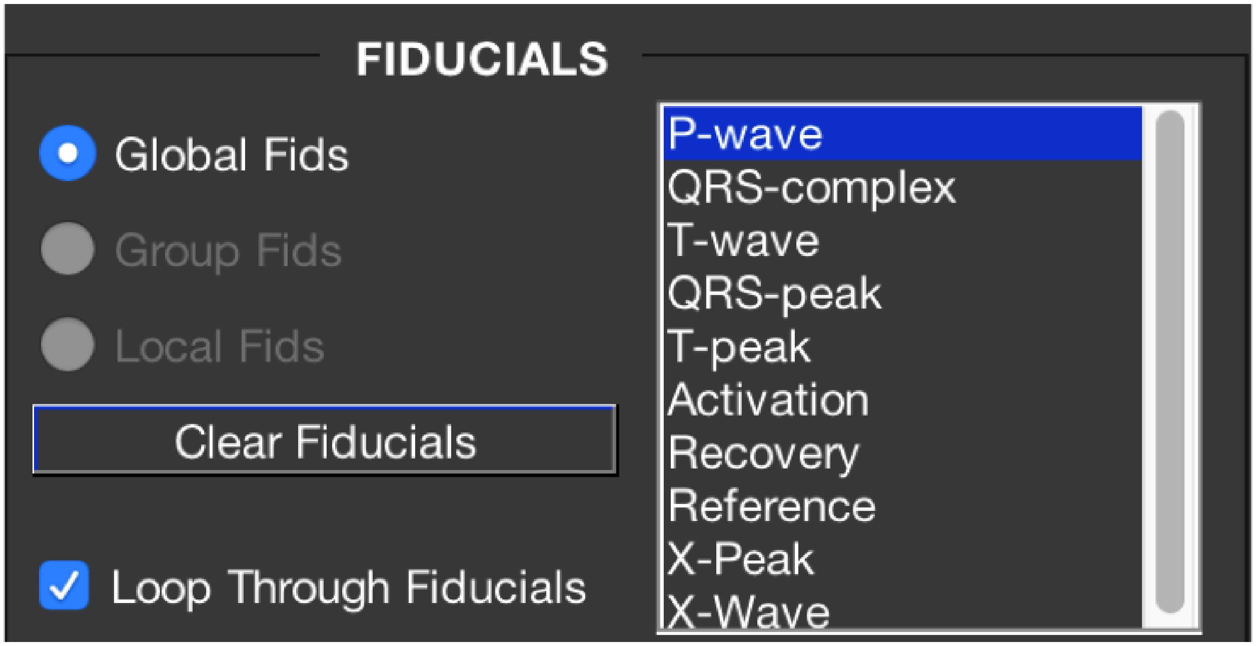


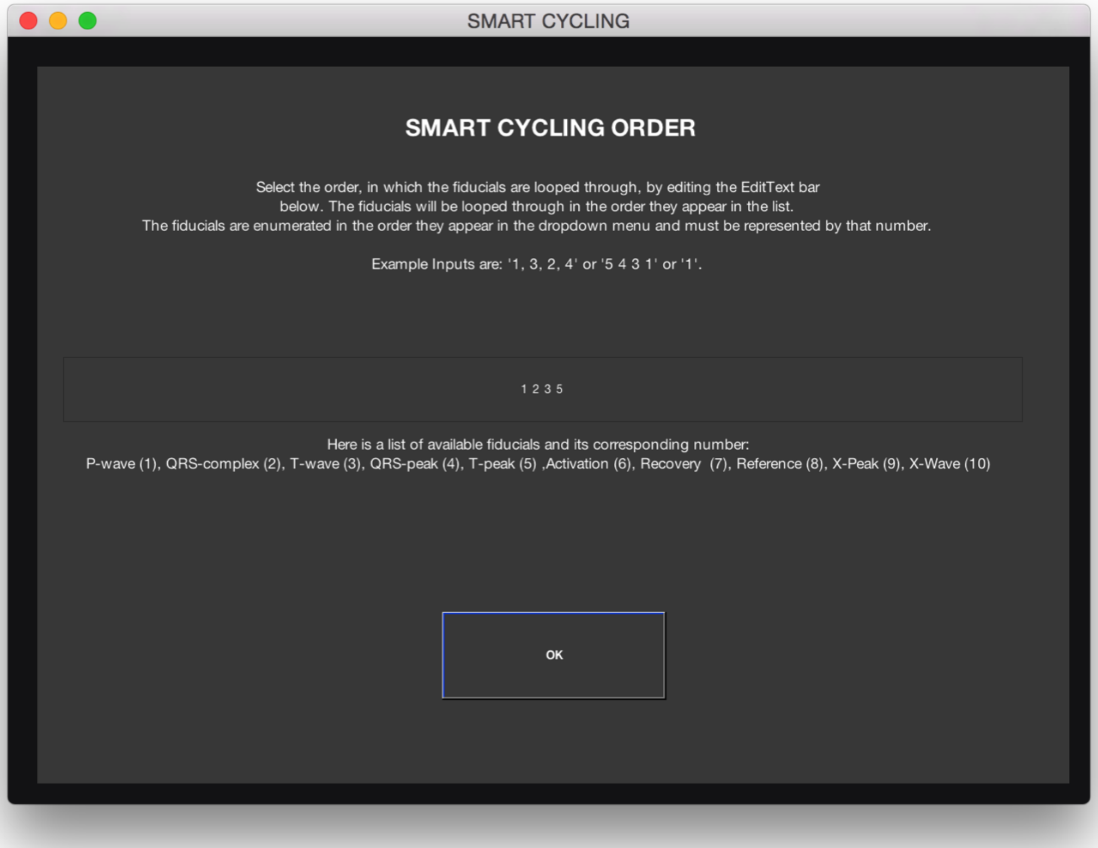
Figure 8.2: This shows the portion of the Fiducializer that allows the user to manually choose the fiducial they wish to place.

- If you are processing many 'Run's you should see the Smart-cycling section

- Fiducializing
  - This window allows the user to place manual fiducials all over the representative beat. If you did not set up you 'Smart-Cycling' of fiducials then you will have to select what fiducials you want to place manually. The P-wave, QRS complex, T-wave, and X-wave place two fiducials each defining the start and end of each fiducial(*e.g.*,QRSon and QRSoff). If the user selected 'Create Integral Maps' and 'Create Activation/Recovery Maps' then the user has to at least place a QRS-complex, a T-wave, and a T-peak fiducial (figure 8.2).

8.3 Using the Expert User Functionality

8.3.1 Smart Cycling Fiducials



SMART CYCLING ORDER

Select the order, in which the fiducials are looped through, by editing the EditText bar
below. The fiducials will be looped through in the order they appear in the list.
The fiducials are enumerated in the order they appear in the dropdown menu and must be represented by that number.

Example Inputs are: '1, 3, 2, 4' or '5 4 3 1' or '1'.

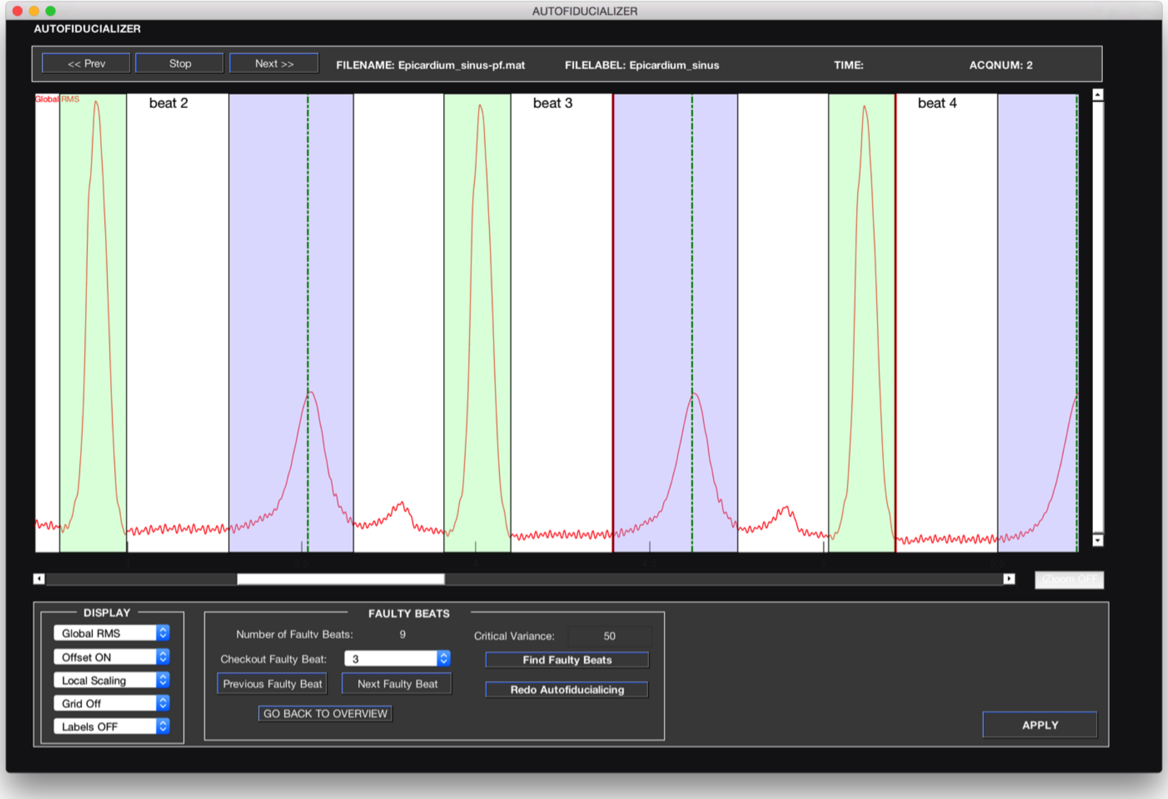
1 2 3 5

Here is a list of available fiducials and its corresponding number:
P-wave (1), QRS-complex (2), T-wave (3), QRS-peak (4), T-peak (5) ,Activation (6), Recovery (7), Reference (8), X-Peak (9), X-Wave (10)

Figure 8.3: This shows the portion of the Fiducializer that allows the user to manually choose the fiducial they wish to place.

This window allows the user to specify the order they would like to manually place fiducials so that they do not need to select the fiducials each time from the selection window. Setting up smart cycling can reduce the manual processing time in half on a 'Run' by 'Run' basis. Right clicking places a new fiducial, therefore after each right click has been released the next fiducial in the specified order will be automatically selected. In the example shown in the figure above the smart-cycler will automatically allow the user to place the P-wave, then the QRS complex, then the T-wave, and finally the T-peak (figure 8.3).

8.3.2 Autofiducialization


AUTOFIDUCIALIZER

FILENAME: Epicardium_sinus-pf.mat

FILELABEL: Epicardium_sinus

TIME:

ACQNUM: 2

DISPLAY

Global RMS

Offset ON

Local Scaling

Grid Off

Labels OFF

FAULTY BEATS

Number of Faultv Beats: 9

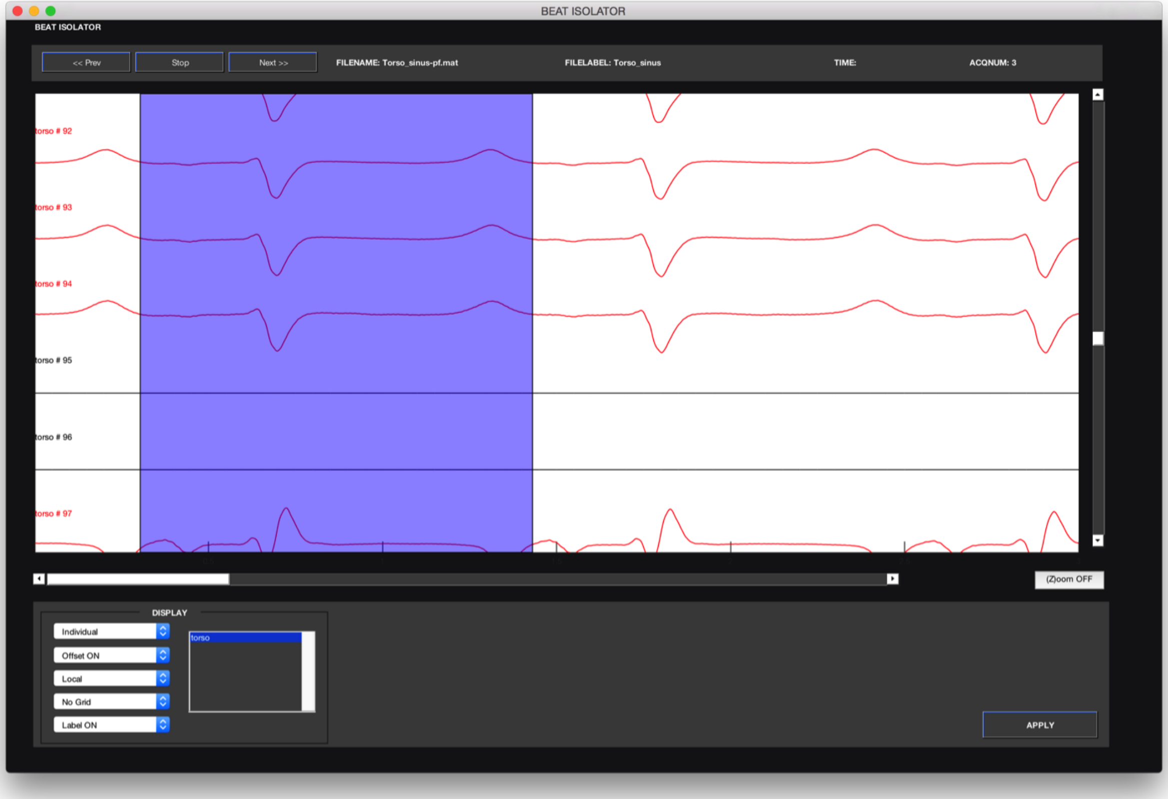
Critical Variance: 50

Checkout Faulty Beat: 3

Figure 8.4: This shows the portion of the Fiducializer that allows the user to manually choose the fiducial they wish to place.

This window pops up when the user selects 'User Interaction' from the toolbox under 'autofiducializer'. This window shows the manually and automaticly placed fiducials in one continuous stream on time signals. 'Original Beat' shows the user's manually fiducialized beat and each subsequent beat 'Beat 2, Beat 3, etc...' are the automatically fiducialized beats. The user can review these beats all at once using the slider bar at the bottom of the time signal window and left clicking to modify the nearest fiducial. Otherwise, the user can select 'Next Faulty Beat' to individually review each of the autofiducialized beats. Hitting APPLY will save all relevant output files (figure 8.4).

8.3.3 Bad Lead Identification



FILENAME: Torso_sinus-pf.mat

FILELABEL: Torso_sinus

TIME:

ACQNUM: 3

Figure 8.5: This shows the portion of the Fiducializer that allows the user to manually choose the fiducial they wish to place.

This is a modified view of the 'Beat Isolator' window showing individual electrode signals instead of the global RMS signal. This window can be used to identify which electrodes have signals that you would like to exclude from the RMS calculation and autofiducialization to make the final results more accurate. To exclude leads simply type the electrode numbers, or ranges, that correspond to bad electrodes. You can tell these have been removed from the calculation by the color rendered to the signal. These signals will be unaltered in the output files unless 'Blank Bad Leads' is selected from the toolbox. If this is left unselected the output data will include values from the bad lead electrodes (figure 8.5).

8.4 Expected Output Files

The files output by PFEIFER depend on what selections were made from the toolbox and how many groups each 'Run' contained. A typical 'Run' from a Utah torso tank experiment will have 3 groups with each run split into 15 second long segments. An example 'Run' from these experiments will produce the following files"

- 3 'Run's containing the manually fiducialized and isolated beat, one for each group specified by the user across all electrodes in that group. These are identified based on the user selected 'Filename Extension'.(*e.g.*,Run0001-ns.mat, Run0001-cs.mat, Run0001-ts.mat)

- 3 Integral Maps and 3 Activation Recovery Maps are created per ‘Run’ with 3 user specified group. The tags -itg and -ari are indicative of the integral and activation recovery maps respectively.(*e.g.,*Run0001-ns-itg.mat, Run0001-cs-itg.mat, Run0001-ts-itg.mat,Run0001-ns-ari.mat, Run0001-cs-ari.mat, Run0001-ts-ari.mat)
- 1 ‘Run’ per user selected group per beat remaining in the ‘Run’ . For a 15 second ‘Run’ anywhere from 15 to 50 beats may be extracted in addition to the single user fiducialized beat. Each of these autofiducialized beats will have an additional -b# tag, with the number symbol replaced with the index of that beat within the ‘Run’ (*i.e.,*the 5th beat will have the tag -b15). For each of these autofiducialized beats there will also be corresponding integral and activation recovery maps with each of the additional files. (*e.g.,*Run0001-b1-ns-itg.mat, Run0001-b1-cs-itg.mat, Run0001-b1-ts-itg.mat,Run0001-b1-ns-ari.mat, Run0001-b1-cs-ari.mat, Run0001-b1-ts-ari.mat)

8.4.1 Structure of the Output Files

The TS structure of the output files contain many of the same fields with some variants. The relevant details stored within each file type are explained below. Each fiducial has a corresponding integer value that identifies it. For example 2 and 4 define the start and end of the QRS complex. The full fiducial set is defined as:

- **Start of Pwave** = 0
- **End of Pwave** = 1
- **Start of QRS** = 2
- **Peak of QRS** =3
- **End of QRS** = 4
- **Start of Twave** = 5
- **Peak of Twave** = 6
- **End of Twave** = 7
- **Activation Time (stored in a numLeads x1 matrix)**= 10
- **Recovery Time (stored in a numLeads x1 matrix)**= 13
- **Isoelectric Points** = 16
- **Pacing Artifact** = 30
- **Start of Xwave** = 26
- **End of Xwave** = 27
- **Peak of Xwave** = 25

- Manually and Automatically Fiducialized ‘Run’s
  - Each processed ‘Run’ will be an isolated beat across all of the electrodes belonging to that group. The processed ‘Run’s will store the processed potvals in the ts.potvals structure and the relevant fiducials in the ts.fids structure. These processed files are tagged with the user specified tags per group.

- Integral Map
  - Each integral map stores the area under the curve on an electrode by electrode basis between certain user specified fiducials. These integrals are stored in the ts.potvals structure within the -itg files. This ts.potvals is numLeads x 5 with each column representing an integral with different bounds. The first column corresponds to an integral of the QRS complex as defined by the user. The second is the integral from the start of the QRS to the end of the Twave. The third column corresponds to an integral from the end of the QRS to a point 37.5% between the end of the QRS and the end of the Twave. The fourth column corresponds to an integral from the end of the QRS to a time point 80 time frames into the ST segment. The final column is an integral from the end of the QRS to the end of the Twave.
- Activation Recovery Map
  - Each activation-recovery interval (ari) file stores the ari's on an electrode by electrode basis in the potvals structure of the size numLeads x 3. The first column is the estimated activation time for each electrode. The second column is the estimated recovery time measured on each electrode. The third and final column is the difference of the first two columns, indicative of the ari itself.

Chapter 9

Tutorials

Video tutorials can be found in the main PFEIFER page located at: https://www.sci.utah.edu/software/pfeifer.html

Chapter 10

Hotkeys

A table summary of the hotkeys used for PFEIFER are shown below:

Table 10.1: Summary of hotkeys used within PFEIFER

Keyboard Action	Function	Active Windows
Space Bar	Progress to the next user interaction window (Similar to cursor selection "apply")	All Processing Windows
delete/backspace key	Delete a selected fiducial.	All Processing Windows
left/right arrows	move a selected fiducial to the left/right	All Processing Windows
q	Go to previous file. Equivalent to pressing the "Prev" button.	All Processing Windows
w	Go back to main menu. Equivalent to pressing the "stop" button.	All Processing Windows
e	Skip current file and move on to next file. Equivalent to the "next" button.	All Processing Windows
Numbers 1 to 9	Select the corresponding fiducial in the "fiducials listbox".	FIDUCIALIZER window.

Chapter 11

Test Data

To test whether PFEIFER is working, one can run unit tests, contained as part of the PFEIFER distribution. There are two unit tests available at this time: a basic unit test and a file splitting unit test.

11.1 Running A Unit Test

In order to be able to run the unit tests, PFEIFER needs to be added to the MATLAB path after cloning it from GitHub. To run the basic unit test, type 'basic_unit_test' into the command line window. To run the file splitting unit test, type 'file_splitting_unit_test' instead. The unit tests will then run fully automatic and may take up to 30 seconds. If the test was successful, a message indicating that the test was successful will be printed to the screen, otherwise you will see some error messages.

11.2 Basic Unit Test

This basic unit test checks whether all the core functionality of PFEIFER is working. It does so by simulating a typical PFEIFER workflow: It opens PFEIFER and loads a processing data and a script data helper file. Next, some example run files are processed using the settings stored in those helper files. Subsequently the resulting output files are compared to some template output files to see if the resulting output files were correctly processed. All necessary files are part of the PFEIFER release.

During that test processing, all the PFEIFER processing tools are tested. These are the baseline correction, the temporal filtering, the detection of the activations and recoveries, and the autofiducializing.

11.3 File Splitting Unit Test

The file splitting unit test checks whether the file splitting functionality works correctly: It simulates the user starting PFEIFER , loading some test files, opening the file splitting window, selecting some files to split, and splitting the files. Subsequently the output files of that process are compared to some template output files to see if the resulting output files were correctly processed. All necessary files are part of the PFEIFER release.

Chapter 12

Community Contribution and Reporting Guidelines

PFEIFER is an open source software package that is always looking to improve via community interaction. Below are ways that the community can contribute, report bugs, and ask questions to the developers on this project. The open source software is available on github where anyone can clone and download the repository to access the source code. All bug reporting and contributions are done via standard github issue tracking and pull requests.

12.1 Bug Reporting

Bugs can be reported via the "issue tracking" tab located on GitHub. Developers will do their best to respond to your comments as quickly as possible.

12.2 Contributions

PFEIFER developers manage contributions to the software via standard pull request and branch creation git techniques. The user must clone or fork the repository, create a new branch, and submit a pull request to have modifications uploaded to the GitHub repository. Developers will do their best to respond to pull requests as quickly as possible to update PFEIFER to the most current version.

12.3 Community Questions

Developers are available to contact via the PFEIFER mailing list (ceg@sci.utah.edu). They will do their best to respond to your questions as quickly as possible.