

# PrepAnnECG: A User Friendly MATLAB ECG Preprocessing and Annotation GUI for Health Professionals

## 1. Software Requirements

This software only requires a basic Matlab setup. All the functions used for preprocessing uses the essential functions of Matlab. The use of filtering tools along with the `preprocessECG` GUI is optional and requires Signal Processing Toolbox. The `preprocessECG` GUI permits running users' custom preprocessing functions.

## 2. Installation

The software does not requires an installation. Only the folders should be added to Matlab path.

## 3. Folder Organization

The software is organized into several folders. The `src` folder includes `fig` and `m` files for two GUIs, a function `PT_rloc.m` for locating R peaks of ECG signal, and another function to anonymize the original filenames for patient privacy. The preprocessed and segmented `csv` files are stored in `csv` folder. The extra preprocessing functions may be saved into the `custom` folder. The `data` folder includes Holter recordings as text files with anonymized names. Designed filter coefficients as `mat` files can be stored in the `filtercoefs` folder. The annotations created by PrepAnnECG can be saved to the `labels` folder. Finally, the raw Holter recording with original names is stored in the `rawdata` folder. This folder structure is only a suggestion; the desired folders can be accessed from the graphical user interface while loading or saving files.

## 4. Quick Start Guide

The files and their functions are briefly explained in this section.

### 4.1. File Name Annomizer

A sample function for anonymizing the patients' names in the filenames, `anonymize_names.m`, exists in the `src` folder. Assume that the naming notation of ECG recording is as patient name, recording date, and protocol number such as "JANE-DOE-20230531-154157.txt"; this file is renamed as "JD20230531154157.txt". Since every Holter recording device has its own naming notation, this function can be modified. Furthermore, the output naming format can also be changed.

### 4.2. Preprocessing ECG

`preprocessECG.m` and `preprocessECG.fig` are the main files of the software. This GUI includes the following panels for preprocessing functions:

**Load ECG Panel:** The `txt` file, including raw Holter recordings, is browsed by the "Select ECG File" pushbutton. The Sampling Frequency of the signal is entered, and if there are multiple channels in

the signal, the channel is selected from the pop-up menu. When the “Load Signal” pushbutton is entered, the signal is plotted in the axes.

**Signal Statistics Panel:** This panel shows the calculated signal statistics, such as minimum, maximum mean, and standard deviation, after the signal is loaded. These values can guide the following preprocessing process and updated after each signal modification.

**Plotting Axes:** The result of all signal processing steps is displayed in these axes.

**Toolbar:** The Holter readings are long signals of 8 to 24 hours. In order to see the details, Matlab built-in functions such as zoom-in, zoom-out, and pan are inserted in the toolbar. Another valuable tool for pinpointing the necessary parts of the signal is the Datatips for learning magnitude and sample number.

**Message Box:** The messages related to the applied step and the following suggested process are displayed in the Message Box to guide the user about the flow.

**Crop and Remove Artifacts Panel:** Usually, there are distorted signals at the beginning and end of the raw ECG signals, which occur during the insertion and removal of the device. Thus, these parts can be cropped by using this panel. All operations are based on the sample numbers. The exact sample number can be read using Datatips. When the “Crop” pushbutton is clicked, the signal is cropped, and the graph and statistics are updated. Usually, some artifacts exist in the signals when the patient moves or sweats. The starting and ending points of the artifact are entered into the related edit boxes, and these parts can be removed using the Remove pushbutton. Two endings of the removed part of the signal are then merged.

**Custom Code Application Panel:** Many preprocessing methods can be applied to ECG signals with different characteristics. This panel lets the user select external Matlab functions to apply the signals. The “Select Custom Script” pushbutton browses the folders for selecting the preferred m file. Two sample functions are shared in the “custom” folder to represent the format of the custom script. The first argument to the function is the ECG signal. The other arguments are entered into the “Arguments” text box, separated by a space. When the “Apply” button is hit, the graph of the application is shown. The “Reset” button will cancel the application, while clicking the “Save” button will update the stored signal.

**Filter Design and Application Panel:** The most critical step of the ECG signal preprocessing is to apply filters. This panel lets the user freely design their custom filters by using the “FilterDesigner” function of the Matlab Signal Processing Toolbox. The “Design Filter” button invokes the FilterDesigner GUI. After designing the filter, the filter object is exported as a mat file to filtercoefs folder. The “Load Filter Coefficients” button browses the folder for predesigned filters. Thus, the same filter can be applied to different ECG signals. As in the custom script application, the filter application results are displayed as a graph after clicking the “Apply Filter” button. If the user is happy with the results, then the filtered signal can be saved, or the signal can be reset. As many filters as the user needs can be applied sequentially.

**Normalization, Peak Location and Segment Panel:** The first function of the last panel is to apply min-max normalization. Most of the machine learning algorithms run effectively for the normalized data. The next step is to find the most essential fiducial point of the signal, R peaks, for aligning the frames. Pan-Tompkins algorithm is the most commonly used QRS detection algorithm [8]. Thus, the Pan-Tompkins algorithm is embedded in preprocessECG GUI. “Locate R Peaks” button denotes R peaks of the signal with red asterix. Machine learning algorithms in different environments, such as Matlab, Python, Java, etc., can be used for analyzing the ECG signals. Therefore, a comma-separated values (csv) table that can be accessed from any environment is

selected as the output file for this GUI. With some simple modifications, the output format can be adapted to any file format thanks to open-source code. The length of each frame is entered as the Segment Length. The signal is segmented and saved to a csv file by aligning the R peak to the center of the frame. The file's name and path are selected in a modal dialog box. In the output file, each row represents a signal segment, and each column includes one sample of that frame.

#### 4.3. Annotate ECG

The second GUI codes are `PrepAnnECG.m` and `PrepAnnECG.fig`. This functionalities of this GUI are explained as follows:

**Toolbar:** The Open symbol calls an open dialog box. The csv files in the folder called csv are listed as default. The other files in the file system can also be browsed. After selecting the file, the first segment of the table is plotted in the graph area. As in preprocessECG GUI, the zoom-in, zoom-out, and datatips tools can also be used for inspecting signals.

**Plotting Area:** One frame of data file is displayed in the plotting area at a time. The annotation is made for the ECG beat aligned at the center of the graph. For a better view, the center of the graph is highlighted in purple. The length of the highlighted area is constant and does not imply QRS complex or the total length of the beat. The file's name and the segment's number are displayed in the area's upper right corner.

**Navigation Buttons:** "Next" and "Previous" buttons move forward and backward along the segments. The graph is updated when one of the buttons is hit. Accordingly, the beat info is updated. The navigation buttons will not change the displayed frame if the first or last segment is reached.

**Label Buttons:** There exists one button for each different type of arrhythmia, such as Atrial Fibrillation (AF), Normal Sinus Rhythm (NSR), Left Bundle Branch Block (LBBB), Right Bundle Branch Block (RBBB), Atrial Premature Beat (APB), Premature Ventricular Contraction (PVC). If the user wants to denote the beat under consideration as unknown, "??" button can be selected. Alternatively, the "Other" label button can be clicked if another type of beat is observed. The labels are shown in the graph area as the legend and stored in a table in the figure handle. Therefore, the labels of the beats will be displayed while navigating through the segments.

**Close Request:** When the Close symbol at the upper right corner of the GUI is clicked, the labels are saved to the label csv file. The phrase "\_label" is appended to the name of the original data file. Each table row includes one label associated with the data segment in the same data file row.

When a new file is opened, if there are saved labels from a previous session, the first unlabeled beat is displayed. This property is crucial since the Holter recordings may include thousands of frames. Previously labeled beats can also be reviewed by using navigation buttons. Clicking another label button will change the stored label of the beat.