

Master Thesis

Analysis of the Noise Characteristics of Ambulatory Electrocardiogram

Félix Chávez José Andrés

Human Support Systems Laboratory
Department of Mechanical Engineering
Graduate School of Engineering
Mie University

March, 2022

Acknowledgements

I would like to express my deepest gratitude to Professor Ryojun Ikeura for his valuable guidance and advice during the development of this research. His constant support, kindness, encouragement, and patience have been very much appreciated.

I would also like to extend this gratitude to Associate Professor Soichiro Hayakawa and Assistant Professor Shigeyoshi Tsutsumi for their guidance. Their questions and opinions helped me think about the direction I should take with my research. I would also like to extend my thanks to Professor Ken'ichi Yano for reviewing my thesis.

My sincere thanks are extended to the Honda Benjiro Scholarship Foundation and the Mie Prefectural Citizen's Mutual Aid Association for their financial support during my studies in Japan. Thanks to them I could stop worrying about financial matters.

I wish to thank Dr. Ziti Apandi for providing the data I needed to proceed with my research and for her words of encouragement. I would also like to thank Ms. Akiko Fujiwara for her kindness, support, and patience during my Master studies. My deep gratitude is extended to the Center for International Education and Research (CIER) of Mie University for their invaluable support during my studies.

Finally, I would like to thank my parents, Andrés Félix Saguchi and Evelia Chávez Reyes, my siblings, and my friends for believing in me, even when I doubted myself. It has been a long journey that couldn't have begun nor finished without any of you. Despite the hard times we have lived through, thank you for lending an ear and sharing a laugh.

Abstract

Cardiovascular diseases are one of the main causes of mortality around the world. Many of these events happen outside-of-hospital conditions when the person is at home or at a public space without being monitored. Advances in technology have improved the patients' monitoring, be it a Holter monitor or a wearable device. However, these devices are still sensitive to noise from movement, therefore it is important to understand how the movement affects electrocardiographic recordings.

In this research three devices were tested. The BIOPAC system is a non-mobile device that represented conventional electrocardiography. The Hitoe system and the Vitalgram system are wearable solutions for ECG monitoring.

Since the goal is to understand how noise from motion affects the ECG signal, based on previous research the motion noise was determined to be the sum of baseline wander (BW), electromyogram (EMG), and electrode motion artifacts (EMA). BW and EMG were obtained by using a low-pass filter and a high-pass filter, respectively. The EMA were obtained by doing a beat-by-beat analysis. To analyze the signal beat-by-beat, the R-peaks of the ECG signal had to be identified. To do this, the discrete wavelet transform (DWT) was employed.

Once all the types were extracted, the signal-to-noise ratio (SNR) was used as a parameter to determine the quality of the ECG recordings from each device. The subjects were shown a video and were asked to perform certain actions in resting and stress conditions. The SNR values were more negative when the subject performed physical activities.

The results show that the ECG recordings from these devices were not affected by EMG noise, since the SNR values for EMG were always above 30 dB. As for the BW noise, the Vitalgram device was not affected, but the Hitoe and BIOPAC systems were mildly affected by it. The Vitalgram system was the only device to present a positive average SNR value for EMA while the subjects were running.

In Apandi's research the performance of several heartbeat detection algorithms was tested, including Apandi's proposed method. Using Apandi's research data and the SNR values from this research, the performance of these algorithms on recordings from these devices was estimated. The results show that using Apandi's proposed algorithm on recordings from the Vitalgram device would render the best results, since the estimated heartbeat detection performance was of 100%.

Table of Contents

Chapter 1: Introduction	6
1.1. Research Motivation	6
1.2. Research Objectives.	7
Chapter 2: Literature Review	8
2.1. The Electrocardiogram	8
2.2. The Discrete Wavelet Transform	9
2.3. Signal-to-Noise Ratio	11
2.4. Types of Noise caused by Movement	12
2.5. Use of the DWT for QRS complex detection	15
2.6. Performance of Heartbeat Detection Methods in Simulated Noise Conditions	17
Chapter 3: Data Collection	18
3.1. ECG Evaluation	18
3.2. Devices	18
3.2.1. BIOPAC System	18
3.2.2. Hitoe System.	19
3.2.3. Vitalgram 2 System	20
Chapter 4: Data Analysis	22
4.1. QRS Complex Detection	22
4.2. Baseline Wandering Noise Obtention	24
4.3. EMG Noise Obtention	25
4.4. Electrode Motion Noise Obtention	27

4.5.	Signal-to-Noise Ratio Calculation	30
Chapter 5: Results		32
5.1.	ECG Signal quality of the tested devices	32
5.2.	Device Performance According to Apandi et al.	34
Chapter 6: Conclusions and Future		38
6.1.	Conclusions	38
6.2.	Future Work	39
Appendix A		40
A.1	“changeEcgFreq” function	40
A.2	“waveletPeakDetection” function	40
A.3	Script used to analyze ECG recordings (Vitalgram, Subject 004)	41
References		50

Chapter 1: Introduction

1.1 Research Motivation

Cardiovascular diseases (CVDs) are one of the main causes of mortality worldwide. The World Health Organization estimates that roughly one third of all deaths globally are caused by CVDs [1].

Moreover, although there are many types of cardiovascular diseases, according to [2], arrhythmias and sudden cardiac death represent 15% to 20% of all deaths worldwide. In other words, about half of CVDs.

Monitoring the electrocardiogram (ECG) of a person can help diagnose and treat the patient faster, narrowing the possibilities of a sudden death. However, this would require the use of a device that can monitor a person at any given time while the person performs its daily activities. This system should also be able to recognize abnormalities in the heartbeats and be able to notify the person in case there is risk of heart failure.

Nowadays, good quality ECG measurements can be done at hospitals and research institutions using devices such as the Holter [3]. There are also other commercial solutions such as wearable devices that can monitor the heart's activity. However, these solutions, even though they can record good quality ECG while the person is resting, when the person is moving the quality of the ECG signal is reduced by the noise caused by the movement. Also, these systems are recording tools, not diagnostic tools. This project is an initial step towards a greater goal: to build a system that is able to classify heartbeats when a person is moving.

1.2 Research Objectives

In this study, mobile and stationary devices are tested while the person is resting and performing physical activities. The noise present in the ECG signal is then analyzed and the result of their performance is compared.

The first objective of this research is to separate different types of noise. The noise from movement can be classified as baseline wander noise, electromyogram noise and electrode motion artifacts. The goal is to be able to extract these noises separately.

The second objective of this research is to determine which device is more suitable for collecting samples while a person is moving. The point of comparison is the Signal-to-Noise ratio (SNR). The device with better SNR values shall be selected to proceed with further research if the quality of the signal recorded while the person is moving is good.

The third objective is to estimate how would each device perform according to the heartbeat detection method proposed by Apandi et al. By taking the SNR values of each device while the subjects are involved in a physical activity, and using the data from Apandi's research, the heartbeat detection rate for each device and activity can be estimated.

Chapter 2 Literature Review

2.1 The Electrocardiogram

The electrocardiogram, also known as ECG or EKG, is a recording of the electrical activity of the heart. When the cardiac impulse passes through the heart, some of the current propagates to the tissues around the heart and some of it reaches the skin. When electrodes are placed on the skin on opposite sides of the heart, these electrical potentials can be recorded [4]. As shown in Figure 1, the typical ECG waveform has different waves that are generated during the cardiac cycle.

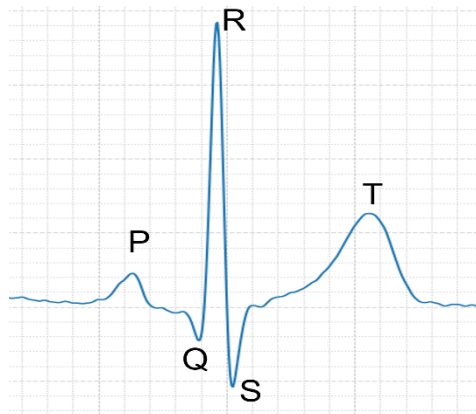


Figure 1. Typical ECG Waveform.

The P wave is generated when the atria depolarize just before they contract and pump blood into the ventricles. The QRS complex is generated when the ventricles depolarize and contract to pump blood outside of the heart. The T wave is generated when ventricles repolarize after contraction [4]. This cycle of depolarizations and repolarization is shown in Figure 2.

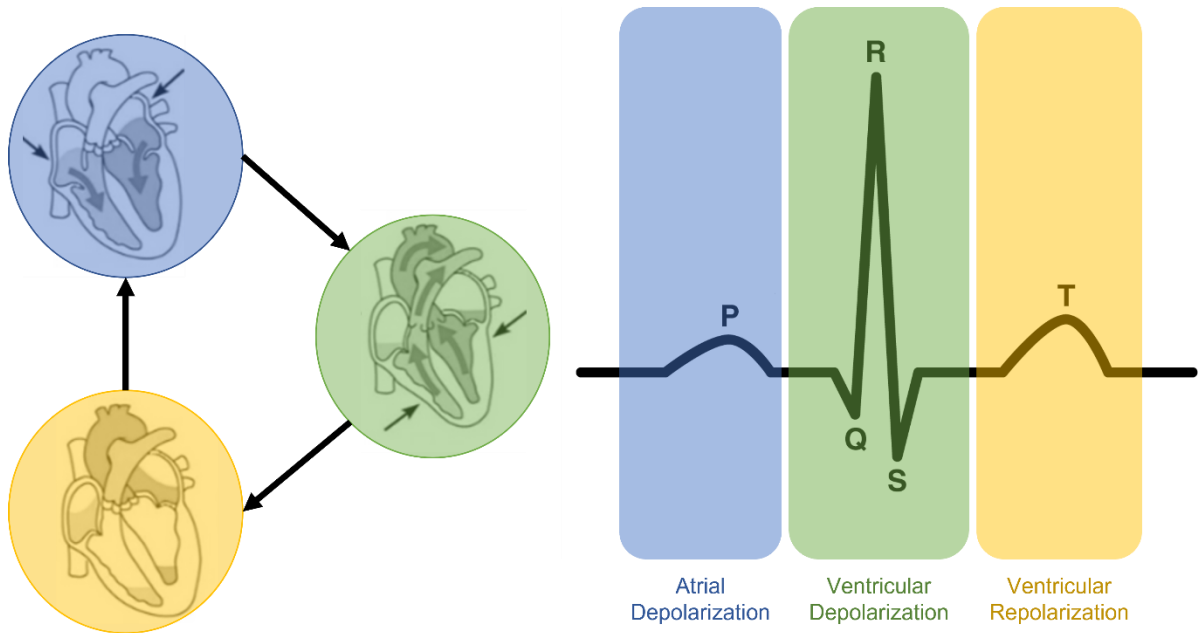


Figure 2. Cycle of the repolarization and depolarization of the atria and ventricles of the heart and its correspondence with the ECG waves.

2.2 The Discrete Wavelet Transform

A wavelet is a waveform that has an average value of zero, but it has a limited time duration, unlike sinusoidal waveforms that extend to infinity [5]. There are many types of wavelets, and a determined wavelet may be used depending on the task at hand. In Figure 3 few types of wavelets are shown.

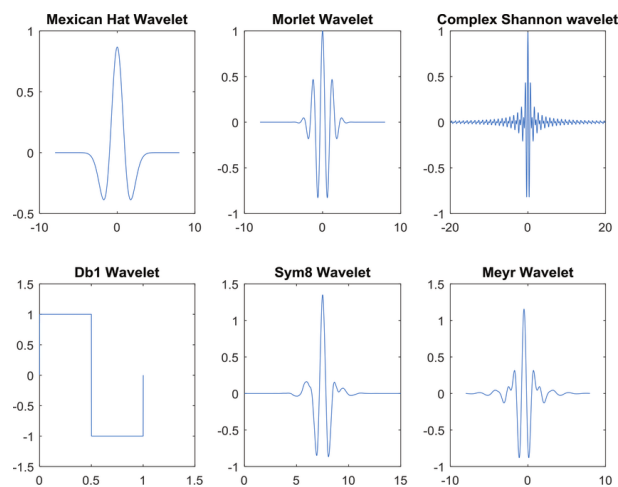


Figure 3. Examples of wavelets.

In signal processing, mathematical transformations are applied to signals in order to obtain information that is not available within the raw signal. Most of the signals are in the time domain and when these signals are plotted, they are plotted in a time-amplitude representation [6].

The Fourier transform (FT) breaks down a signal into different sine waves of different frequencies. It is a mathematical transformation that provides a frequency-amplitude representation of the signal. However, this transformation doesn't provide information of when these frequencies happen in time.

This is not a problem when the signal is stationary, which means that the frequency components of the signal are constant throughout the signal. However, an ECG signal's frequency, for example, changes in time, thus it is non-stationary. The FT can be used to see if a determined frequency exists in the signal, but not when it happens.

When a time-frequency representation is needed, the Wavelet Transform (WT) can be employed. Sometimes, a particular feature of interest happens abruptly in a signal. So, in a case like this it can be of interest to know when it occurs. An example of this are the R peaks of an ECG signal.

But how does the discrete wavelet transform (DWT) work? By passing the time domain signal through a series of high-pass and low-pass filters, filtering out either low or high frequency portions of the signal [6]. By repeating this process, a portion of the signal is removed on each iteration. This process is called decomposition and Figure 4 shows a diagram that represents it.

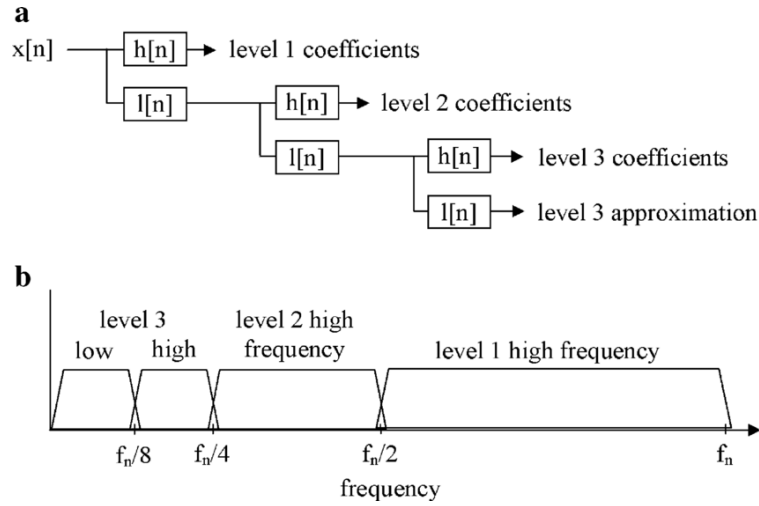


Figure 4. Diagram showing a three-level decomposition process. (a) Schematic of the filter bank. (b) Results of the filtering in the frequency domain. *K. Amolins, Y. Zhang and P. Dare, "Wavelet based image fusion techniques — An introduction, review and comparison", ISPRS Journal of Photogrammetry and Remote Sensing, vol. 62, no. 4, pp. 249-263, 2007. Available: 10.1016/j.isprsjprs.2007.05.009.*

The result is a group of signals categorized by frequency bands. This shows which frequencies within the band exist at a certain time interval.

2.3 Signal-to-Noise Ratio

The signal-to-noise ratio (SNR) is a measure of how much of a desired signal there is against the level of undesired signals [8]. In other words, it is the ratio of the signal of interest against the noise. The SNR, in decibels, can be expressed by the following equation [9-10]:

$$SNR_{dB} = 10 \log_{10} \left(\frac{P_{signal}}{P_{noise}} \right)$$

Where:

P_{signal} = Power of the desired signal

P_{noise} = Power of the undesired signal(s)

If the SNR value is zero, it means that the magnitude of the signal and the magnitude of the noise are the same. If the SNR value is negative, it means that the power of the noise is superior to that of the signal. On the other hand, if the SNR value is positive, it means that the desired signal has more power than the noise.

In this study, both the signal and the noise are voltages measured across the same impedance. The SNR is calculated using the averages of the powers of the noise and the signal, in other words, it is calculated using the root mean square (RMS) voltage squared.

2.4 Types of Noise caused by Movement

There are multiple factors that can contaminate an ECG signal. These unwanted signals that are mixed with ECG signal are considered noise, since they distort the shape of the pure ECG waveform. The main sources for noise in the ECG signal are [11]:

- Power line interference
- Baseline wandering
- Electromyogram (EMG)
- Motion artifacts

The power line interference is a common noise that can be found not only in ECG but also other biometric signals. Depending on the region this noise has is sinusoidal wave of 50 Hz or 60 Hz [12]. However, this noise is not caused by the movement of a person. This type of noise, if present, can be reduced by using a notch type filter for the exact frequency of the power line in the country or region where the recording is taking place.

The baseline wander is a noise that is commonly caused by the respiration of the patient and movement of the instrumentation [13]. It is a low frequency noise that typically has a frequency below 1 Hz. A sample of ECG with baseline wander can be seen in Figure 5.

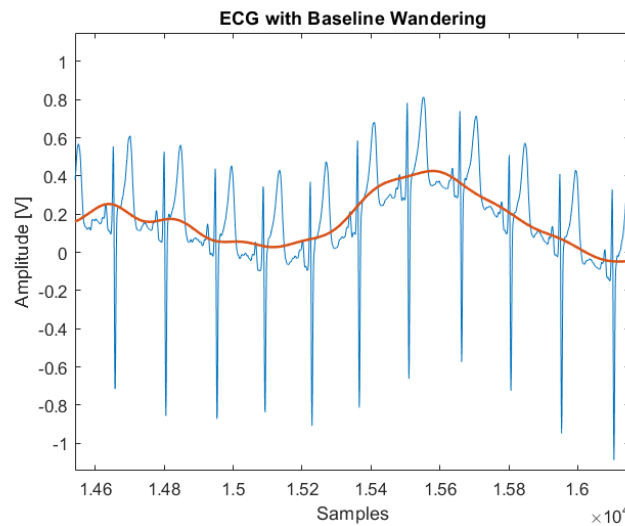


Figure 5. ECG signal contaminated with baseline wandering. The blue line represents the contaminated ECG signal, and the red line represents the baseline wander.

The electromyogram (EMG) is a signal generated when the muscles contract. However, it is considered noise when the signal of interest is the ECG. The EMG is high frequency noise [13]. The power spectrum of the EMG ranges from 0 Hz to 500 Hz [14] and most of the power can be found in the frequency band of 50 to 250 Hz as shown in Figure 6.

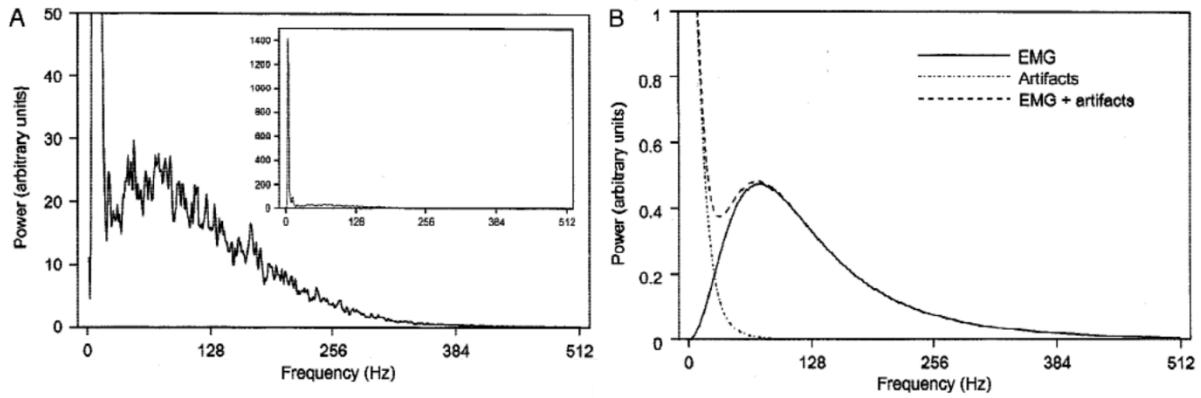


Figure 6. (A) Power Spectrum of orbicularis oculi EMG with low frequency contamination. (B) Diagram of the power spectrum of an EMG signal showing the low frequency noise, the EMG signal and the mixture of both. A. Bostel, "Optimal signal bandwidth for the recording of surface EMG activity of facial, jaw, oral, and neck muscles", *Psychophysiology*, vol. 38, no. 1, pp. 22-34, 2001.

Available: 10.1111/1469-8986.3810022.

The last type of noise produced by movement are electrode motion artifacts. These artifacts are caused by the shaking or movement of the patient [15]. The movement can be random or rhythmical and its causes can be several, such as tremors caused by a disease or a patient moving the limbs during the ECG recording. When the person moves, the cables and/or electrodes move too, generating noise and contaminating the ECG recording. Figure 7 shows an ECG signal contaminated with electrode motion artifacts.

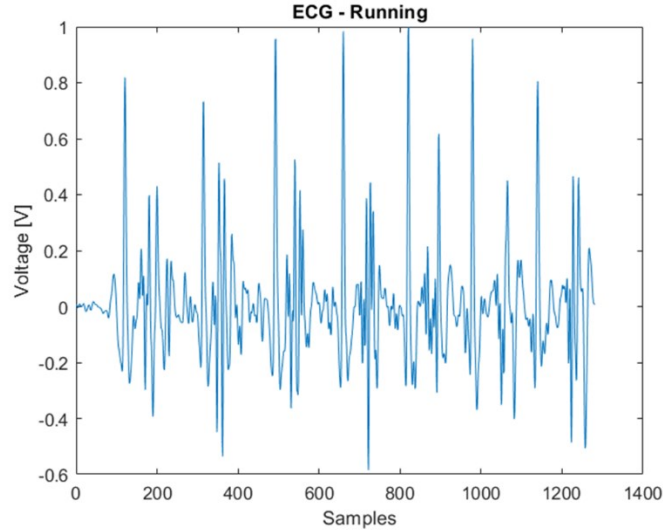


Figure 7. ECG sample recorded when the subject was running. It is contaminated by electrode motion artifacts.

2.5 Use of the DWT for QRS complex detection

In the research paper by Haddadi et al. the application of DWT is proposed to detect the QRS complex, which is the most notorious feature of the ECG. In that study, the ECG samples from the MIT-BIH database were used.

Haddadi et al. decided to use the Daubechies 4 (Db4) wavelet to implement the DWT. There is no absolute way to determine which wavelet shall be used for each application. They found that using the Db4 wavelet gave details more accurately than other wavelets. The energy spectrum of this wavelet has similarities with the QRS complex.

In this study they did an 8-level decomposition and the d4 Coefficients showed the highest cross correlation with the original signal, so Haddadi et al. used this coefficient for the QRS complex detection. Figure 8 shows the ECG signal and the d4 coefficients. According to [16], an average QRS complex detection rate of 98.1% was achieved.

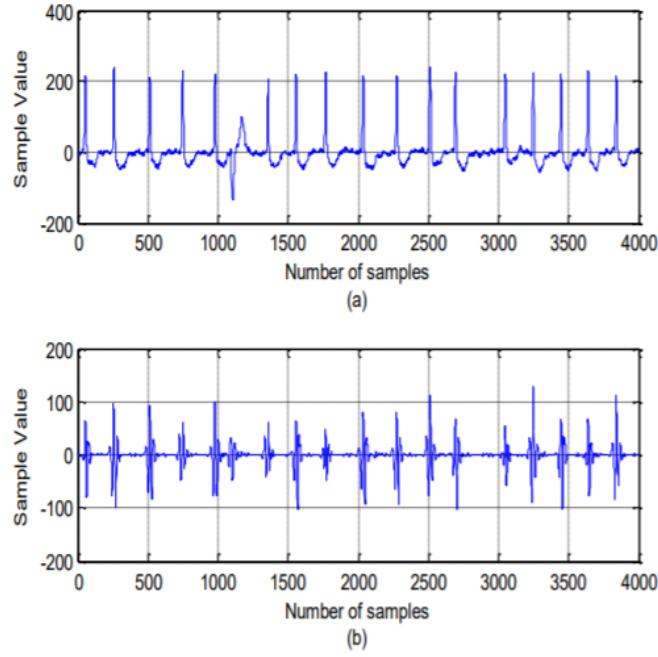


Figure 8. (a) ECG signal with filtered baseline wander. (b) The filtered ECG signal reconstructed using the d4 coefficients. Image taken from: R. Haddadi, E. Abdelmounim, M. El Hanne and A. Belaguid, "Discrete Wavelet Transform Based Algorithm for Recognition of QRS Complexes", World of Computer Science and Information Technology Journal, vol. 4, no. 9, pp. 127-132, 2014. [Accessed 16 January 2022].

In the research paper by Kumar et al. the DWT method was also used to detect the QRS complexes. However, in that study the Symlet 4 (sym4) wavelet was used instead of the Db4 wavelet. Symlet wavelets are versions of the Daubechies wavelets but with increased symmetry [17].

Similarly, samples from the MIT-BIH database were used. And since the sym4 wavelet has similar characteristics to the ECG waveform, it was selected to perform the multi-level decomposition using the DWT. According to [17], the R peak detection rate was of 100%.

2.6 Performance of Heartbeat Detection Methods in Simulated Noise Conditions

Apandi et al. used the ECG recordings from the MIT-BIH database. This database has 48 records and the samples were recorded in resting conditions. To simulate ECG samples in ambulatory conditions, different types of noise were recorded when a person was moving. These noise samples were also obtained from the MIT-BIH database. The types of noise are baseline wander (BW), EMG noise and electrode motion artifacts (EMA).

In her thesis, Apandi simulated noisy signals by embedding these noise samples with different intensities to the original ECG recordings. She used SNR values ranging from -12dB to 12dB. Then, she tested several heartbeat detection methods on the ECG recordings with simulated noise to test their performance. The Figures below show the performance of different methods for heartbeat detection, such as the Pan-Tompkins algorithm, WQRS, Hamilton and Apandi's proposed method for sample 100 of the MIT-BIH database.

Chapter 3: Data Collection

3.1 ECG Evaluation

The sample size for this study is 5 subjects. These subjects are male individuals whose age is between 20 and 37 years old. All of the individuals reported to have no problems regarding their health. The subjects were shown a video that contained instructions about what to do during the test.

They were asked to perform simple actions such as resting while lying on a bed, resting while sitting, resting while standing, and also performing physical activities such as walking and running.

The video not only serves as guidance for the subjects, but also serves as a timeline of events. Although some of the devices had integrated an accelerometer and gyroscope (which were quite useful when identifying the subject's behavior), the video too, was useful when identifying which actions the subject was performing at a given time.

3.2 Devices

In this study three devices were used to record samples. One of the devices is not mobile, and the other two are portable devices. The non-mobile device represents conventional electrocardiographic recording.

3.2.1 Biopac System

The BIOPAC system is not a mobile device. It is capable of recording ECG signals at high sampling rates. However, to keep the sampling rates similar throughout all of the devices, it was set to 200Hz.

The BIOPAC system uses cable electrodes with gel to sense the heart's electrical potentials on the skin. These electrodes connect to BIOPAC's ECG module to amplify the signal. The ECG module is connected to an Analog Output and Power Source module. A shielded cable connects the analog output module to a device called Aproc 1, which will digitalize the signal. Finally, the Aproc 1 device connects to the PC using a USB cable. The figure below shows a schematic of this system.

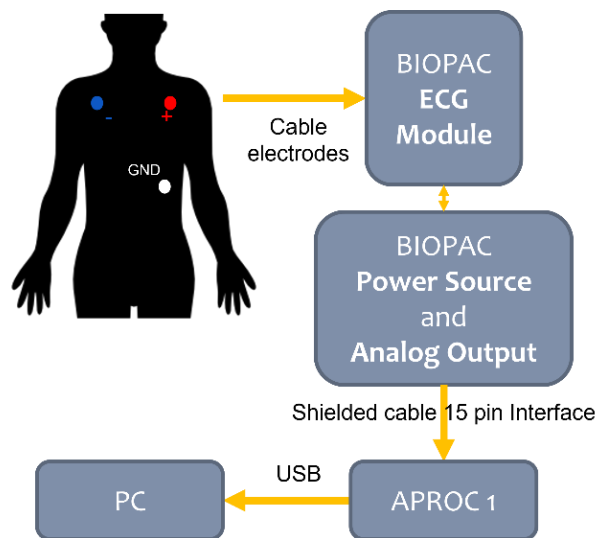


Figure 9. Schematic of the BIOPAC System.

3.2.2 Hitoe System

The Hitoe System consists of a transmitter device, a special T-shirt, an Android application, and a mobile device such as a smartphone with Android OS. The Hitoe transmitter was developed by Docomo and has a maximum sampling frequency of 200 Hz. The electrodes are on the T-shirt, and they are made of some fabric-like proprietary material.

The device also includes a 3-axis accelerometer that was used to record acceleration and identify the subjects' actions during the ECG evaluation. It connects

to an Android device using a low energy Bluetooth (BLE) connection. The figure below shows a schematic of this system.

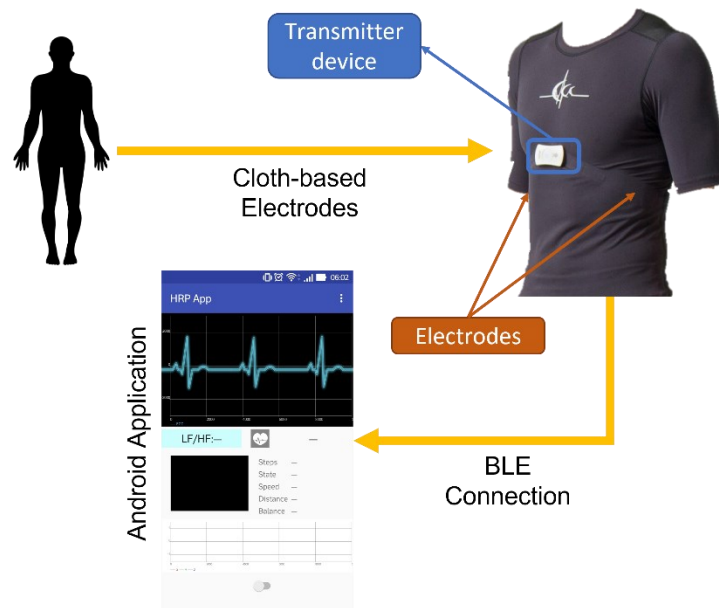


Figure 10. Schematic of the Hitoe System

3.2.3 Vitalgram 2 System

The Vitalgram System consists of the Vitalgram 2 device, patch-type electrodes, an iOS application, and an iOS device. The Vitalgram 2 device is made by AffordSENS Corporation, and it has sampling frequencies ranging from 128Hz to 1024Hz. However, to keep the measurements similar throughout all the devices, a sampling frequency of 256 Hz was selected. Like the Hitoe device, it is also capable of recording acceleration with a 3-axis accelerometer. It connects to an iOS device using a BLE connection.

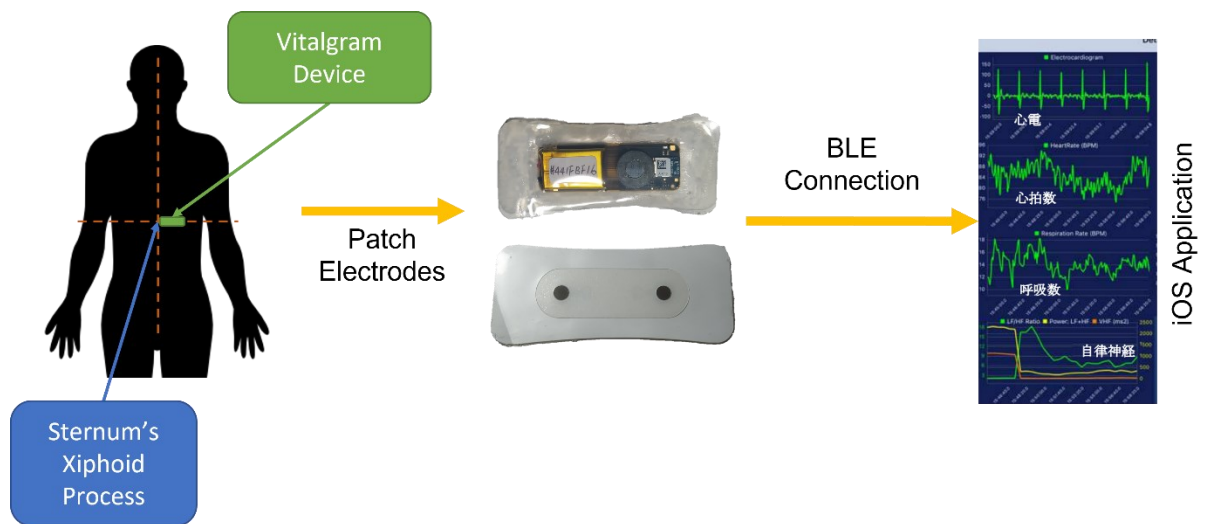


Figure 10. Schematic of the Vitalgram System

Chapter 4: Data Analysis

4.1 QRS Complex Detection

The DWT method was used to detect the QRS complexes. This method was selected since it has shown good results in detecting the R peaks and it has a lower impact when computing according to previous research [16-17].

The wavelet Symlet 4 was selected to perform the DWT since it has similarities with the ECG signal, and it had shown to detect more details of interest in previous research [17]. First, a five-level decomposition was performed. This decomposition can be seen in Figure 11.

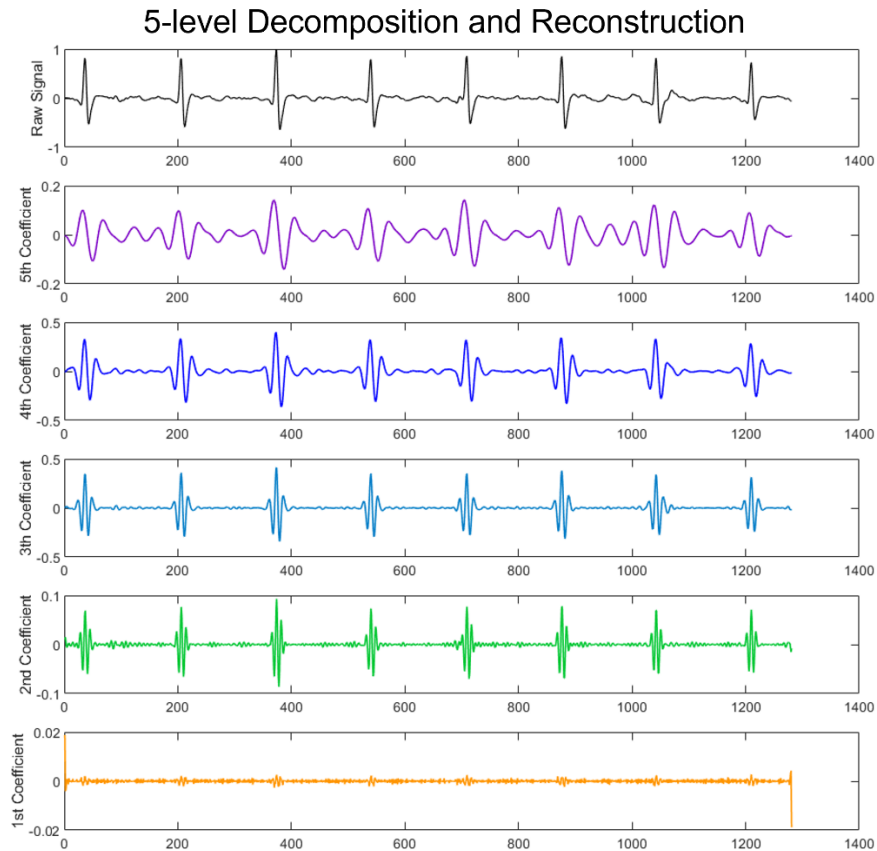


Figure 11. Five-level decomposition and reconstruction of an ECG signal using the coefficients for each level of decomposition.

After decomposing the signal into 5 levels, to emphasize the R peaks in the reconstruction process, the 3rd and 4th coefficients with a correlation average of 0.71 and 0.83 were selected, since they showed more correlation to the original signal compared to the other coefficients. In the table below the correlation of the reconstruction for each coefficient to the original signal is shown.

Coefficient	Correlation Average
1st	0.13327
2nd	0.361929
3rd	0.715412
4th	0.830637
5th	0.510984

Table 1. Correlation average of the original signal and the signal reconstruction for each coefficient.

After reconstructing the signal using the 3rd and 4th coefficients, a simple method for detecting peaks was employed using the “findpeaks” function using MATLAB [20]. The results of this method are shown in the figures below, for an ECG signal recorded while resting and an ECG signal recorded while walking, respectively. It can be seen that this method effectively identifies the location in time of the R peaks in the ECG signal.

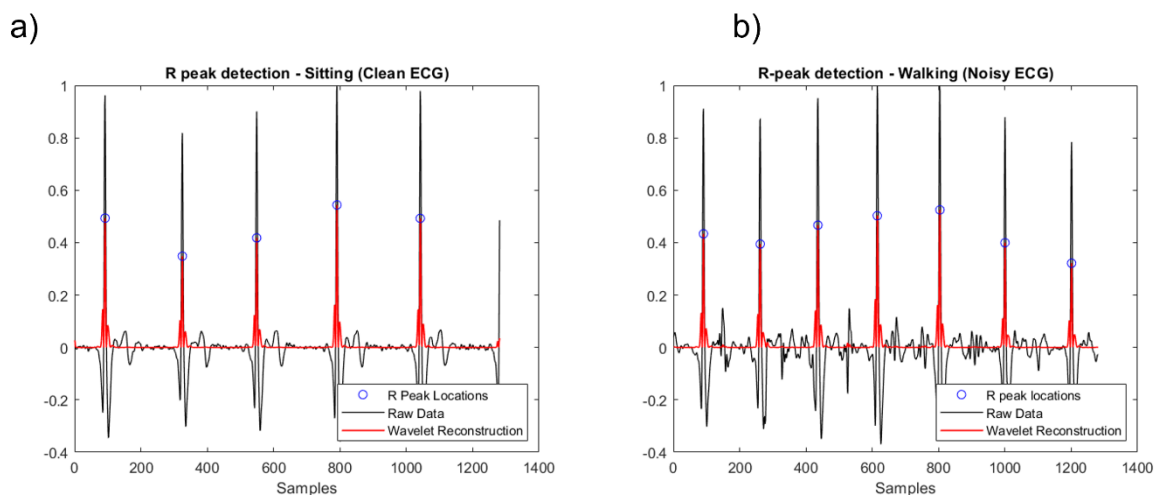


Figure 12. a) Locations of the R-peaks in a clean ECG signal. b) Locations of the R-peaks in a noisy ECG signal

4.2 Baseline Wandering Noise Obtention

The baseline wander is a common noise that comes from the motion of the subject. It is most commonly caused by the person's respiration motion although it can also be caused by a slow movement of the cables. Since its frequency is very low, typically below 1 Hz, a low pass filter with a cut frequency of 1 Hz was implemented.

The filter was designed using the MATLAB function “designfilt”. The type of filter is a Low-Pass IIR filter of order 12. The passing frequency is 1Hz and the passband ripple is of 0.001. The Magnitude response is shown in Figure 13, and an ECG signal with baseline wandering noise is shown in Figure 14.

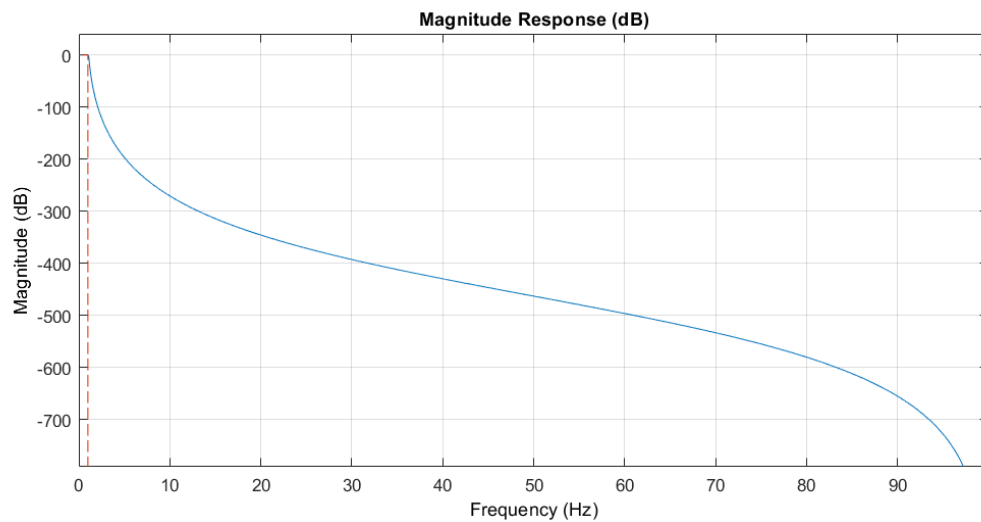


Figure 13. Magnitude response of the low-pass filter. The passband frequency is 1Hz.

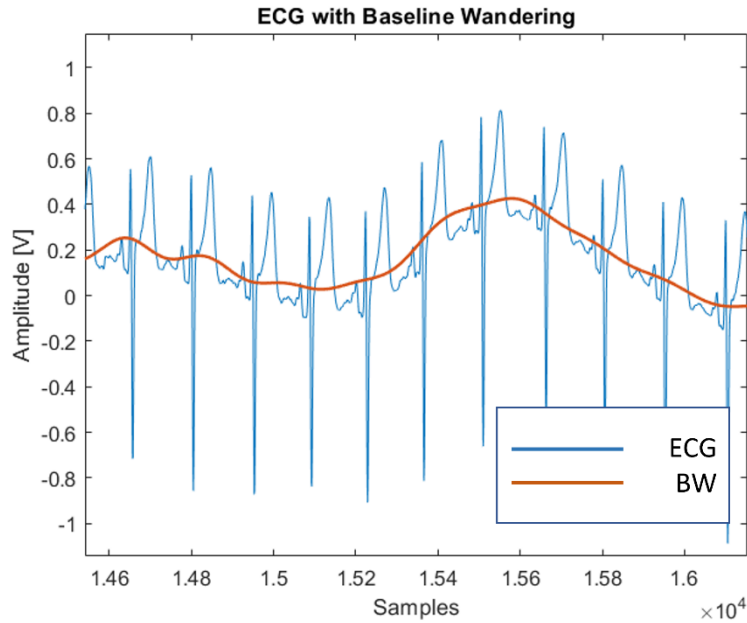


Figure 14. ECG with baseline wandering. The blue line is the ECG signal with baseline wandering. The red line is the result of using the lowpass filter.

4.3 EMG Noise Obtention

The EMG is a biological signal generated when there is muscle activity. Therefore, when a person moves during an ECG recording, it is sometimes possible to record EMG activity as well. However, when recording an ECG signal, the EMG signal is only getting in the way of what would be a clean ECG recording, so it is considered noise.

As shown before, the power spectrum of an EMG signal ranges from 0 Hz to 500Hz. However, most of the energy lies in the range of 50Hz to 200Hz. Since its frequency is high compared to the frequencies of an ECG signal, to obtain this noise a high-pass filter was used.

The filter was designed using the MATLAB function “designfilt”. The type of filter is a High-Pass IIR filter of order 12. The passing frequency is 50Hz and the

passband ripple is 0.01. Figure 15 shows the magnitude response of the filter, and Figure 16 shows an ECG signal with EMG noise.

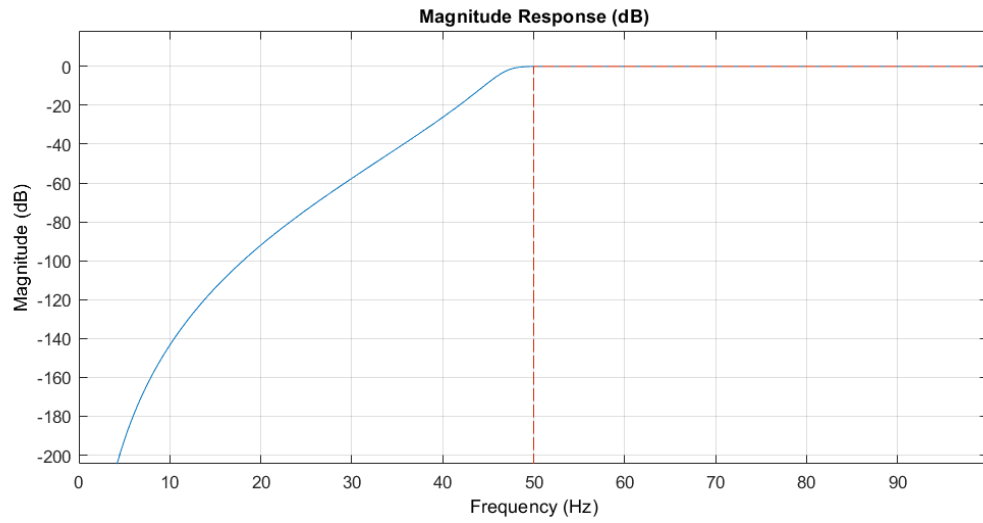


Figure 15. Magnitude response of the high-pass filter. The passband frequency is 50Hz.

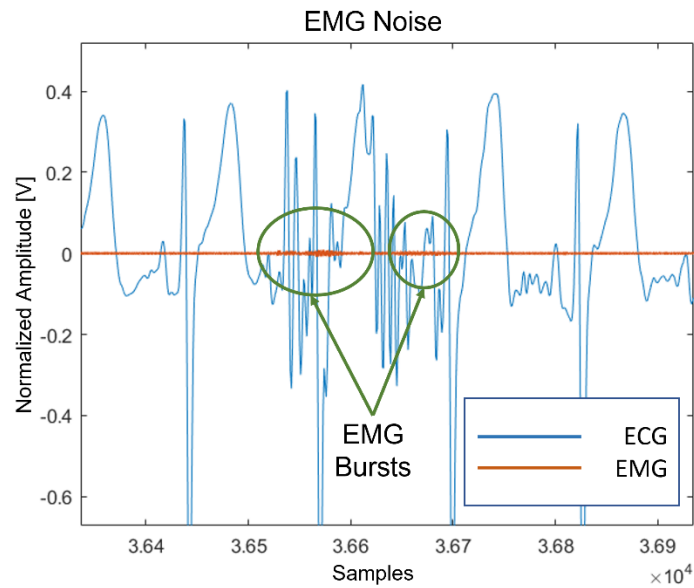


Figure 16. ECG signal with EMG noise. The blue line is the ECG signal with EMG noise and electrode motion artifacts. The red line represents the EMG signal. a few EMG bursts can barely be seen, since their amplitude is small.

4.4 Electrode Motion Noise Obtention

While the frequency bands where most of the energy of the BW and EMG noise lies are well known, electrode motion artifacts (EMA) are not as simple to isolate. EMA can vary widely. They can sometimes be rhythmical and sometimes sudden, and how the electrodes movement is going to generate a change in the voltage of the signal can be unpredictable at times.

To obtain the EMA, an indirect approach was done. In this study two assumptions were made. The first assumption is that when the EMG and BW are removed from a noisy signal, any leftover noise is considered EMA, as shown in the following equations:

$$Noise_{All} = BW + EMG + EMA$$

$$EMA = Noise_{All} - BW - EMG$$

The second assumption is that the cleanest signal that can be recorded is the one recorded when the subjects are resting. This signal, in the case of the Hitoe and Vitalgram devices, has no EMG, BW and EMA. In the case of the BIOPAC device, the BW had to be subtracted to obtain the cleanest signal possible since it had some BW noise, but not EMG and EMA.

Let's take the case when the subject is running. Since it is the most noise inducing activity, that ECG recording has all types of motion noise. To obtain the noise in the signal, the noisy ECG can be compared to the resting (clean) ECG as in:

$$Noise = Noisy ECG - Clean ECG$$

However, this cannot be done directly. The noisy ECG signal is a signal that is recorded when a person is moving, thus, it has a higher heart rate. The clean ECG signal is recorded when a person is resting, thus it has a lower heartrate. Since both

signals have a different frequency, they cannot be directly subtracted. If they were directly subtracted, there would be a mismatch between the R peaks of both signals and the resulting “noise” would not be accurate. To minimize the impact of the mismatch between R-peaks, a solution would be to analyze the signal heartbeat by heartbeat.

Before analyzing the signal beat by beat, the noisy signal must be pre-processed. Since the current goal is to obtain the EMA, and the EMG and BW noises had already been isolated, the EMG and BW noise are subtracted from the noisy signal. Any leftover noise in the noisy signal comes from EMA.

$$ECG_{EMA} = ECG_{Noisy} - BW - EMG$$

Then, using the DWT to detect the R-peaks on the resting ECG signal, a single “clean” heartbeat is selected from R-peak to R-peak using the locations obtained. The heartbeat would be delimited by:

$$Heartbeat_{clean} = ECG_{rest}[locs(n) : locs(n + 1)]$$

Next, the R-peaks of the noisy ECG signal should be identified using the DWT. Once the locations of the R-peaks are known, the noisy signal can be analyzed beat by beat. Let’s take the first beat of the noisy signal. The period of this noisy heartbeat is known, and the period of the clean heartbeat is known.

Using this information, the clean heartbeat can be rescaled to have the same duration as the noisy heartbeat using the resample function from MATLAB. Once the signal has been rescaled, the sampling frequency has changed. But to analyze the signal it must have the original sampling frequency, so it is resampled back to the original frequency. The following code shows the function developed to change the frequency of the clean heartbeat:

```

function [tResamp,ecgResamp] =
changeEcgFreq(tRest,ecgRest,ecgFreqRest,ecgFreqMove)
%changeEcgFreq: Changes the heart rate of an ecg sample.
%
%       tRest,ecgRest,ecgFreqRest,ecgFreqMove
%   Changes the time vector of the ECG sample by a factor determined by the
%   heartrate of the ECG sample and a target heartrate.
%   The new signal (time vector and ECG) has now a new sampling frequency.
%   Higher if the heart rate was increased. Lower if the heart rate was
%   decreased.
%   To match the original sampling frequency, the ECG signal is now
%   resampled.
%   The output arguments correspond to a time vector and ECG that have a
%   different frequency than the original, but have the original sampling
%   frequency.

tRestTemp = tRest*(ecgFreqRest/ecgFreqMove);
delayFactor = tRest(1) - tRestTemp(1);
tRest2 = tRestTemp + delayFactor;

samplingFreq1 = ceil((1/(tRest(2)-tRest(1))));
samplingFreq2 = ceil((1/(tRest2(2)-tRest2(1))));

ecgResamp = resample(ecgRest,samplingFreq1,samplingFreq2);
tResamp = [];
temp = 0;
for i = 1:size(ecgResamp)
    tResamp(i) = temp;
    temp = temp + (1/samplingFreq1);
end
tResamp = tResamp + tRest(1);
tResamp = tResamp.';
end

```

Once the clean heartbeat has been properly rescaled to the same length of the noisy heartbeat, and resampled to the same sampling frequency of the noisy heartbeat, the heartbeats can be subtracted to obtain the noise from EMA:

$$EMA_{One\ beat} = Heartbeat_{EMA} - Heartbeat_{Clean}$$

$$EMA_{N\ beats} = ECG_{EMA_{N\ beats}} - ECG_{CLEAN_{N\ beats}}$$

This process can be repeated for N heartbeats to obtain a longer sample of the noise. Figure 17 shows the noise signals in one heartbeat when the subject was walking while being recorded with the BIOPAC System.

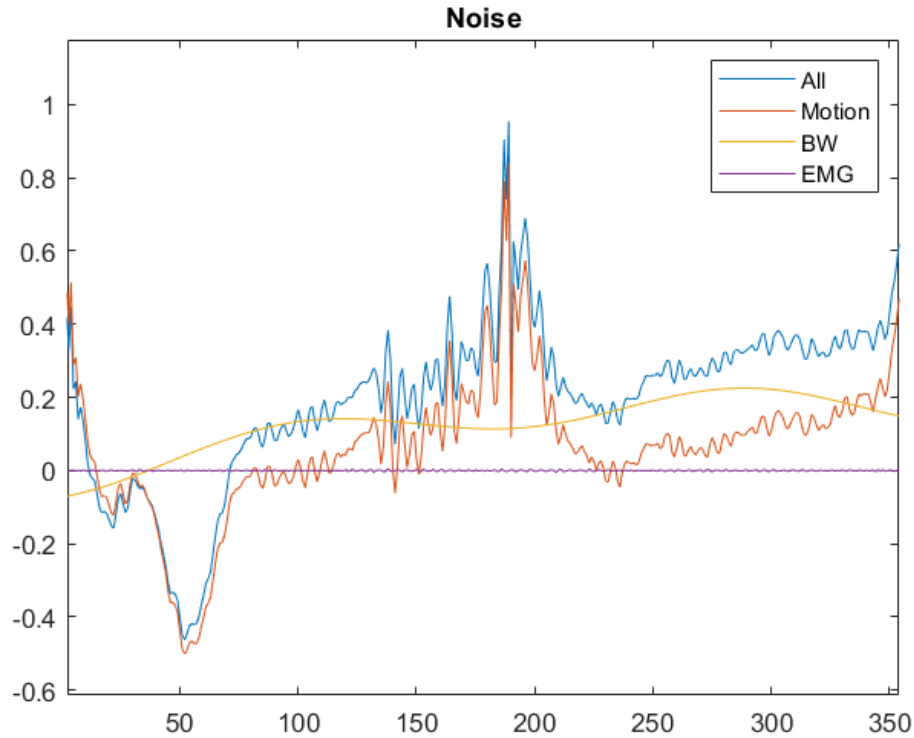


Figure 17. This figure shows all types of noise extracted from an ECG signal. The blue line is the combination of all noises. The red line represents the EMA. The yellow line represents the BW. The violet line represents the EMG.

It is worth noting that in the Hitoe System and Vitalgram System The signals were barely, if not at all, affected by EMG and BW noise. Therefore, almost all of the noise came from EMA.

4.5 Signal-to-Noise Ratio Calculation

As mentioned in chapter 1, the signal to noise ratio is a measure of the relationship between the power of the signal and the power of the noise. In this research, both the signal and the noise are voltages measured across the same impedance, so the SNR is calculated using the averages of the powers of the noise and the signal, in other words, it is calculated using the root mean square (RMS) voltage squared.

To calculate the SNR for BW the following equation was used:

$$SNR_{BW} = 10\log_{10}\left(\frac{\sum_{i=1}^n ECG_{clean_i}^2}{\sum_{i=1}^n BW_i^2}\right)$$

To calculate the SNR for EMG the following equation was used:

$$SNR_{EMG} = 10\log_{10}\left(\frac{\sum_{i=1}^n ECG_{clean_i}^2}{\sum_{i=1}^n EMG_i^2}\right)$$

To calculate the SNR for EMA the following equation was used:

$$SNR_{EMA} = 10\log_{10}\left(\frac{\sum_{i=1}^n ECG_{clean_i}^2}{\sum_{i=1}^n EMA_i^2}\right)$$

Chapter 5: Results

5.1 ECG Signal Quality of the Tested Devices

After isolating each type of noise from motion, the equations in chapter 4.5 were used to calculate their SNR values for the walking and running motions. These calculations were done for the three systems.

The BIOPAC system presented an SNR of 9.7 dB for BW, an SNR of 46 dB for EMG and an SNR of 3.8 dB for EMA for the walking motion. As for the running motion, the BIOPAC system presented an SNR of 3.4 dB for BW, 30.1 dB for EMG, and -3.5 dB for EMA. Figure 18 shows the SNR values for the BIOPAC system.

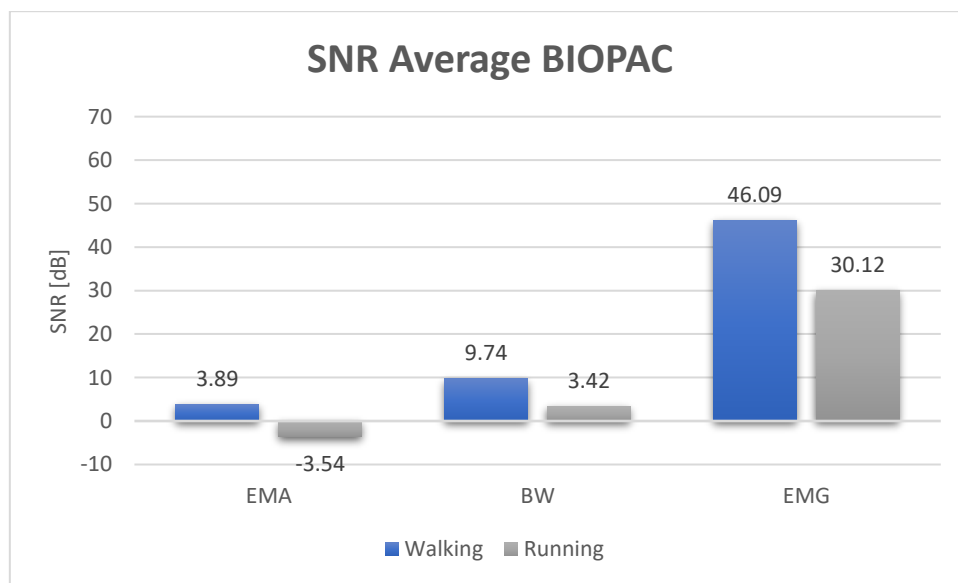


Figure 18. Average of the SNR values for the walking and running motions recorded with BIOPAC.

The Hitoe system presented an SNR of 11.4 dB for BW, an SNR of 35.8 dB for EMG and an SNR of 3.1 dB for EMA for the walking motion. As for the running motion, the Hitoe system presented an SNR of 11.32 dB for BW, 35.7 dB for EMG, and -0.26 dB for EMA. Figure 19 shows the SNR values for the Hitoe system.

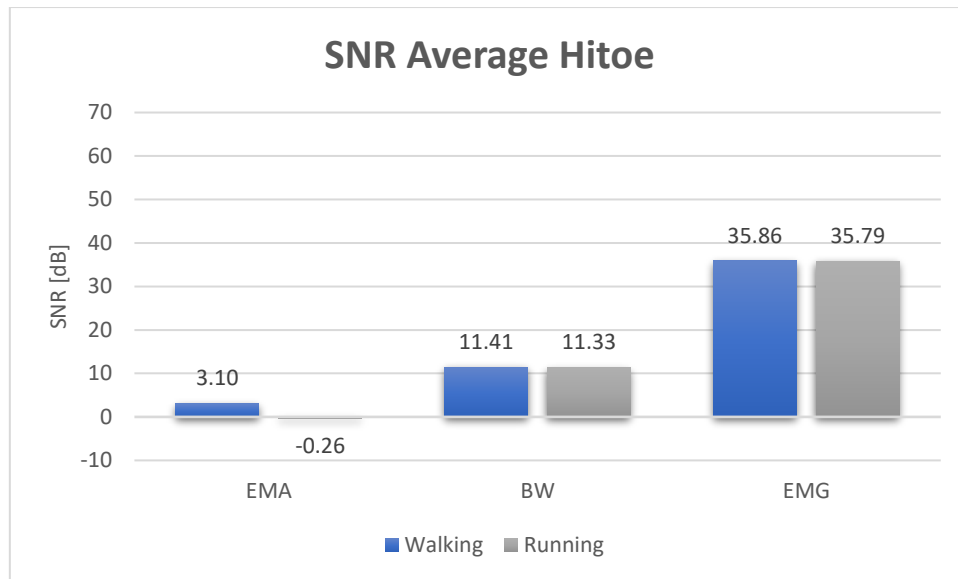


Figure 19. Average of the SNR values for the walking and running motions recorded with Hitoe.

The Vitalgram system presented an SNR of 50.5 dB for BW, an SNR of 45.4 dB for EMG and an SNR of 8 dB for EMA for the walking motion. As for the running motion, the Vitalgram system presented an SNR of 53.3 dB for BW, 44 dB for EMG, and 3.55 dB for EMA. Figure 20 shows the SNR values for the Vitalgram system.

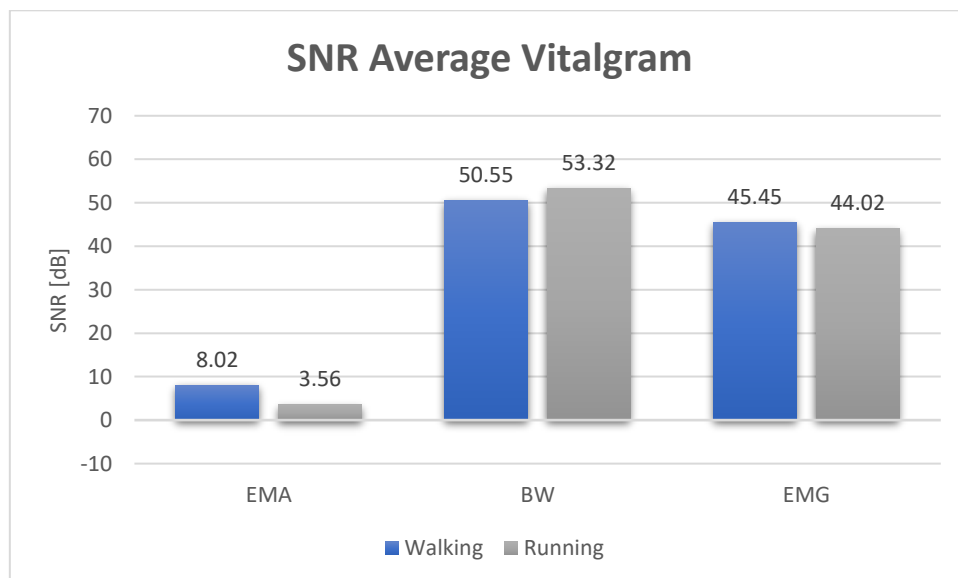


Figure 20. Average of the SNR values for the walking and running motions recorded with Vitalgram.

5.2 Device Performance According to Apandi et al.

Apandi simulated noisy ECG signals using the MIT-BIH database. She simulated noisy signals with BW, EMG and EMA with different SNR values from -9 dB to 12 dB. Then, she tested several algorithms for heartbeat detection. The algorithms she tested were the Pan-Tompkins, Hamilton, WQRS and Apandi et al. proposed algorithm.

To measure the performance of the algorithms she used the parameter of Sensitivity (SE) which denotes the percentage of true positive beats that are correctly identified. She also used the parameter of Positive Predictivity (PP) to describe the percentage of detected true heartbeats [18].

$$SE = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \times 100\%$$

$$PP = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \times 100\%$$

In Apandi's research, all the algorithms had an excellent performance at detecting heartbeats in signals contaminated with BW noise as shown in Figure 21. Moreover, all the devices presented a positive SNR value for the BW noise. For those reasons it has been determined that all of the algorithms would have a 100% of SE and PP if they were applied on recordings from these devices.

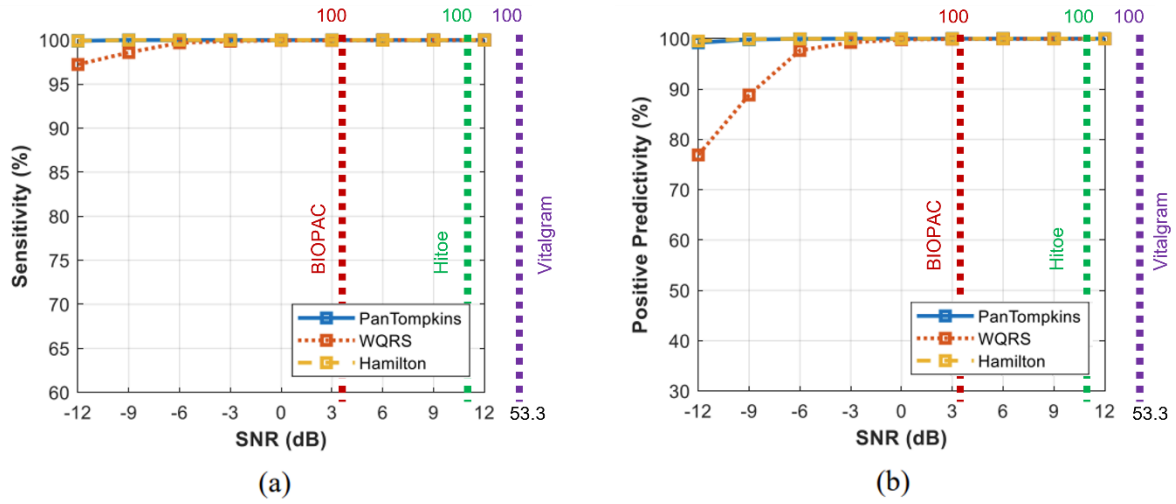


Figure 21. (a)Relationship between SE performance of heartbeat detection for BW and the predicted performance for each device. (b)Relationship between PP performance of heartbeat detection for BW and the predicted performance for each device.

Similarly, the devices presented very positive SNR values for the EMG noise for the most noise inducing activity (running). The values were 30.1 dB, 35.7 dB and 44 dB for the BIOPAC, Hitoe and Vitalgram respectively. For this reason, it was determined that the SE and PP for the EMG noise is 100% for all of the algorithms if they were applied on recordings from these devices.

For the BIOPAC system, the SNR value for EMA while running was -3.54 dB. According to Apandi's research, the WQRS algorithm presents the worst results. The SE and PP for recordings of the BIOPAC system would be 89% and 40.7%, respectively. Apandi's proposed method presents the best results. 97.7% and 95%, for SE and PP, respectively. These results are shown in Figure 22.

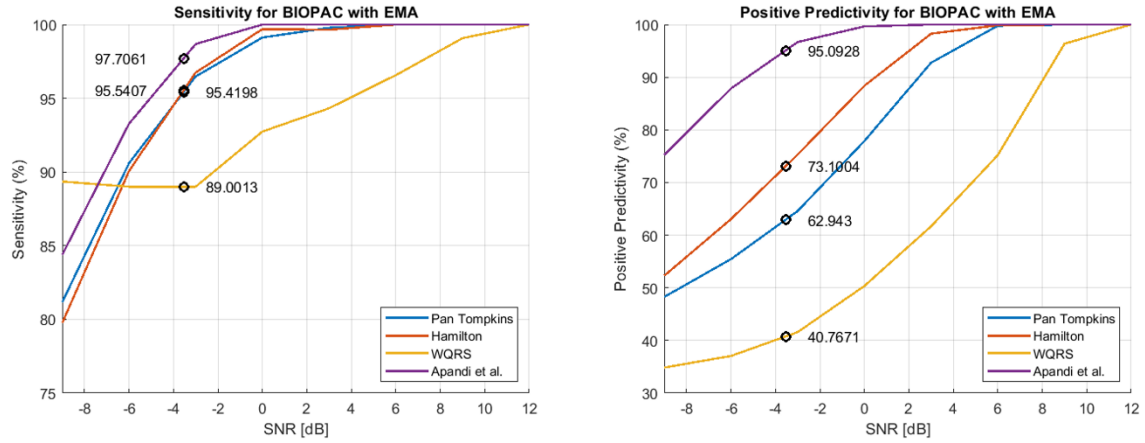


Figure 22. Performance of the heartbeat detection algorithms. (a) Relationship between SE performance of heartbeat detection for EMA and the estimated performance for BIOPAC ECG recordings. (b) Relationship between PP performance of heartbeat detection for EMA and the estimated performance for BIOPAC ECG recordings.

For the Hitoe system, the SNR value for EMA while running was -0.26 dB. According to Apandi's research, once again, the WQRS algorithm presents the worst results. The SE and PP for recordings of the Hitoe system would be 92.4% and 49.5%, respectively. Apandi's proposed method presents the best results. 99.8% and 99.3%, for SE and PP, respectively. These results are shown in Figure 23.

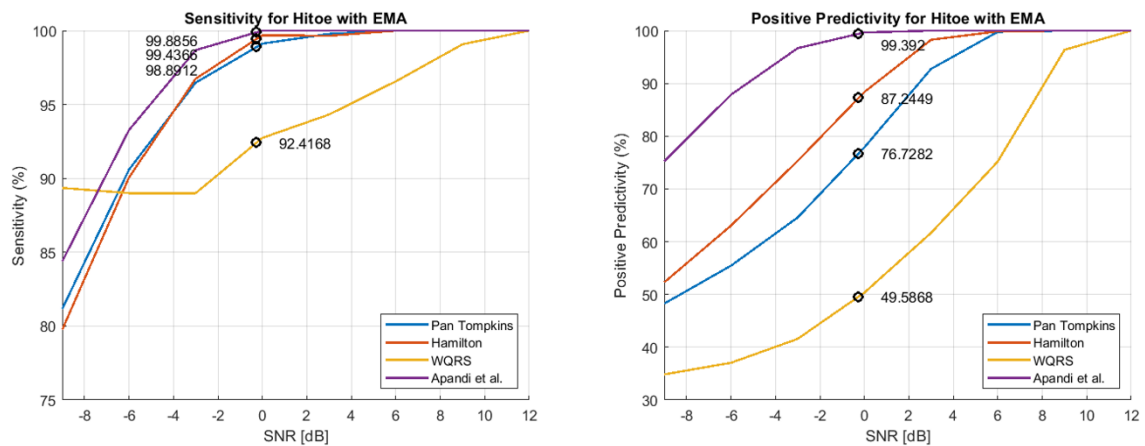


Figure 23. Performance of the heartbeat detection algorithms. (a) Relationship between SE performance of heartbeat detection for EMA and the estimated performance for Hitoe ECG recordings. (b) Relationship between PP performance of heartbeat detection for EMA and the estimated performance for Hitoe ECG recordings.

(b) Relationship between PP performance of heartbeat detection for EMA and the predicted performance for Hitoe ECG recordings.

For the Vitalgram system, the SNR value for EMA while running was 3.56 dB. According to Apandi's research, once more, the WQRS algorithm presents the worst results. The SE and PP for recordings of the Vitalgram system would be 94.7% and 64.2%, respectively. Apandi's proposed method presents the best results. 100% for both SE and PP. These results are shown in Figure 24.

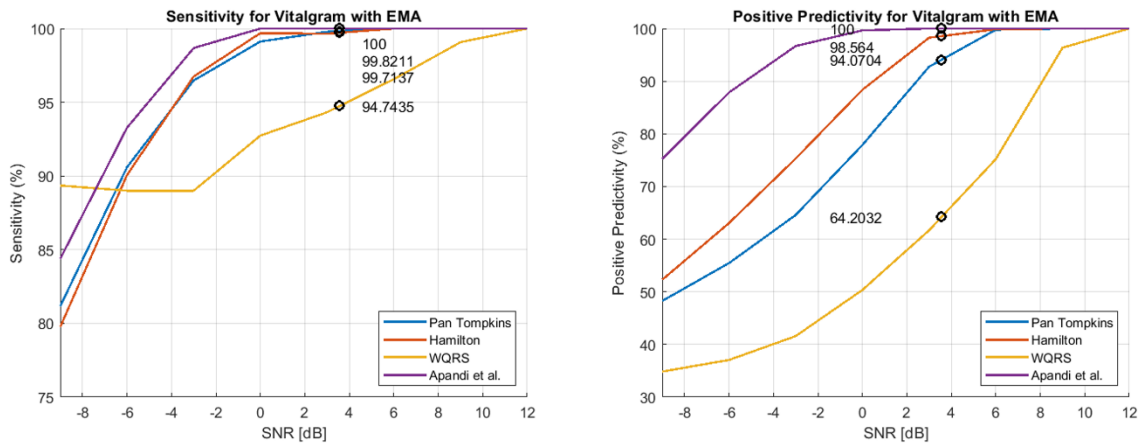


Figure 24. Performance of the heartbeat detection algorithms. (a) Relationship between SE performance of heartbeat detection for EMA and the estimated performance for Vitalgram ECG recordings. (b) Relationship between PP performance of heartbeat detection for EMA and the estimated performance for Vitalgram ECG recordings.

Chapter 6: Conclusions and Future

6.1 Conclusions

Observing the overall SNR results, it can be seen that the type of noise that affects the ECG recordings the most are the electrode motion artifacts (EMA).

Looking at the results of the SNR calculations, it can be seen that all the devices presented and SNR greater than 30 dB for the EMG noise for both the walking and running motions. This is evidence that the EMG barely affects the recordings of these devices.

Even though the BIOPAC system had positive SNR values for EMA and BW for the walking motion, for the running motion the SNR for BW was only slightly positive, meaning that the BW is mildly affecting the ECG while running. Moreover, for EMA, the BIOPAC system had the most negative value (-3.54), meaning that the noise's power is greater than the ECG signal's power.

The Hitoe system's BW SNR values were positive but not as positive as the values of the Vitalgram system. Hitoe's SNR values for the walking and running motions were close to 0 (3.1 dB and -0.26 dB, respectively), indicating that motion, if not severely, it still affects the quality of the signal significantly.

The SNR values of the Vitalgram for EMG and BW were very positive (over 44 dB), so it can be said that these noises do not affect the signal. Vitalgram's SNR values for the walking and running motions were 8 dB and 3.5 dB, respectively, indicating that the ECG signal was only mildly affected.

According to Apandi's research, the algorithms presented an excellent performance on recordings with BW noise. Since all the devices presented a positive SNR value for in running conditions, the heartbeat detection would be 100% on any

device using any of the algorithms. For the EMG noise, all devices presented SNR values over 30 dB. Therefore, the heartbeat detection performance would be 100% on any device using any of the algorithms.

For the EMA, since the Vitalgram has better SNR values than the rest of the devices, the algorithms would perform better on Vitalgram's recordings. According to the SNR values and Apandi's research, it can be estimated that the Apandi's proposed method would have 100% performance on recordings from the Vitalgram system.

Taking into consideration all the previous statements, it can be concluded that the Vitalgram system is better suited than the other systems to take samples and continue this line of research.

7.2 Future Work

According to the results from this research, the Vitalgram system shall be used to take more samples and continue further research. The samples must be taken from real patients working in collaboration with the hospital of Mie University. This is important because, in order to test and develop heartbeat classification algorithms, machine learning must be used. The samples shall contain annotations by experts from the hospital of Mie University that will help with this purpose.

The final goal of this line of research is to develop a wearable electrocardiograph that is not only capable of ECG monitoring in real-time but also capable of processing the signal and be able to classify each heartbeat whether they are normal or present some sort of abnormality, while the person wearing the device performs daily activities.

Appendix A

“changeEcgFreq” function

```
function [tResamp,ecgResamp] = changeEcgFreq(tRest,ecgRest,ecgFreqRest,ecgFreqMove)
%changeEcgFreq: Changes the heart rate of an ecg sample.
%               tRest,ecgRest,ecgFreqRest,ecgFreqMove
%   Changes the time vector of the ECG sample by a factor determined by the
%   heart rate of the ECG sample and a target heart rate.
%   The new signal (time vector and ECG) has now a new sampling frequency.
%   Higher if the heart rate was increased. Lower if the heart rate was
%   decreased.
%   To match the original sampling frequency, the ECG signal is now
%   resampled.
%   The output arguments correspond to a time vector and ECG that have a
%   different frequency than the original, but have the original sampling
%   frequency.

tRestTemp = tRest*(ecgFreqRest/ecgFreqMove);
delayFactor = tRest(1) - tRestTemp(1);
tRest2 = tRestTemp + delayFactor;
samplingFreq1 = ceil((1/(tRest(2)-tRest(1))));
samplingFreq2 = ceil((1/(tRest2(2)-tRest2(1))));
ecgResamp = resample(ecgRest,samplingFreq1,samplingFreq2);
tResamp = [];
temp = 0;
for i = 1:size(ecgResamp)
    tResamp(i) = temp;
    temp = temp + (1/samplingFreq1);
end
```

“waveletPeakDetection” function

```
function [rPeaks_wavelet,locs_wavelet,rPeaks,locs,y] =
waveletPeakDetection(t,ecg,level,band1,band2,minPeakH1,minPeakD1,minPeakH2,minPeakD2)
%waveletPeakDetection: Detects R peaks
%   t,ecg,level,band1,band2,minPeakH1,minPeakD1,minPeakH2,minPeakD2
%   Deconstruction and reconstruction of the ECG signal
%   using the DWT transformation method.
%   After the reconstruction is done, a simple findpeaks function
%   is used to find the R-peak locations.

waveletTransform = modwt(ecg,level);
waveletRecon = zeros(size(waveletTransform));
waveletRecon(band1:band2,:) = waveletTransform(band1:band2,:);
y = imodwt(waveletRecon,'sym4');
y = abs(y).^2;
[rPeaks_wavelet,locs_wavelet] =
findpeaks(y,t,'MinPeakHeight',minPeakH1,'MinPeakDistance',minPeakD1);
[rPeaks,locs] =
findpeaks(ecg,t,'MinPeakHeight',minPeakH2,'MinPeakDistance',minPeakD2);

end
```

Script used to analyze ECG recordings (Vitalgram, Subject 004)

```
% ----- Vitalgram Subject 004 Script v3 -----  
%-----  
clc  
clear  
close all  
  
%% ----- Read ECG file -----  
[file, path] = uigetfile('D:\Documents\Academic\Mie University\Master Thesis  
Research\Data Samples\VITALGRAM\2021-12-29-Electrocardiogram.csv');  
filename = strcat(path, file);  
table1 = readtable(filename);  
table1.Properties.VariableNames{'Time_HH_mm_ss_SSSSSSS'} = 'Time_ECG';  
VarNames = table1.Properties.VariableNames; % Recover Variable Names (Column  
Titles)  
%ColIdx = find(strcmp(VarNames, 'c')); % Return Column Index For Variable  
Name 'c'  
  
%% ----- Read Acceleration file -----  
[file2, path2] = uigetfile('D:\Documents\Academic\Mie University\Master Thesis  
Research\Data Samples\VITALGRAM\2021-12-29-Acceleration.csv');  
filename2 = strcat(path2, file2);  
table2 = readtable(filename2);  
table2.Properties.VariableNames{'Time_HH_mm_ss_SSSSSSS'} = 'Time_ACC';  
VarNames2 = table2.Properties.VariableNames; % Recover Variable Names (Column  
Titles)  
%ColIdx = find(strcmp(VarNames, 'c')); % Return Column Index For Variable  
Name 'c'  
  
% Sampling Frequency  
Fs = 256; % If Vitalgram  
% Fs = 200;  
  
% 11 bit resolution  
BD_Constant = 3.3/1023;  
  
% ECG amplitude and time vectors  
mV = table1.('Amplitude')* BD_Constant;  
  
%mV = mV - mean(mV);  
%mV = table1.('Amplitude');  
tECG = table1.('Time_ECG');  
tECGduration = tECG;  
tECGseconds = seconds(tECG); % Change time vector from duration to seconds  
ax = table2.('X_axis_g');  
ay = table2.('Y_axis_g');  
az = table2.('Z_axis_g');  
at = table2.('Time_ACC');  
  
% figure(1);  
% subplot(2,1,1);  
% plot(mV);  
% subplot(2,1,2);  
% plot(ax);  
% hold on  
% plot(ay);  
% hold on  
% plot(az);  
%% ----- Divide Sample by Action -----
```

```
[ecgLying, tLying, ecgSitting, tSitting, ecgStanding, tStanding, ecgWalking,
tWalking, ecgJogging, tJogging, ecgUpDownLeft, tUpDownLeft, ecgUpDownRight,
tUpDownRight, ecgPushPullLeft, tPushPullLeft, ecgPushPullRight, tPushPullRight,
ecgWavingLeft, tWavingLeft, ecgWavingRight, tWavingRight, ecgStandUp, tStandUp,
ecgSitDown, tSitDown] = divideEcgSample(tECGseconds,
mV,80000,81280,90000,91280,94000,95280,106000,107280,116000,117280,132100,133380,14
3400,144680,157200,158480,170000,171280,183000,184280,191800,193080,196900,198180,1
98180,199460);
```

```
%% ----- Plot Whole Test -----%
figure(1); %Full Test Plot
axisX(2) = subplot(2,1,1);
plot(tECG,mV); %ECG Plot
axisX(2) = subplot(2,1,2);
title('ECG - Vitalgram');
xlabel('Time [s]');
ylabel('Voltage [V]');
hold on
plot(at,ax); %Acc X plot
plot(at,ay); %Acc Y plot
plot(at,az); %Acc Z plot
title('ECG - Vitalgram');
xlabel('Time [s]');
ylabel('Acceleration [g]');
hold off
linkaxes(axisX,'x');

%% ----- Max Value For Each Test -----%
maxEcgLying = max(ecgLying);
maxEcgSitting = max(ecgSitting);
maxEcgStanding = max(ecgStanding);
maxEcgWalking = max(ecgWalking);
maxecgJogging = max(ecgJogging);
maxEcgUpDownLeft = max(ecgUpDownLeft);
maxEcgUpDownRight = max(ecgUpDownRight);
maxEcgPushPullLeft = max(ecgPushPullLeft);
maxEcgPushPullRight = max(ecgPushPullRight);
maxEcgWavingLeft = max(ecgWavingLeft);
maxEcgWavingRight = max(ecgWavingRight);
maxEcgStandUp = max(ecgStandUp);
maxEcgSitDown = max(ecgSitDown);

%% ----- Normalize All Tests -----%
ecgLying = ecgLying/max(ecgLying);
ecgSitting = ecgSitting/max(ecgSitting);
ecgStanding = ecgStanding/max(ecgStanding);
ecgWalking = ecgWalking/max(ecgWalking);
ecgJogging = ecgJogging/max(ecgJogging);
ecgUpDownLeft = ecgUpDownLeft/max(ecgUpDownLeft);
ecgUpDownRight = ecgUpDownRight/max(ecgUpDownRight);
ecgPushPullLeft = ecgPushPullLeft/max(ecgPushPullLeft);
ecgPushPullRight = ecgPushPullRight/max(ecgPushPullRight);
ecgWavingLeft = ecgWavingLeft/max(ecgWavingLeft);
ecgWavingRight = ecgWavingRight/max(ecgWavingRight);
ecgStandUp = ecgStandUp/max(ecgStandUp);
ecgSitDown = ecgSitDown/max(ecgSitDown);

%% ----- Create Sample Vector For All Tests -----%
samplesLying = 1:size(ecgLying);
samplesSitting = 1:size(ecgSitting);
samplesStanding = 1:size(ecgStanding);
samplesWalking = 1:size(ecgWalking);
samplesJogging = 1:size(ecgJogging);
samplesUpDownLeft = 1:size(ecgUpDownLeft);
samplesUpDownRight = 1:size(ecgUpDownRight);
samplesPushPullLeft = 1:size(ecgPushPullLeft);
samplesPushPullRight = 1:size(ecgPushPullRight);
samplesWavingLeft = 1:size(ecgWavingLeft);
```

```

samplesWavingRight = 1:size(ecgWavingRight);
samplesStandUp = 1:size(ecgStandUp);
samplesSitDown = 1:size(ecgSitDown);
samplesEcg = 1:size(mV);

%% ----- Get Baseline Wandering (Low-Pass) -----
lpFilt = designfilt('lowpassiir','FilterOrder',12, ...
    'PassbandFrequency',1,'PassbandRipple',0.001, ...
    'SampleRate',Fs);
fvtool(lpFilt)
baselineWanderShifted = filter(lpFilt,mV);

[c, lag] = xcorr(mV, baselineWanderShifted); % Find Correlation
[M, I] = max(c); % Find Max Correlation Index
lags = lag(I) % Find Max Correlation lag
baselineWander = circshift(baselineWanderShifted, lags);

figure(201);
plot(mV);
hold on
plot(baselineWander,'linewidth',1.5);
%% ----- Divide Baseline Wandering
[bwLying, tLying, bwSitting, tSitting, bwStanding, tStanding, bwWalking, tWalking,
bwJogging, tJogging, bwUpDownLeft, tUpDownLeft, bwUpDownRight, tUpDownRight,
bwPushPullLeft, tPushPullLeft, bwPushPullRight, tPushPullRight, bwWavingLeft,
tWavingLeft, bwWavingRight, tWavingRight, bwStandUp, tStandUp, bwSitDown, tSitDown]
=
divideEcgSample(tECG,
baselineWander,80000,81280,90000,91280,94000,95280,106000,107280,116000,117280,1321
00,133380,143400,144680,157200,158480,170000,171280,182800,184080,191800,193080,196
900,198180,198180,199460);

%% Normalize Baseline Wandering
bwLying = bwLying/max(ecgLying);
bwSitting = bwSitting/max(ecgSitting);
bwStanding = bwStanding/max(ecgStanding);
bwWalking = bwWalking/max(ecgWalking);
bwJogging = bwJogging/max(ecgJogging);
bwUpDownLeft = bwUpDownLeft/max(ecgUpDownLeft);
bwUpDownRight = bwUpDownRight/max(ecgUpDownRight);
bwPushPullLeft = bwPushPullLeft/max(ecgPushPullLeft);
bwPushPullRight = bwPushPullRight/max(ecgPushPullRight);
bwWavingLeft = bwWavingLeft/max(ecgWavingLeft);
bwWavingRight = bwWavingRight/max(ecgWavingRight);
bwStandUp = bwStandUp/max(ecgStandUp);
bwSitDown = bwSitDown/max(ecgSitDown);

%% ----- Get EMG Noise (High-Pass) -----
hpFilt = designfilt('highpassiir','FilterOrder',12, ...
    'PassbandFrequency',50,'PassbandRipple',0.01, ...
    'SampleRate',Fs);
fvtool(hpFilt)
emgNoiseShifted = filter(hpFilt,mV);

[c, lag] = xcorr(mV,emgNoiseShifted); % Find Correlation
[M, I] = max(c); % Find Max Correlation Index
lags = lag(I) % Find Max Correlation lag

emgNoise = circshift(emgNoiseShifted, lags);

figure(202);plot(mV - baselineWander);
hold on
plot(emgNoise,'linewidth',1.5);

%% ----- Divide EMG noise

```

```

[emgLying, tLying, emgSitting, tSitting, emgStanding, tStanding, emgWalking,
tWalking, emgJogging, tJogging, emgUpDownLeft, tUpDownLeft, emgUpDownRight,
tUpDownRight, emgPushPullLeft, tPushPullLeft, emgPushPullRight, tPushPullRight,
emgWavingLeft, tWavingLeft, emgWavingRight, tWavingRight, emgStandUp, tStandUp,
emgSitDown, tSitDown] = divideEcgSample(tECG,
emgNoise,80000,81280,90000,91280,94000,95280,106000,107280,116000,117280,132100,133
380,143400,144680,157200,158480,170000,171280,182800,184080,191800,193080,196900,19
8180,198180,199460);
%% Normalize EMG Noise
emgLying = emgLying/max(ecgLying);
emgSitting = emgSitting/max(ecgSitting);
emgStanding = emgStanding/max(ecgStanding);
emgWalking = emgWalking/max(ecgWalking);
emgJogging = emgJogging/max(ecgJogging);
emgUpDownLeft = emgUpDownLeft/max(ecgUpDownLeft);
emgUpDownRight = emgUpDownRight/max(ecgUpDownRight);
emgPushPullLeft = emgPushPullLeft/max(ecgPushPullLeft);
emgPushPullRight = emgPushPullRight/max(ecgPushPullRight);
emgWavingLeft = emgWavingLeft/max(ecgWavingLeft);
emgWavingRight = emgWavingRight/max(ecgWavingRight);
emgStandUp = emgStandUp/max(ecgStandUp);
emgSitDown = emgSitDown/max(ecgSitDown);

%% ----- Change Time Duration Array to Seconds Array -----%
tECG = seconds(tECG); % Change time vector from duration to seconds

%% ----- Selecting the clean heartbeat -----%
cleanEcgSignal = ecgSitting/max(ecgSitting);
samplesCleanEcgSignal = (1:length(cleanEcgSignal))';
[qrspkaps_wavelet,locs_wavelet,qrspkaps,locs,y] =
waveletPeakDetection(samplesCleanEcgSignal,cleanEcgSignal,5,3,4,0.25,64,0.7,64);

figure(91);
rawData = plot(samplesCleanEcgSignal,cleanEcgSignal,'k','linewidth',0.75);
hold on
waveletRec = plot(samplesCleanEcgSignal,y,'r','linewidth',1);
hwav = plot(locs_wavelet,qrspkaps_wavelet,'bo');
title('R peak detection - Sitting (Clean ECG)');
legend([hwav rawData waveletRec],'R Peak Locations','Raw Data','Wavelet
Reconstruction','Location','Southeast');
xlabel('Samples');
hold off;

cleanHeartBeat = cleanEcgSignal(locs_wavelet(4):locs_wavelet(5));
tCleanHeartBeat = ((0:length(cleanHeartBeat)-1)/Fs)';
periodCleanHeartBeat = length(cleanHeartBeat)/Fs; % In seconds
cleanHeartRate = 1/periodCleanHeartBeat;

%% ----- Check Noisy signal R peaks Walking -----%
[qrspkaps_wavelet,locs_wavelet,qrspkaps,locs,y] =
waveletPeakDetection(samplesWalking,ecgWalking,5,3,4,0.25,64,0.7,64);
figure(4);
rawData = plot(samplesWalking,ecgWalking,'k','linewidth',0.75);
hold on
waveletRec = plot(samplesWalking,y,'r','linewidth',1);
hwav = plot(locs_wavelet,qrspkaps_wavelet,'bo');
title('R-peak detection - Check Walking');
legend([hwav rawData waveletRec],'R Peaks Locations','Raw Data','Wavelet
Reconstruction','Location','Southeast');
xlabel('Samples');
hold off;

[qrspkaps_wavelet,locs_wavelet,qrspkaps,locs,y] =
waveletPeakDetection(samplesWalking,ecgWalking,5,3,4,0.25,64,0.7,64);
figure(92);
rawData = plot(samplesWalking,ecgWalking,'k','linewidth',0.75);

```

```

hold on
waveletRec = plot(samplesWalking,y,'r','linewidth',1);
hwav = plot(locs_wavelet,qrspeaks_wavelet,'bo');
title('R-peak detection - Walking (Noisy ECG)');
legend([hwav rawData waveletRec],'R peak locations','Raw Data','Wavelet Reconstruction','Location','Southeast');
xlabel('Samples');
hold off;

ecgWalkingSemiClean = ecgWalking - bwWalking - emgWalking;
ecgWalkingSemiClean = ecgWalkingSemiClean / max(ecgWalkingSemiClean);
%% ----- Calculate SNR for "N" heartbeats Walking -----%
motionNoise = [];
sumXs = 0;
sumYs = 0;

for i = 1:length(locs_wavelet)-1
    per = (locs_wavelet(i+1)-1-locs_wavelet(i))+1;
    period = per/Fs;
    goalheartRate = 1/period;
    [tCleanHeartBeatResamp, cleanHeartBeatResamp] =
changeEcgFreq(tCleanHeartBeat, cleanHeartBeat, cleanHeartRate, goalheartRate);
    cleanHeartBeatResamp = cleanHeartBeatResamp / max(cleanHeartBeatResamp);
    ecgWalkingSemiCleanPortion =
ecgWalkingSemiClean(locs_wavelet(i):locs_wavelet(i+1)-1) /
max(ecgWalkingSemiClean(locs_wavelet(i):locs_wavelet(i+1)-1));
    %noise = ecgWalkingSemiClean(locs_wavelet(i):locs_wavelet(i+1)-1) -
cleanHeartBeatResamp;
    noise = ecgWalkingSemiCleanPortion - cleanHeartBeatResamp;
    sumXs = sumXs + sum((cleanHeartBeatResamp).^2);
    sumYs = sumYs + sum((noise).^2);
    for j = 1:length(noise)
        motionNoise(end+1,:) = noise(j);
    end
end
rSumXs = (sqrt(sumXs)).^2;
rSumYs = (sqrt(sumYs)).^2;

motionNoiseWalking = motionNoise;
allNoise = motionNoise + bwWalking(locs_wavelet(1):locs_wavelet(end)-1) +
emgWalking(locs_wavelet(1):locs_wavelet(end)-1);

%SNR (General) beat by beat
SNR_General_Walking =
10*log10((rssq(cleanEcgsignal(locs_wavelet(1):locs_wavelet(end)-
1)).^2)/(rssq(allNoise)).^2)

%SNR (Motion Noise) cleanEcgsignal / motionNoise Walking
SNR_Motion_Walking =
10*log10((rssq(cleanEcgsignal(locs_wavelet(1):locs_wavelet(end))).^2)/(rssq(motionN
oise)).^2)

%SNR (Baseline Wander) Sitting / bwWalking
SNR_BW_Walking =
10*log10((rssq(cleanEcgsignal(locs_wavelet(1):locs_wavelet(end))).^2)/(rssq(bwWalki
ng(locs_wavelet(1):locs_wavelet(end))).^2))

%SNR (EMG noise) Sitting / emgWalking
SNR_EMG_Walking =
10*log10((rssq(cleanEcgsignal(locs_wavelet(1):locs_wavelet(end))).^2)/(rssq(emgWalk
ing(locs_wavelet(1):locs_wavelet(end))).^2))

%% ----- Check Noisy signal R peaks Jogging -----%
[qrspeaks_wavelet, locs_wavelet, qrspeaks, locs, y] =
waveletPeakDetection(samplesJogging, ecgJogging, 5, 3, 4, 0.25, 64, 0.7, 64);
figure(5);

```

```

rawData = plot(samplesJogging,ecgJogging,'k','linewidth',0.75);
hold on
waveletRec = plot(samplesJogging,y,'r','linewidth',1);
hwav = plot(locs_wavelet,qrspkaps_wavelet,'bo');
title('R-peak detection - Check Jogging');
legend([hwav rawData waveletRec],'R Peaks Locations','Raw Data','Wavelet Reconstruction','Location','Southeast');
xlabel('Samples');
hold off;

ecgJoggingSemiClean = ecgJogging - bwJogging - emgJogging;
ecgJoggingSemiClean = ecgJoggingSemiClean / max(ecgJoggingSemiClean);
% ----- Calculate SNR for "N" heartbeats Jogging -----%
motionNoise = [];
sumXs = 0;
sumYs = 0;

for i = 1:length(locs_wavelet)-1
    per = (locs_wavelet(i+1)-locs_wavelet(i))+1;
    period = per/Fs;
    goalheartRate = 1/period;
    [tCleanHeartBeatResamp, cleanHeartBeatResamp] =
changeEcgFreq(tCleanHeartBeat, cleanHeartBeat, cleanHeartRate, goalheartRate);
    cleanHeartBeatResamp = cleanHeartBeatResamp / max(cleanHeartBeatResamp);
    ecgJoggingSemiCleanPortion =
ecgJoggingSemiClean(locs_wavelet(i):locs_wavelet(i+1)-1) /
max(ecgJoggingSemiClean(locs_wavelet(i):locs_wavelet(i+1)-1));
    %noise = ecgJoggingSemiClean(locs_wavelet(i):locs_wavelet(i+1)-1) -
cleanHeartBeatResamp;
    noise = ecgJoggingSemiCleanPortion - cleanHeartBeatResamp;
    sumXs = sumXs + sum((cleanHeartBeatResamp).^2);
    sumYs = sumYs + sum((noise).^2);
    for j = 1:length(noise)
        motionNoise(end+1,:) = noise(j);
    end
end
rSumXs = (sqrt(sumXs)).^2;
rSumYs = (sqrt(sumYs)).^2;

motionNoiseJogging = motionNoise;
allNoise = motionNoise + bwJogging(locs_wavelet(1):locs_wavelet(end)-1) +
emgJogging(locs_wavelet(1):locs_wavelet(end)-1);

%SNR (General) beat by beat
SNR_General_Jogging =
10*log10((rssq(cleanEcgsignal(locs_wavelet(1):locs_wavelet(end)-1)).^2)/rssq(allNoise).^2)

%SNR (Motion Noise) cleanEcgsignal / motionNoise Jogging
SNR_Motion_Jogging =
10*log10((rssq(cleanEcgsignal(locs_wavelet(1):locs_wavelet(end))).^2)/(rssq(motionNoise).^2))

%SNR (Baseline Wander) Sitting / bwJogging
SNR_BW_Jogging =
10*log10((rssq(cleanEcgsignal(locs_wavelet(1):locs_wavelet(end))).^2)/(rssq(bwJogging(locs_wavelet(1):locs_wavelet(end))).^2))

%SNR (EMG noise) Sitting / emgJogging
SNR_EMG_Jogging =
10*log10((rssq(cleanEcgsignal(locs_wavelet(1):locs_wavelet(end))).^2)/(rssq(emgJogging(locs_wavelet(1):locs_wavelet(end))).^2))
%SNR_EMG_Jogging2
snr(cleanEcgsignal(locs_wavelet(1):locs_wavelet(end)),emgJogging(locs_wavelet(1):locs_wavelet(end)))

```

```

%SNR_EMG_Jogging3
10*log10((sum((cleanEcgSignal(locs_wavelet(1):locs_wavelet(end))).^2))/(sum((emgJogging(locs_wavelet(1):locs_wavelet(end))).^2)))

%% ----- Check Noisy signal R peaks WavingLeft -----%
[qrspeaks_wavelet, locs_wavelet, qrspeaks, locs, y]
waveletPeakDetection(samplesWavingLeft, ecgWavingLeft, 5, 3, 4, 0.25, 64, 0.7, 64);
figure(6);
rawData = plot(samplesWavingLeft, ecgWavingLeft, 'k', 'linewidth', 0.75);
hold on
waveletRec = plot(samplesWavingLeft, y, 'r', 'linewidth', 1);
hwav = plot(locs_wavelet, qrspeaks_wavelet, 'bo');
title('R-peak detection - Check WavingLeft');
legend([hwav rawData waveletRec], 'R Peak Locations', 'Raw Data', 'Wavelet Reconstruction', 'Location', 'Southeast');
xlabel('Samples');
hold off;

ecgWavingLeftSemiClean = ecgWavingLeft - bwWavingLeft - emgWavingLeft;
ecgWavingLeftSemiClean = ecgWavingLeftSemiClean / max(ecgWavingLeftSemiClean);
%% ----- Calculate SNR for "N" heartbeats WavingLeft -----%
motionNoise = [];
sumXs = 0;
sumYs = 0;

for i = 1:length(locs_wavelet)-1
    per = (locs_wavelet(i+1)-1-locs_wavelet(i))+1;
    period = per/Fs;
    goalheartRate = 1/period;
    [tCleanHeartBeatResamp, cleanHeartBeatResamp]
changeEcgFreq(tCleanHeartBeat, cleanHeartBeat, cleanHeartRate, goalheartRate);
% length(cleanHeartBeatResamp)
% length(Ya(locs_wavelet(i):locs_wavelet(i+1)))
noise = ecgWavingLeftSemiClean(locs_wavelet(i):locs_wavelet(i+1)-1) - cleanHeartBeatResamp;
sumXs = sumXs + sum((cleanHeartBeatResamp).^2);
sumYs = sumYs + sum((noise).^2);
for j = 1:length(noise)
    motionNoise(end+1,:) = noise(j);
end
end
rSumXs = (sqrt(sumXs)).^2;
rSumYs = (sqrt(sumYs)).^2;

motionNoiseWavingLeft = motionNoise;
allNoise = motionNoise + bwWavingLeft(locs_wavelet(1):locs_wavelet(end)-1) + emgWavingLeft(locs_wavelet(1):locs_wavelet(end)-1);

%SNR (General) beat by beat
SNR_General_WavingLeft
10*log10((rssq(cleanEcgSignal(locs_wavelet(1):locs_wavelet(end)-1)).^2)/rssq(allNoise).^2)

%SNR (Motion Noise) cleanEcgSignal / motionNoise WavingLeft
SNR_Motion_WavingLeft
10*log10((rssq(cleanEcgSignal(locs_wavelet(1):locs_wavelet(end))).^2)/(rssq(motionNoise).^2)

%SNR (Baseline Wander) Sitting / bwWavingLeft
SNR_BW_WavingLeft
10*log10((rssq(cleanEcgSignal(locs_wavelet(1):locs_wavelet(end))).^2)/(rssq(bwWavingLeft(locs_wavelet(1):locs_wavelet(end))).^2)

%SNR (EMG noise) Sitting / emgWavingLeft

```



```

SNR_EMG_WavingLeft =
10*log10((rssq(cleanEcgsignal(locs_wavelet(1):locs_wavelet(end))).^2)/(rssq(emgWavingLeft(locs_wavelet(1):locs_wavelet(end))).^2))

%% ----- Check Noisy signal R peaks WavingRight -----%
[qrspeaks_wavelet,locs_wavelet,qrspeaks,locs,y] =
waveletPeakDetection(samplesWavingRight,ecgWavingRight,5,3,4,0.25,64,0.7,64);
figure(7);
rawData = plot(samplesWavingRight,ecgWavingRight,'k','linewidth',0.75);
hold on
waveletRec = plot(samplesWavingRight,y,'r','linewidth',1);
hwav = plot(locs_wavelet,qrspeaks_wavelet,'bo');
title('R-peak detection - Check WavingRight');
legend([hwav rawData waveletRec],'R Peaks Locations','Raw Data','Wavelet Reconstruction','Location','Southeast');
xlabel('Samples');
hold off;

ecgWavingRightSemiClean = ecgWavingRight - bwWavingRight - emgWavingRight;
ecgWavingRightSemiClean = ecgWavingRightSemiClean / max(ecgWavingRightSemiClean);
%% ----- Calculate SNR for "N" heartbeats WavingRight -----%
motionNoise = [];
sumXs = 0;
sumYs = 0;

for i = 1:length(locs_wavelet)-1
    per = (locs_wavelet(i+1)-1-locs_wavelet(i))+1;
    period = per/Fs;
    goalheartRate = 1/period;
    [tCleanHeartBeatResamp,cleanHeartBeatResamp] =
changeEcgsFreq(tCleanHeartBeat,cleanHeartBeat,cleanHeartRate,goalheartRate);
% length(cleanHeartBeatResamp)
% length(Ya(locs_wavelet(i):locs_wavelet(i+1)))
noise = ecgWavingRightSemiClean(locs_wavelet(i):locs_wavelet(i+1)-1)-
cleanHeartBeatResamp;
sumXs = sumXs + sum((cleanHeartBeatResamp).^2);
sumYs = sumYs + sum((noise).^2);
for j = 1:length(noise)
    motionNoise(end+1,:) = noise(j);
end
end
rSumXs = (sqrt(sumXs)).^2;
rSumYs = (sqrt(sumYs)).^2;

motionNoiseWavingRight = motionNoise;
allNoise = motionNoise + bwWavingRight(locs_wavelet(1):locs_wavelet(end)-1) +
emgWavingRight(locs_wavelet(1):locs_wavelet(end)-1);

%SNR (General) beat by beat
SNR_General_WavingRight =
10*log10((rssq(cleanEcgsignal(locs_wavelet(1):locs_wavelet(end)-1)).^2)/rssq(allNoise).^2)

%SNR (Motion Noise) cleanEcgsignal / motionNoise WavingRight
SNR_Motion_WavingRight =
10*log10((rssq(cleanEcgsignal(locs_wavelet(1):locs_wavelet(end))).^2)/(rssq(motionNoise).^2))

%SNR (Baseline Wander) Sitting / bwWavingRight
SNR_BW_WavingRight =
10*log10((rssq(cleanEcgsignal(locs_wavelet(1):locs_wavelet(end))).^2)/(rssq(bwWavingRight(locs_wavelet(1):locs_wavelet(end))).^2))

%SNR (EMG noise) Sitting / emgWavingRight

```

```

SNR_EMG_WavingRight
10*log10((rssq(cleanEcgSignal(locs_wavelet(1):locs_wavelet(end))).^2)/(rssq(emgWavingRight(locs_wavelet(1):locs_wavelet(end))).^2))

%% ----- Creating SNR Table and File -----
NoiseType = {'All Noise'; 'Motion'; 'BW'; 'EMG'};
Walking = [SNR_General_Walking; SNR_Motion_Walking; SNR_BW_Walking; SNR_EMG_Walking];
Running = [SNR_General_Jogging; SNR_Motion_Jogging; SNR_BW_Jogging; SNR_EMG_Jogging];
WavingLeft = [SNR_General_WavingLeft; SNR_Motion_WavingLeft; SNR_BW_WavingLeft; SNR_EMG_WavingLeft];
WavingRight = [SNR_General_WavingRight; SNR_Motion_WavingRight; SNR_BW_WavingRight; SNR_EMG_WavingRight];

SNR_Table = table(Walking, Running, WavingLeft, WavingRight, 'RowNames', NoiseType)

writetable(SNR_Table, 'SNR_Vit_004.xlsx', 'WriteRowNames', true);

```

References

- [1]"Cardiovascular diseases", Who.int, 2022. [Online]. Available: https://www.who.int/health-topics/cardiovascular-diseases#tab=tab_1. [Accessed: 12- Jan- 2022].
- [2]N. Srinivasan and R. Schilling, "Sudden Cardiac Death and Arrhythmias", *Arrhythmia & Electrophysiology Review*, vol. 7, no. 2, p. 111, 2018. Available: 10.15420/aer.2018:15:2.
- [3]P. Kowey and D. Kocovic, "Ambulatory Electrocardiographic Recording", *Circulation*, vol. 108, no. 5, 2003. Available: 10.1161/01.cir.0000082930.04238.8c.
- [4]J. Hall, M. Hall and A. Guyton, *Guyton and Hall textbook of Medical Physiology*, 14th ed. Elsevier, 2021.
- [5]S. Talebi, "The Wavelet Transform", Medium, 2022. [Online]. Available: <https://towardsdatascience.com/the-wavelet-transform-e9cfa85d7b34>. [Accessed: 15- Jan- 2022].
- [6] R. Polikar, *The Wavelet Tutorial*, 2nd edition. Available: <http://users.rowan.edu/~polikar/WTtutorial.html>
- [7] K. Amolins, Y. Zhang and P. Dare, "Wavelet based image fusion techniques — An introduction, review and comparison", *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 62, no. 4, pp. 249-263, 2007. Available: 10.1016/j.isprsjprs.2007.05.009.
- [8] "Signal-to-Noise Ratio Monitor - TechLibrary - Juniper Networks", Juniper.net, 2022. [Online]. Available: https://www.juniper.net/documentation/en_US/junos-

space-apps/network-director4.0/topics/reference/general/monitor-snr.html.

[Accessed: 14- Jan- 2022].

[9] "snr", www.mathworks.com, 2022. [Online]. Available: <https://www.mathworks.com/help/signal/ref/snr.html>. [Accessed: 14- Jan- 2022].

[10] "How to Calculate Signal to Noise Ratio", Horiba.com, 2022. [Online]. Available: <https://www.horiba.com/int/technology/spectroscopy/fluorescence-spectroscopy/how-to-calculate-signal-to-noise-ratio/>. [Accessed: 14- Jan- 2022].

[11]M. D'Aloia, A. Longo and M. Rizzi, "Noisy ECG Signal Analysis for Automatic Peak Detection", *Information*, vol. 10, no. 2, p. 35, 2019. Available: 10.3390/info10020035.

[12]J. Madeiro, *Developments and applications for ECG signal processing*. London: Academic Press, an imprint of Elsevier, 2019.

[13]M. Blanco-Velasco, B. Weng and K. Barner, "ECG signal denoising and baseline wander correction based on the empirical mode decomposition", *Computers in Biology and Medicine*, vol. 38, no. 1, pp. 1-13, 2008. Available: 10.1016/j.compbimed.2007.06.003.

[14]A. Boxtel, "Optimal signal bandwidth for the recording of surface EMG activity of facial, jaw, oral, and neck muscles", *Psychophysiology*, vol. 38, no. 1, pp. 22-34, 2001. Available: 10.1111/1469-8986.3810022.

[15]A. Pérez-Riera, R. Barbosa-Barros, R. Daminello-Raimundo and L. de Abreu, "Main artifacts in electrocardiography", *Annals of Noninvasive Electrocardiology*, vol. 23, no. 2, p. e12494, 2017. Available: 10.1111/anec.12494.

- [16]R. Haddadi, E. Abdelmounim, M. El Hanne and A. Belaguid, "Discrete Wavelet Transform Based Algorithm for Recognition of QRS Complexes", World of Computer Science and Information Technology Journal, vol. 4, no. 9, pp. 127-132, 2014. [Accessed 16 January 2022].
- [17] Patro K.K., Rajesh Kumar P. (2016) A Novel Frequency-Time Based Approach for the Detection of Characteristic Waves in Electrocardiogram Signal. In: Satapathy S., Rao N., Kumar S., Raj C., Rao V., Sarma G. (eds) Microelectronics, Electromagnetics and Telecommunications. Lecture Notes in Electrical Engineering, vol 372. Springer, New Delhi. https://doi.org/10.1007/978-81-322-2728-1_6
- [18] Z. F. M. Apandi, "Development of Noise-Tolerant Method for Arrhythmia Heartbeat Detection in Ambulatory Electrocardiogram", Ph.D., Mie University, 2021.
- [19] Moody GB, Mark RG. The impact of the MIT-BIH Arrhythmia Database. IEEE Eng in Med and Biol 20(3):45-50 (May-June 2001). (PMID: 11446209)
- [20]MATLAB MATHWORKS.