Problem Statement and Goals
RwaveDetection

Junwei Lin

January 16, 2025

Table 1: Revision History

Date	Developer(s)	Change
January 16, 2025	Junwei Lin	Creation

1 Problem Statement

R-wave detection is a critical task in Electrocardiogram (ECG) signal processing, serving important purposes such as heart rate calculation, arrhythmia analysis, cardiac conduction evaluation, and heart disease diagnosis. The patient's ECG signal is sampled by the sensor and converted into a digital signal, where we can detect R-waves. However, since the sampling is not under an ideal condition, a large amount of clutter wave is mixed into the original signal, such as electrical noises and utility frequency.

1.1 Problem

A strong filtering algorithm is needed to filter out redundant signals, and then accurately capture the index of R-wave peaks.

1.2 Inputs and Outputs

1.2.1 Inputs

1. Sampling data file, comes from MIT-BIH Arrhythmia Database
2. Sampling frequency of the data file
3. (Optional) Annotated data from cardiologists, used to evaluate program detecting accuracy

1.2.2 Outputs

1. Index of R-wave peaks
2. A graph that shows all the R-wave peaks
3. (Optional) Root Mean Square Error (RMSE) between each detected R-wave peak time and annotated time from cardiologists

1.3 Stakeholders

The stakeholders include but are not limited to patients, doctors, cardiologists, nurses, technicians, medical device manufacturers, researchers, all of whom rely on accurate, reliable, and efficient ECG signal processing for diagnosis, treatment, innovation, and patient care.

1.4 Environment

Windows, Linux, MacOS, and most light-weight embedding systems are able to run this program.

2 Goals

1. Given a single-channel unfiltered ECG signal, find the index of each R-wave peak.
2. Given correct annotated data, calculate the RMSE between each detected R-wave peak time and annotated time.

3 Stretch Goals

Implements a variety of R-wave detection algorithms for users to choose from.

4 Similar libraries

The signal processing module of SciPy can calculate filter parameters like this program partly does, so it can be used as a pseudo-oracles for later testing.

5 Challenge Level and Extras

This is a problem with a general-level challenge. It is not a research project, but it still requires knowledge of IIR and FIR filter design. Additionally, solving this problem requires understanding some algorithms belonging to this field. Doxygen will be widely used to generate the required documentation, which could be a potential extra.