Phase II Report

Ace of Hearts

Mia Giandinoto, Hannah Hagenau, Anna Johnson, Shivani Kulkarni, Dhruv Modi, Sherina Thomas, Matthew Tran

Georgia Institute of Technology

BMED 2250: Problems in Biomedical Engineering

Professors: Dr. Abouelnasr & Dr. Newstetter

October 16, 2020

Introduction

Our team's goal is to create a noninvasive electrocardiogram (ECG) device to recognize and alert users to stress and symptoms of heart attacks through the parameters of low frequency to high frequency ratio within heart rate variability, ST segment elevation/depression, and T wave inversion within ECG signals to mitigate the effects of a potential myocardial infarction (MI). In order to accurately detect MI and measure heart rate variability (HRV) for stress monitoring, the signal input requires several processing steps in order to filter out noise and accurately detect R-peaks. Due to the fact that the device is being used in day to day life and not on a stationary patient, noise is inevitable. Noise present in an ECG signal can interfere with feature extraction and lead to misdiagnosis (Kumar et al, 2020). MI identification requires locating the ST segment and T wave to detect inversions, depressions, and elevations. Filtering the noise out of the signal enhances these features (Serhani et al, 2020). For HRV, accurate RR intervals are necessary for the Low Frequency power to High Frequency power (LF/HF) ratio. Because of the integral roles of R-peak detection and filtering within signal processing, the model posed within this phase addresses the optimization of these two components. Missed R-peaks (false negatives) can be interpolated based on neighboring RR intervals. Detection of ectopic R-peaks (false positives) pose much more of a challenge because the deletion of ectopic beats can alter ECG data and distort the LF and HF power (Li et al, 2019). Thus, an increased importance was placed on specificity for R-peak detection.

The model posed seeks to determine which R-peak detection method optimizes device specificity. In addition, the model also seeks to determine which filtering method maximises signal-to-noise ratio (SNR) difference and best facilitates R-peak detection. Ambulatory ECG data that has been artificially noised is used as the model input. Two digital filters - a Butterworth's low pass filter and wavelet transform - as well as an RC analog filter were modeled. These were the most common filtering methods present in literature. Each filter's respective signal-to-noise ratio difference from the SNR of the original noisy ECG signal was calculated and compared, with larger differences equating to higher quality filtering. Furthermore, different R-peak detection methods were analyzed, and compared to each other through the metrics of sensitivity and specificity.

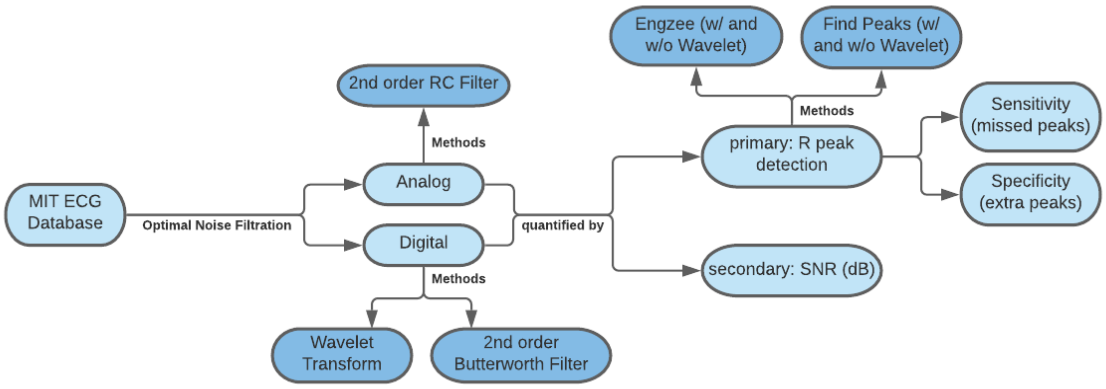
The answer to this modeling question could not be found from research because the optimal filtering and R-peak detection method depends on the context in which the device is used. For instance, one study found that a wavelet transform was the most accurate filtration method, but a Butterworth filter was faster and therefore better for real-time devices (Lenis, 2017). For R-peak detection, one study argued in favor of an adaptive Fourier transform R-peak detection method used on clinical ECG signals (Ze Wang, 2017). However, another study

discussed an empirical mode decomposition (EMD) for R-peak detection in electrocardiogram signals that gave results comparable to ones by the Pan-Tompkins algorithm (Nimunkar, 2007).

Since our device is unique in concept and in purpose (analysis of ambulatory ECG signals for stress analysis and MI detection), our specific criteria could not be matched by any existing research. Thus, a model that catered optimization criteria to our specific device was needed.

Diagrammatic Model

Figure 1: Device Modeling Flowchart

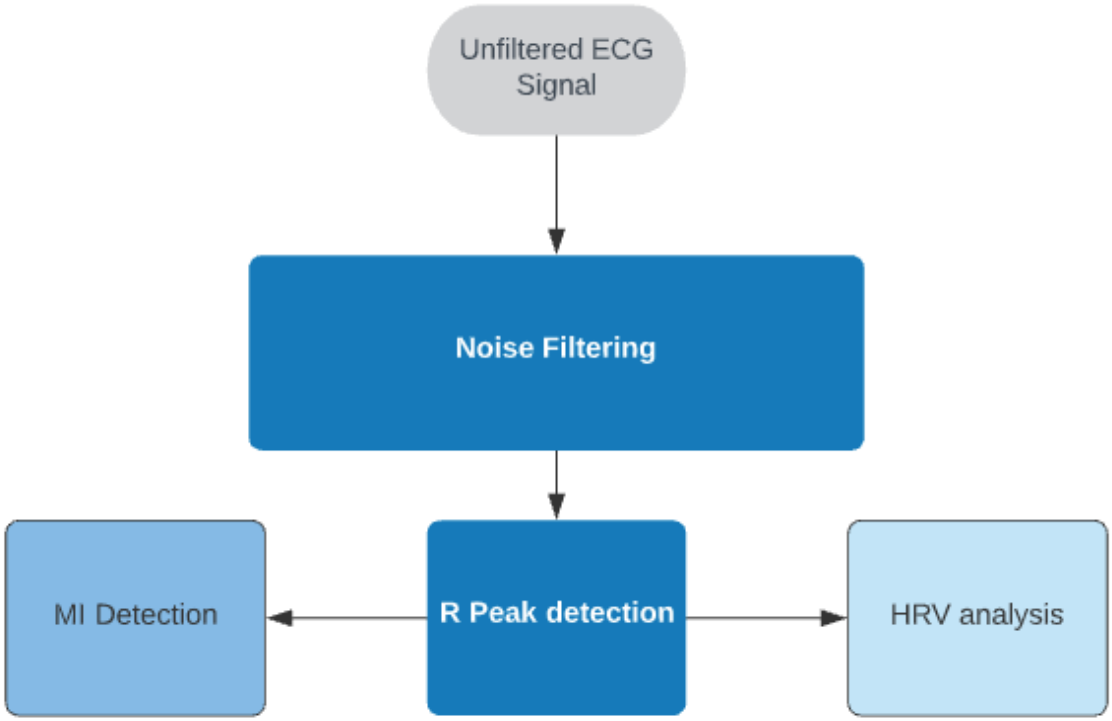


The flow chart above describes the diagrammatic model. The model focuses on optimizing two aspects of the device: noise filtering and R-peak detection. Data from the MIT Noise Stress Test Database is used to model a noisy ambulatory ECG signal. When reading in the input ECG signals, different types of noise such as baseline wander, powerline interference, muscle noise, and electrode motion artifact can corrupt and obscure the signal. Baseline wander is caused by movement of the patient and respiration, and typically contains frequencies below 0.5 Hz. Powerline interference is noise in the form of sinusoidal interference in the 50-60 Hz range, and is caused by the electric components of the device (Vinzio et al, 2012). Muscle noise (EMG) is caused by muscle contractions. The frequency range of EMG partially overlaps that of the ECG, and extends to higher frequencies (Moody et al, 1984). Electrode motion artifact (EM) noise is caused primarily by skin stretching or changes in electrode contact. The frequency range of EM is similar to that of the ECG signal, and can closely resemble R-peaks (Moody et al, 1984). These noise sources complicate R-peak detection and further analysis of the ECG.

The first step of the model was to test three different filters with ambulatory ECG signals from the MIT database. This database contained clean ECG records, to which different types of noise had been added at specific signal to noise ratios (Moody et al, 1984). Using this database, analog and digital filtering methods were applied to the signals. The output of this step is three filtered signals that would ideally have clearly defined R-peaks. Using the parameter of SNR

difference, these filters were tested and numerically compared to determine the effectiveness of each. The three denoised signals, along with the original clean signal, were then tested with the four R-peak detection methods. The second diagrammatic model describes further the R-peak detection component of the model.

Figure 2: R-peak Detection Flowchart



As shown in Figure 2, R-Peak detection is the basis for both the MI detection and HRV analysis. Correctly detecting R-peaks from the unfiltered ECG signal input allows for the computation of accurate RR intervals, which can further be analyzed within a tachogram to extract the LF/HF ratio of HRV. This output parameter is crucial in HRV analysis as a low ratio is indicative of high levels of stress and higher risk of heart disease (Järvelin-Pasanen, 2018). Furthermore, locating the R-peak also serves to identify other important features of the ECG signal, including the ST-segment and the T-wave, which are markers of a possible MI. Thus, without proper R-peak detection, the device would not be able to accurately monitor stress or detect a potential MI. The main R-peak detection methods modeled were the Engzee R-peak detection method along with the Find_Peaks method from Scipy's signal processing library in Python. Both were tested with and without a Wavelet transform creating a total of four methods. They were used on the denoised signals from each filtration method, as well as the original clean signal. The output from these tests could then be evaluated using the parameters of missed peaks and extra peaks detected by the R-peak detection methods.

Due to the time constraints of this phase, the RC Analog filter was modeled digitally using a transfer function within Python. Because of this, a key assumption of the model is that this transfer function replicates the functionality of the true physical analog RC filter. Furthermore, the model assumed that there are no irregularities in the ECG signal such as premature ventricular contraction or arrhythmia. Constraints of this model include using artificially noised ECG data from the MIT Database which may not identically represent the noise present when reading in actual ECG signals from the device.

Mathematical Modeling Methods

Throughout this modeling phase, the team developed and executed two modeling questions that correlate with each other, giving two different types of models to analyze for an end result. To test the performance of the filters, SNR was calculated as a metric to compare how similar the filtered data is to the given, clean ECG data. The SNR was computed in decibels through equation 1 below:

Equation 1: Signal to Noise Ratio

$$10 * \log_{10}\left(\frac{mean\left((signal)^2\right)}{mean\left((signal-cleansignal)^2\right)}\right)$$

The larger the signal to noise ratio, the more it resembles the clean data. An SNR difference was used to compare each filtering method: the original SNR of the noisy ECG data tested was -16.0 so subtracting this value from the calculated SNR gives the signal-to-ratio improvement from the noisy data to the filtered data in decibels. Since a larger SNR is indicative of better filtering, the final parameter of SNR difference also follows this same pattern, where a larger difference equates to higher quality of filtering.

The first digital filter modeled was a Butterworth lowpass filter. A lowpass threshold eliminates all frequencies in a signal above a certain threshold. The majority of noise present in ECG signals that can interfere with R-peak detection comes from higher frequencies. A low pass filter will eliminate this high frequency noise. 30 Hertz was used as a cutoff frequency. This value was applied as it's significantly lower than the frequency of powerline interference (approximately 50-60 Hz). A Fourier Transform of the data was graphed and analyzed at what frequencies the extra noise was being added at, and a value lower than that was used.

The other digital filter was a discrete wavelet transform (DWT). The DWT decomposes the signal into a series of approximation and detail coefficients, each of which contain different

frequency components of the signal. When performing a DWT, a mother wavelet ψ is scaled by a scaling parameter m and translated across the signal by translation parameter n. The mother wavelet chosen or this model was the db6 wavelet, which closely resembles an ECG signal (Lin et al, 2014).

The method employed to select these parameters was multiresolution analysis. In this type of analysis, the scaling and dilation parameters are based on powers of two, and the discretization of the mother wavelet is given by equation 2:

Equation 2: Discretization of the Mother Wavelet

$$\psi_{m,n}(k) = 2^{-m/2}\psi(2^{-m}k - n)$$

The signal is passed through a series of high-pass and low-pass filters, in which the mother wavelet is scaled and translated across the signal to capture different frequencies. This separates the signal into a series of approximation and detail coefficients, which contain both time and frequency information about the signal. The approximation coefficient contains the low-frequency components, and the detail coefficients contain the higher frequencies. These different coefficients can be thresholded and reconstructed in different ways for differing purposes. For filtering, a 9 level DWT was performed. Approximation coefficient cA9, which contains frequencies below 0.5 Hz, removes baseline wander noise. Removal of cD1, which contains the highest frequencies of the signal, removes high frequency noise. In order to remove EMG and electrode motion artifact, all other detail coefficients are thresholded using the universal selection method, using cD1 to estimate the standard deviation of the noise. The threshold is calculated through Equations 3 and 4 (Lin et al, 2014).

Equation 3: Thresholding

$$t = \sigma\sqrt{2\log N}$$

Equation 4: Thresholding

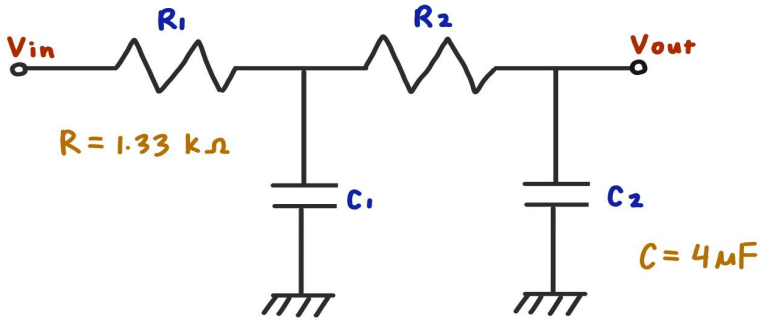
$$\sigma = \frac{median|cD1|}{.6457}$$

The coefficients are thresholded using the soft threshold method, wherein all components of the coefficient array below t are set to zero, and the rest of the components are normalized.

The analog filtering method that was modeled was that of a 2nd order lowpass RC filter. Analog filters differ from digital filters in that they have the benefit of filtering signals before

they are read in by the arduino, allowing for filtering and analyzing without any latency. Analog filters are made up of physical circuit components and this model uses resistors and capacitors as they are very simple and effective to implement into a device. The circuit diagram for the arrangement of the components to be used is shown below in Figure 3 with specified values. The voltage readings enter through V_{in} and the filtered readings, V_{out} , are read and examined from there.

Figure 3: Circuit Diagram of the RC Filter



Without building the device itself due to time and resource constraints for the phase, the filter effectiveness was modeled through a transfer function. Equation 5 shown below is the formula for the transfer function of a 2nd Order Lowpass RC Filter, which is a time-invariant system as it is calculated in the frequency domain.

Equation 5: Transfer Function of 2nd Order Lowpass RC Filter

$$G(s) = \frac{v_0}{v_i} = \frac{1}{\left\{1+(j\omega)[C_1R_1+C_2(R_1+R_2)]+(j\omega)^2C_1R_1C_2R_2\right\}}$$

Equation 6 below is the transfer function simplified with values for the resistors and capacitors with values of 1,330 Ohms and 4.0 micro-Farads respectively. These specific numbers were assigned to create a cutoff frequency of approximately 30 Hertz. This is to match the digital lowpass filter to compare the effectiveness of the two by keeping variables between them the same (“2nd order CR filter”, 2020).

Equation 6: Simplified Transfer Function

$$G(s) = \frac{35332.69}{(s^2+563.91s+35332.69)}$$

These coefficients were implemented into a laplace transform function within Python to convert the time series data into a frequency domain, and graphed these results with a function to simulate the output of a continuous-time linear system.

We assessed our filtering methods primarily through their ability to facilitate R-peak detection. We modeled different R-peak detection methods which are crucial for HRV analysis, quantifying them through sensitivity and specificity. Sensitivity corresponds to missed R-peaks within the detection method and was calculated via Equation 7.

Equation 7: Sensitivity

$$sensitivity = \frac{true\ positive}{true\ positive + false\ negative}$$

Specificity, on the other hand, correlates to extra peaks detected and was calculated through Equation 8.

Equation 8: Specificity

$$specificity = \frac{true\ negative}{true\ negative + false\ positive}$$

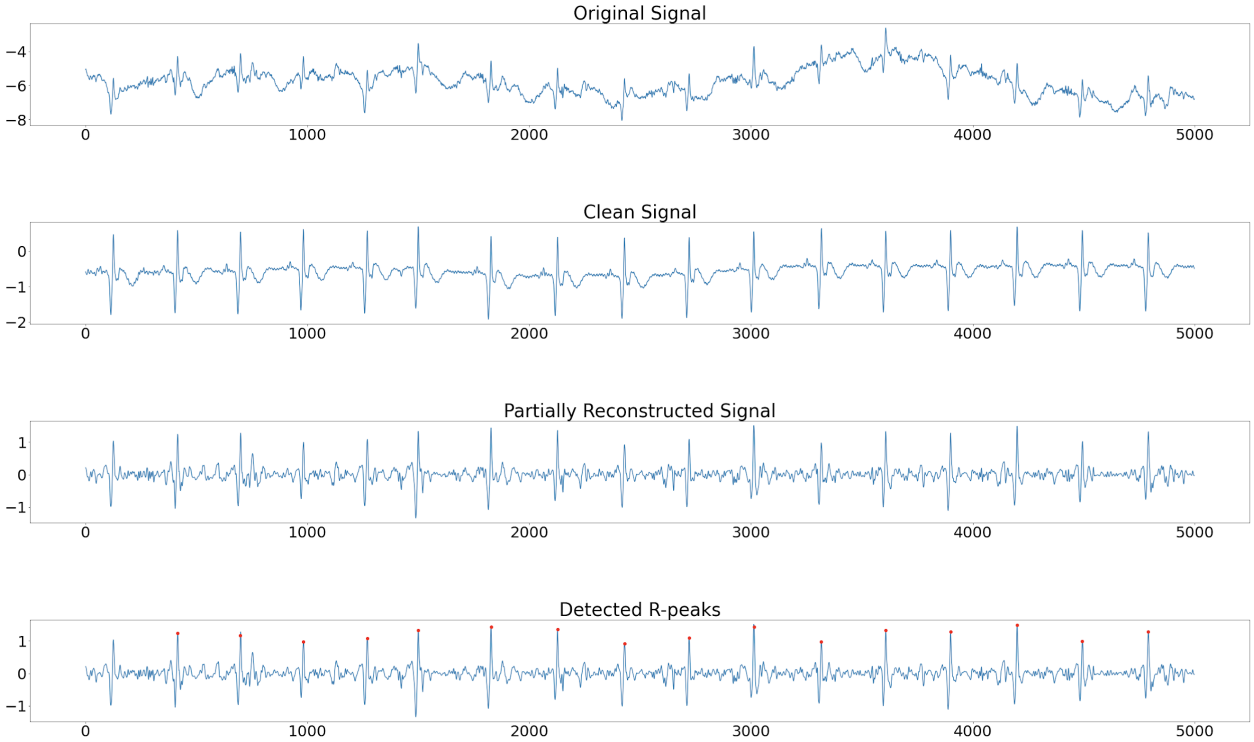
True positives are correctly identified R-peaks, true negatives are all possible local maxima that are not R-peaks that weren't marked, false positives are extra peaks that were detected, and false negatives are missed R-peaks.

Ten seconds of noisy ECG data were used and visually analyzed for missed and extra peaks in comparison to the clean ECG's correct R-peaks. Multiple popular peak detection algorithms were implemented and used with these ten seconds of data, including Christov detection, Pan-Tompkins, and Engzee. Out of these three methods, Engzee was the most accurate in detecting R-peaks in regards to the fewest missed and extra R-peaks, which correlates to optimal sensitivity and specificity. This algorithm works by examining intervals where the amplitude is greater than a predefined threshold (0.6 times the max of the signal). Then, the signals are viewed in 160 millisecond windows to the right of where the signal is less than the negative threshold for at least 10 points, and the R-peak is determined by scanning for the highest amplitude in that window. Another detection method used was the "find_peaks" function within Scipy's signal processing library in Python. This method can be used to detect R-Peaks with given parameters. For each peak to be counted as an R-peak, a minimum distance between each peak of 200 ms and a prominence of 1 mV was defined. The end of S waves are found around 240 ms after an R-peak, so the minimum time before the next R-peak should be detected was set slightly under this value (Rabbani et al, 2011). The prominence threshold was determined by using a loop to substitute values for prominence, and the ones that detected every R-Peak properly were added to an array. 1 mV was the maximum value in this array so it was used as the prominence a peak must have to be considered an R-peak as lower values have the potential to incorrectly identify R-peaks.

For the next methods, a 5 level discrete wavelet transform was performed on the filtered ECG data. The signals were partially reconstructed with detail coefficients cD3,cD4, and cD5 to

isolate the frequencies that contain the R-Peaks as seen in Figure 4 below. This makes the peaks more prominent and allows for more accurate R-peak detection. The Engzee and find_peaks method were then applied to the partially reconstructed data.

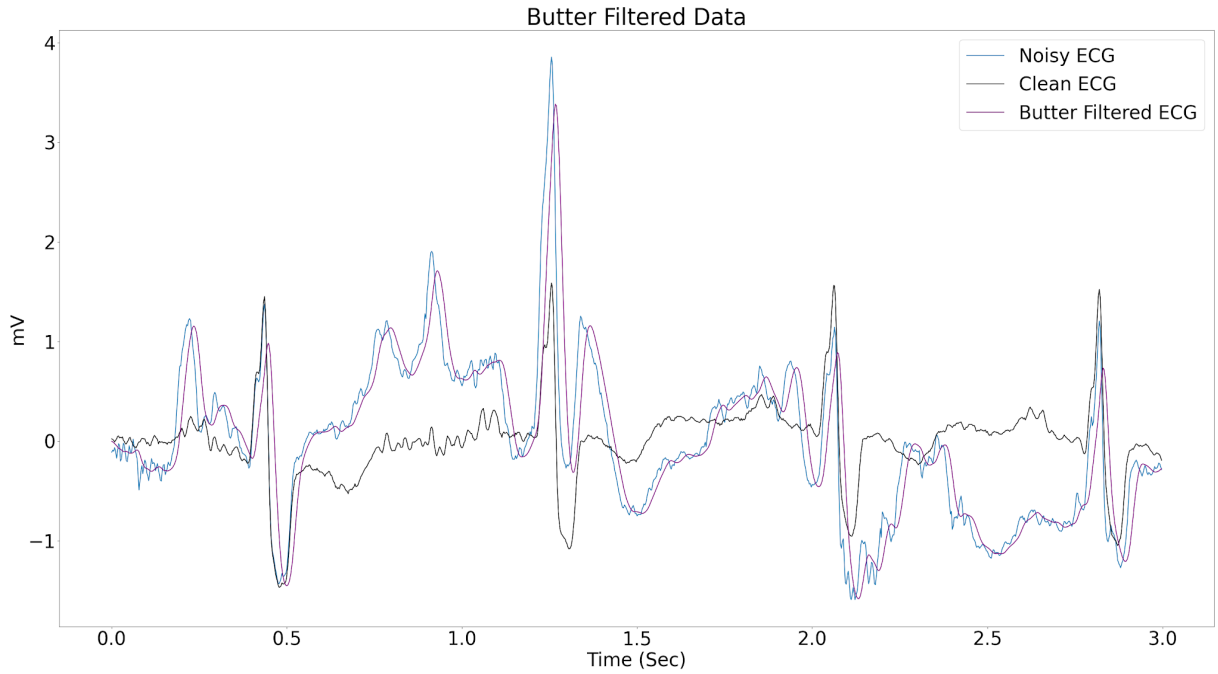
Figure 4: Partial Wavelet Reconstruction



Mathmetical Modeling Results

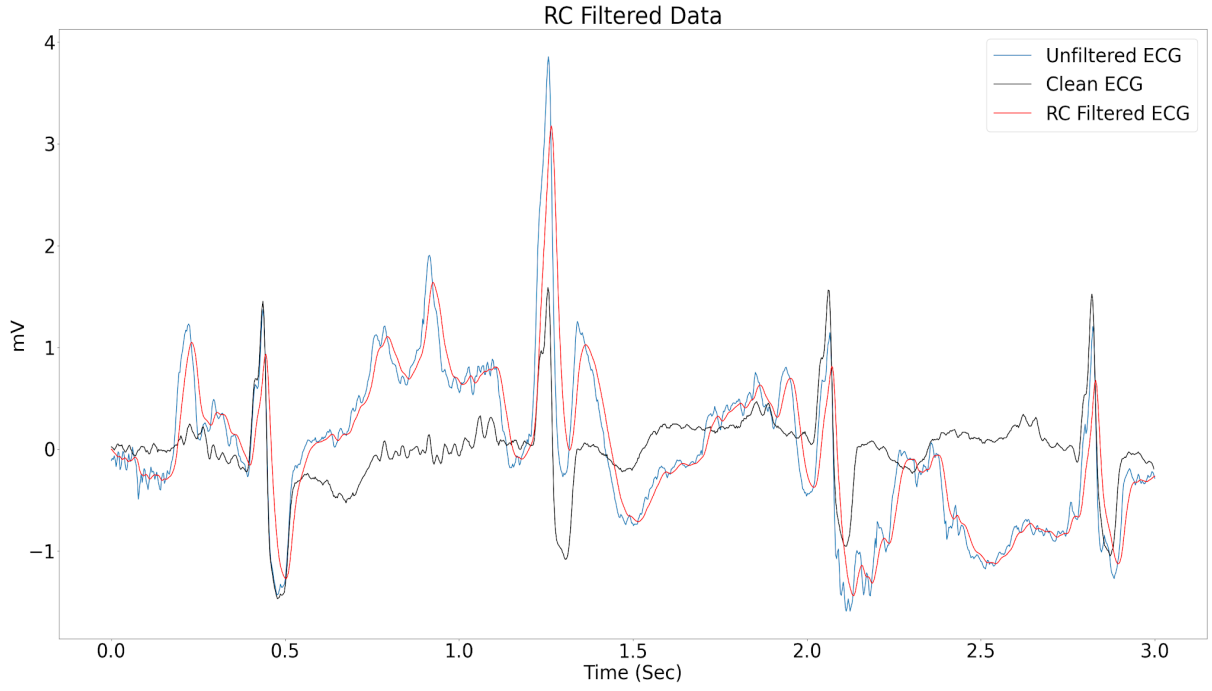
Over a 30 second period, the Butterworth filter model yielded an SNR difference of 16.20 dB. The first three seconds of this data can be seen below in Figure 5.

Figure 5 : Butterworth Filtered Data



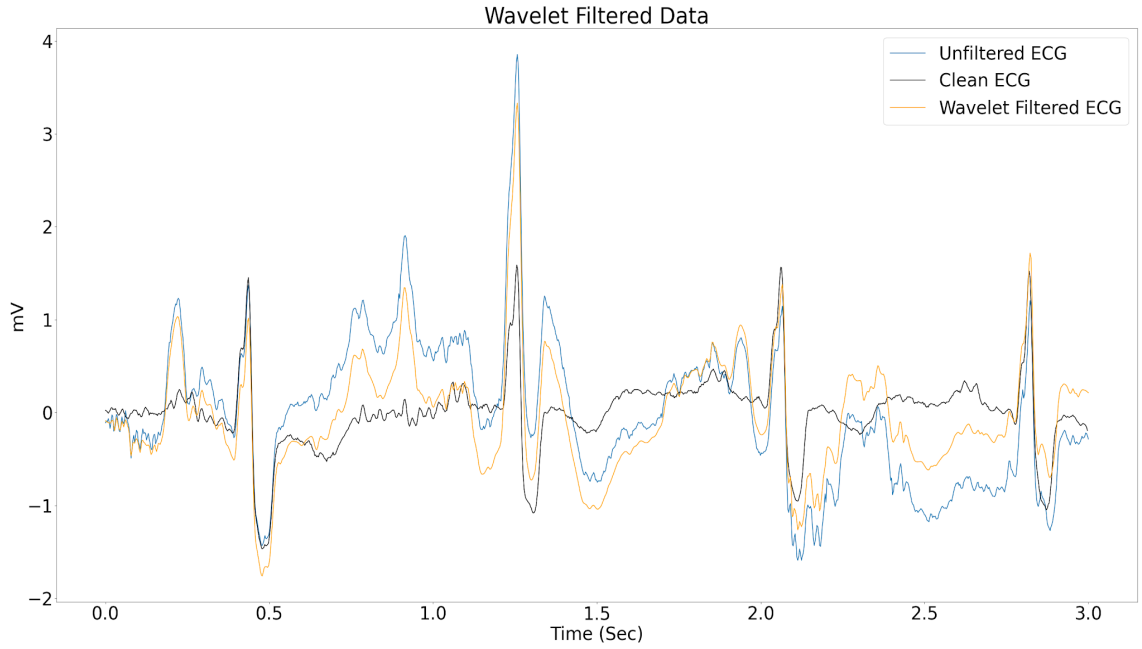
The next method, a 2nd Order RC Model, yielded a SNR difference of 14.27 dB, thus proving less effective than the Butterworth filter. The analog filtered data is shown below in Figure 6.

Figure 6 : RC Filtered Data



The Discrete Wavelet Transform Filter yielded an SNR difference of 12.84 dB. The first three seconds can be seen in Figure 7.

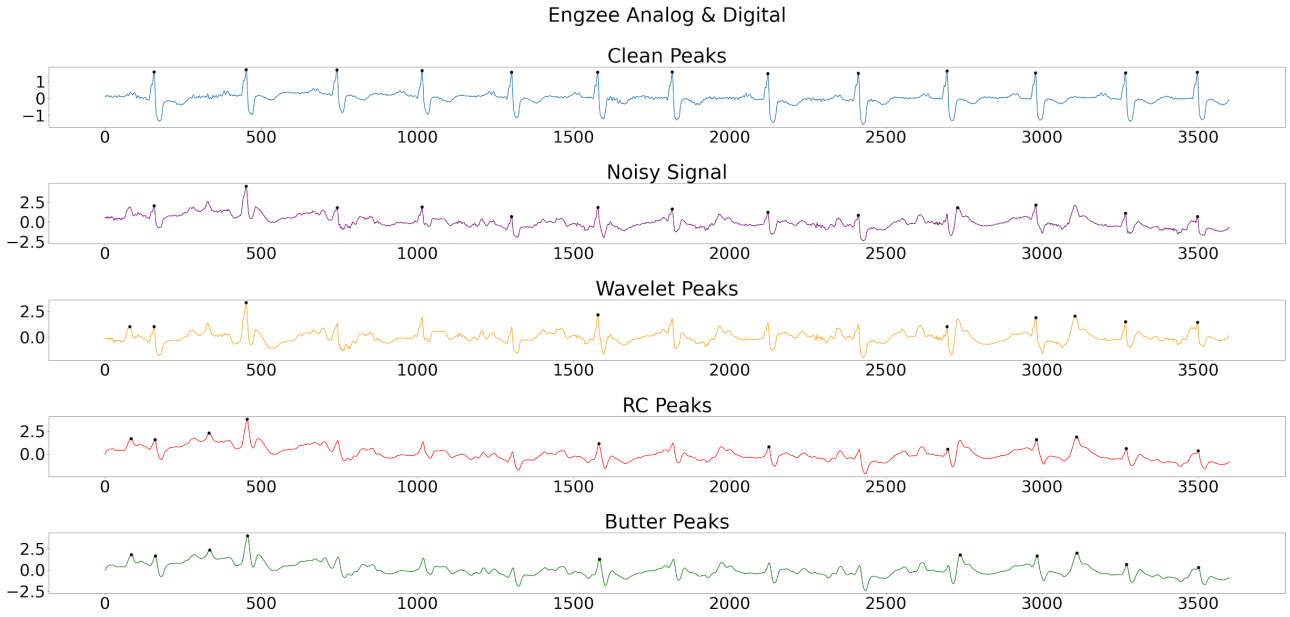
Figure 7 : Wavelet Filtered Data



Four main R-peak detection methods were then tested to find the algorithm generating the optimal combination of sensitivity and specificity, measured through counting missed and extra detected peaks. Each of the figures below shows the results of R-peak detection methods performed on the clean ECG signal, the noisy ECG signal, and the filtered signals over a 10 second period.

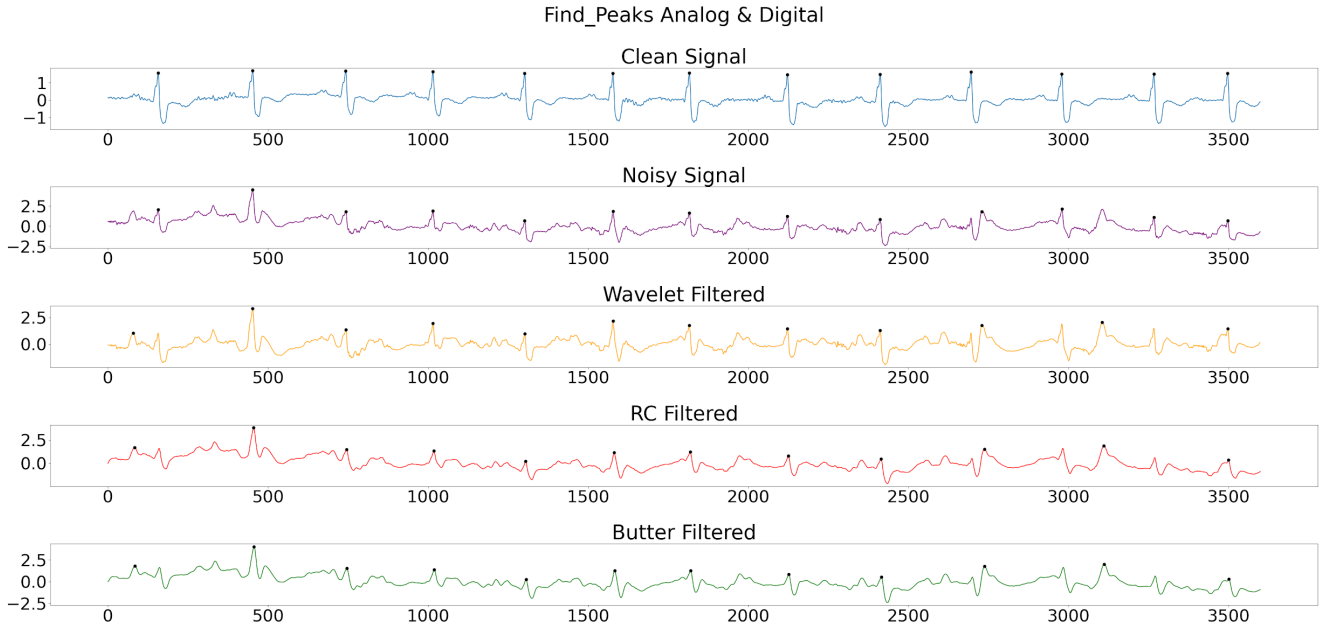
The first method tested was the Engzee detection algorithm. Results obtained using this method can be seen in figure 8.

Figure 8 : R-Peak Detection with Engzee Algorithm



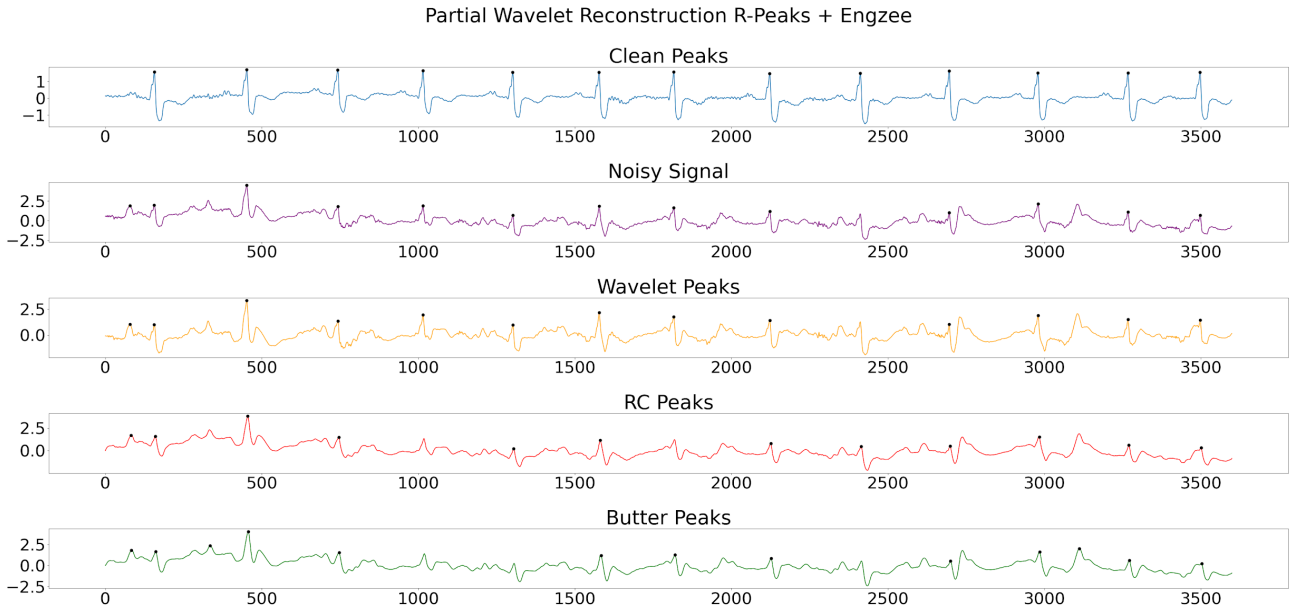
The second R-peak detection method that was tested was the find_peaks function within Scipy's signal processing library in Python. Results of this method can be seen in figure 9.

Figure 9 : R-peak Detection with Find_Peaks Function



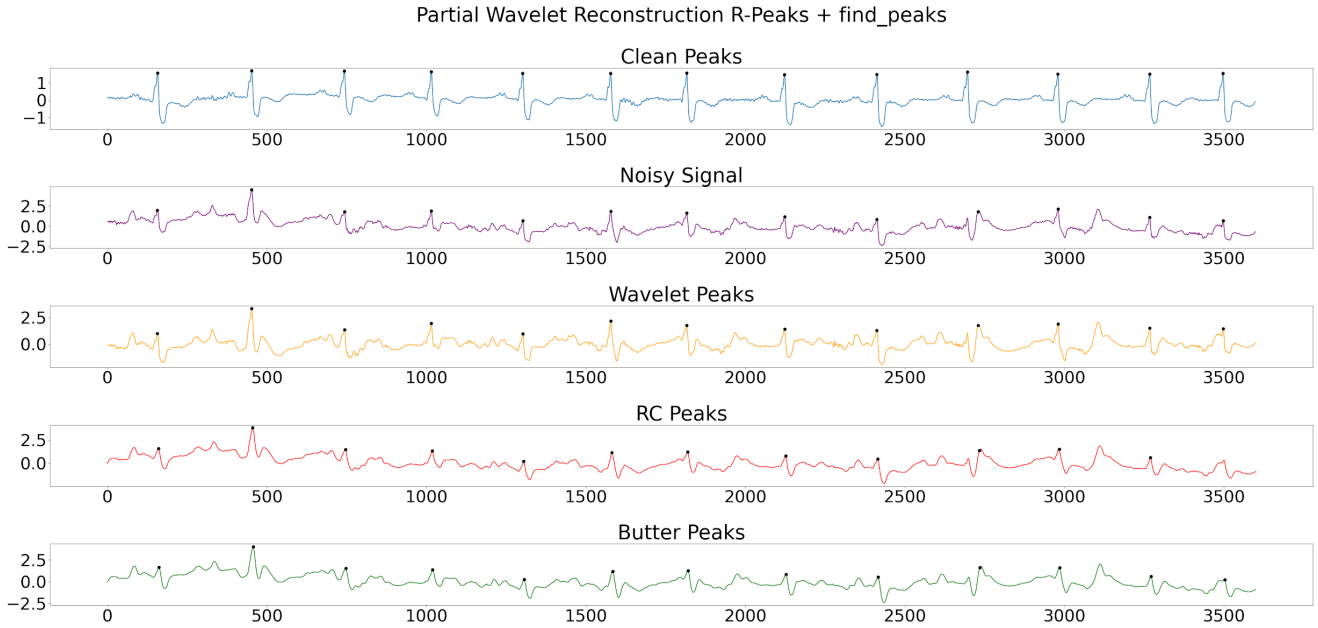
The next two methods performed a discrete wavelet transform on the filtered ECG data and partially reconstructed the signal to isolate the frequencies containing the R-peaks. This made the peaks more prominent, ultimately allowing for more accurate R-peak detection. The model below displays data after wavelet isolation used alongside the Engzee algorithm. Universally, there are more correct peaks and fewer extra peaks detected when compared to the use of the Engzee filter alone.

Figure 10 : R-peak Detection using Engzee Algorithm (with Partial Wavelet)



The final model shown below displays our data with the “find_peaks” method after performing wavelet isolation. This method also exhibited both improved R-peak detection sensitivity and specificity when compared to the find_peaks function on its own.

Figure 11 : R-peak Detection using Find_Peaks (with Partial Wavelet)



Below is a matrix of the various R-peak detection methods discussed combined with the three types of filters. Each possible combination of the filters was tested with R-peak detectors in order to determine the optimal combination, quantified by the metrics of specificity and sensitivity. Results show that using a lowpass Butterworth filter with a combination of a partial wavelet transform and the find_peaks Python function yields the best R-peak detection results, as well as providing the highest SNR.

Figure 12 : Results Matrix

	Butterworth	RC Filter	Wavelet Filter
Find Peaks	0.714 0.994	0.714 0.994	0.714 0.994
Engzee Detector	0.625 0.992	0.667 0.994	0.625 0.996
Find Peaks w/ Wavelet	1 1	0.91 1	0.91 0.998
Engzee Detector w/ Wavelet	0.769 0.994	0.83 0.998	0.91 0.998
SNR Difference	16.20	14.27	12.84

Analysis, Discussion, and Future Work

Overall, the goals for the mathematical modeling phase were to find the best R-peak detection and filtering methods to maximize specificity. The results of the model have indicated that the optimal combination is a peak detection algorithm of a partial wavelet transform & find_peaks Scipy function combined with a lowpass Butterworth filter.

The expected functionality of the device includes the ability to filter an ECG signal in real time and accurately detect the R-peaks of the signal through a combination of the optimal filter and R-peak detection methods modeled. The device software should detect R-peaks, RR intervals, and subsequently calculate various HRV metrics.

Going forward, the prototype of the device will use a combination of Arduino, circuits, and pre-gelled ECG electrodes. A 3-lead system will be utilized with two electrodes on the chest and one on the hip. The electrodes would be connected to our device, which is housed within a portable pouch that can be worn around the waist. An arduino will send data to the computer via bluetooth. The modeled peak detection algorithm of a partial wavelet transform & find_peaks Scipy function combined with a lowpass Butterworth filter will be implemented in order to process the ECG data. The algorithm and filter will be applied through a sliding window, wherein data is collected and analyzed over small increments of time. The modeled peak detection algorithm and filtering mechanism will be tested with real-time ECG readings from our phase III prototype. The device software will then use this data to analyze the LF/HF ratio of HRV in order to quantify stress. Also, by analyzing specific segments and qualities of an ECG signal, MI metrics and patterns can be determined to alert patients undergoing a myocardial infarction. Within the ECG signal, major signals of myocardial infarction to detect include ST segment elevation, ST segment depression, and T wave inversion. There are two possible models to analyze these patterns using machine learning. The first model will feed images of the electrocardiogram into a convolutional neural network, and the second model will analyze the data using a recurrent neural network. Both models will be tested in phase III through Python.

Since the users of this portable ECG device will be moving, a better understanding is needed regarding how to remove electrode motion artifacts caused by skin stretching from the data. Electrode motion artifacts can produce signals similar to R-peaks making it a crucial component to mitigate for accurate data and results. Furthermore, additional research will be needed to reduce the possible negative effects of hair and sweat on the skin to ECG signal data. These external factors can lead to artifacts on the signal data which the current code may not be equipped to handle. An additional question requiring focus during phase III is the possibility of incorporating a high-pass butterworth filter within the device to better remove baseline wander

from signals. The filtering code can always be improved on and will be adapted based on further testing during Phase III.

Works Cited

2nd order CR filter Design tools. (n.d.). Retrieved October 11, 2020, from http://sim.okawa-denshi.jp/en/CRCRkeisan.htm

Goldberger, A., Amaral, L., Glass, L., Hausdorff, J., Ivanov, P. C., Mark, R., ... & Stanley, H. E. (2000). PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. Circulation [Online]. 101 (23), pp. E215–e220.

Järvelin-Pasanen, S., Sinikallio, S., & Tarvainen, M. P. (2018). Heart rate variability and occupational stress-systematic review. *Industrial health*, *56*(6), 500–511. https://doi.org/10.2486/indhealth.2017-0190\

Kumar, P., & Sharma, V. K. (2020). Detection and classification of ECG noises using decomposition on mixed codebook for quality analysis. Healthcare technology letters, 7(1), 18–24. https://doi.org/10.1049/htl.2019.0096

Lenis, G., et al. (2017). "Comparison of Baseline Wander Removal Techniques considering the Preservation of ST Changes in the Ischemic ECG: A Simulation Study." Computational and mathematical methods in medicine 2017: 9295029-9295029.

Li, K., Rüdiger, H., & Ziemssen, T. (2019). Spectral Analysis of Heart Rate Variability: Time Window Matters. Frontiers in neurology, 10, 545. https://doi.org/10.3389/fneur.2019.00545

Lin, H. Y., et al. (2014). "Discrete-wavelet-transform-based noise removal and feature extraction for ECG signals." IRBM 35(6): 351-361. https://doi.org/10.1016/j.irbm.2014.10.004

Moody GB, Muldrow WE, Mark RG(1984). “A noise stress test for arrhythmia detectors.” Computers in Cardiology 11:381-384.

Nimunkar, A. J., & Tompkins, W. J. (2007). R-peak detection and signal averaging for simulated stress ECG using EMD. Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual International Conference, 2007, 1261–1264. https://ieeexplore.ieee.org/document/4352526/

Pan, J., & Tompkins, W. J. (1985). A Real-Time QRS Detection Algorithm. Retrieved 2020, from https://www.robots.ox.ac.uk/~gari/teaching/cdt/A3/readings/ECG/Pan+Tompkins.pdf

Peltola, M. A. (2012). Role of editing of R–R intervals in the analysis of heart rate variability. Frontiers in Physiology, 3. doi:10.3389/fphys.2012.00148

Rabbani, H., Mahjoob, M. P., Farahabadi, E., & Farahabadi, A. (2011). R peak detection in electrocardiogram signal based on an optimal combination of wavelet transform, hilbert transform, and adaptive thresholding. *Journal of medical signals and sensors*, *1*(2), 91–98.

Serhani, M. A., et al. (2020). "ECG Monitoring Systems: Review, Architecture, Processes, and Key Challenges." Sensors (Basel, Switzerland) 20(6): 1796.

Vinzio Maggio, A. C., et al. (2012). Quantification of Ventricular Repolarization Dispersion Using Digital Processing of the Surface ECG, InTech.

Ze Wang, Chi Man Wong, & Feng Wan (2017). Adaptive Fourier decomposition based R-peak detection for noisy ECG Signals. *Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Annual International Conference*, *2017*, 3501–3504. https://doi.org/10.1109/EMBC.2017.8037611