Digital Signal Processing of PPG

for evaluation of Atrial Fibrillation

Kaan Demir
Jasper Konijn

TU Delft

Digital Signal Processing of PPG

for evaluation of Atrial Fibrillation

by

Kaan Demir
Jasper Konijn

to obtain the degree of Bachelor of Science
at the Delft University of Technology,
to be defended publicly on June 29, 2021 at 15:30.

Student number: 4701712
4383788
Project duration: April 19, 2021 – June 18, 2021
Thesis committee: Dr.ir. R.C. Hendriks, TU Delft
MSc B. Abdi, TU Delft
Prof.dr. A. Neto, TU Delft

TU Delft

Abstract

The performance of wearable Photoplethysmogram (PPG) sensors is highly influenced by noise. This thesis describes the methods and results of designing a filtering system for PPG in context of Atrial Fibrillation (AF) detection. The developed work is an adaptive filtering system combined with a robust heart rate detection mechanism for validation of the proposed method. Additionally, the heart rate estimation can potentially be used to detect AF episodes using machine learning. Research has been done regarding an optimal reference signal for the adaptive filtering structure. Accelerometer data, being commonly used as reference signal for the noise did not show good correlation with the motion induced artefacts in the signal. Therefore, a reference for the signal component is generated from the PPG itself, which is achieved by applying a narrow bandpass filter. Here the center frequency is determined from an autocorrelation of the signal in a sliding-window. The optimal settings for the sliding window in AF context were found to be 2 seconds with 80% overlap. Furthermore, a comparison is made between NLMS, RLS and Kalman adaptive algorithms, in which RLS showed the best overall performance. The validation of the filtering structure is based on peak detection from the enhanced signal compared with the ECG reference peaks. The results indicates that the system significantly improves the heart rate error in signal disturbed by noise and during AF episodes.

Preface

This thesis is written in context of the Bachelor Graduation Project. The goal of this project is to apply the knowledge and skills obtained during our studies for the BSc Electrical Engineering. In this project we did this by investigating different approaches for recording heart signals, improving its quality through preprocessing steps, and using machine learning algorithms for automatic detection of atrial fibrillation. In the past 2 months we have learned a lot about atrial fibrillation, electrocardiograms, photoplethysmography, adaptive filtering and machine learning classification. The project was a valuable learning experience, because it not only required the application of what we learned during our previous courses like the EPO projects did, but also introduced us to doing research on recent academic literature and setting up a research project ourselves. The experience gained during this project will definitely help us in organising future research assignments.

We would like to thank Bahareh Abdi and Richard Hendriks from the circuits and systems research group at TU Delft for proposing the project and supervising us during the project. The feedback really helped us to focus on what we wanted to achieve during moments we were stuck. Furthermore we would like to thank the following people that helped us during the project: Mathijs van Schie and Fons Wesselius for introducing us to atrial fibiliration and the methods to measure heart signals from their perspective as technical physicians at Erasmus MC; Pourya Omidi from the company Praxa Sense for explaining the challenges and opportunities in developing devices to detect atrial fibrillation; And Eelko Ronner, cardiologist at the Reinier de Graaff hospital, for his insight in how atrial fibrillation is diagnosed and how technology can aid in the diagnosis and monitoring of atrial fibrillation patients.

Kaan Demir
Jasper Konijn
June 18, 2021

Contents

1 Introduction

Atrial Fibiliration (AF) is a common cardiovascular disease, about 1-3% of the general population suffers from a form of AF[1]. It is a progressive disease that is more prevalent amongst elderly than among younger adults. The disease is present in a large part of the general population, but often goes unnoticed until symptoms of damage start showing. During an AF episode, the two upper chambers, the atria, beat irregular and chaotically. This chaotic heart activity can cause damage to the heart and blood vessels in the body, causing blood clots to form, which increases the risk of having a stroke or a heart failure. AF periods are not necessarily persistent but can occur in episodes as short as 30 seconds, until the disease becomes so severe that a patient has continuous episodes and the AF has become persistent [2][1][3].

Currently heart activity is monitored by making an electrocardiogram (ECG) for a period of time to diagnose AF. However, ECG monitoring has some disadvantages that make it hard to detect AF in an early stage. As AF episodes are often not persistent they are difficult to detect on a doctor's visit and patients need to monitor their heart activity at home. Using a Holter device for a period of time is the most common method to monitor the ECG at home. It is a wearable device, but it heavily restricts the user in its day-to-day activities. Due to the inconvenience of wearing the device, it is difficult to use for longer than a few days, which leaves a chance the patient has sporadic AF episodes but no episodes occurred during the monitoring period. The inconvenience and the high cost of the device also make it unsuitable to monitor potential patients that do not show any symptoms. Early stage AF is valuable to diagnose since AF can be treated to prevent the disease to progress to a more severe state. The preventive treatment can prevent or limit the damage to the heart and thereby reduce the chance of more severe health problems.

Photoplethysmography (PPG) is a relatively new technique used to monitor heart activity. The PPG signal shows a pulse waveform that is caused by the heart pumping blood trough the body. The sensors for PPG are non-invasive and can be applied on a small surface on a single location on the body. These advantages over ECG make PPG a popular choice to use in, for example, smartwatches that track the heart rate for sport or general health purposes.

Recent research on using PPG signals to classify AF with machine learning algorithms show promising results, but do not reach the accuracy of algorithms using ECG signals[4][5][6]. Several commercially available devices use PPG to detect irregularities in the heart rhythm and warn their users that AF might be present and advise them to do further checks to verify the condition of their heart. However, the main problem is that the performance of wearable PPG sensors is very susceptible to noise and movement induced artefacts. This significantly limits the reliability of PPG sensors for AF detection. To address this problem a signal enhancement algorithm has been developed with the aim of reducing motion artefacts and other sources of noise in the signal.

1.1. Project description and subgroup division

This thesis is written in the context of the bachelor graduation project. The goal is to design a wearable device for early detection of AF. The device should be affordable, convenient to wear and non-invasive. The proposed solution uses a continuous PPG recording to monitor heart activity. Acceleration data is recorded to provide a reference signal to movements made with the device. The PPG signal is processed to remove noise and motion artefacts. Machine learning based classification is used to predict the presence of AF in the PPG signal. When AF is detected, the user is warned and has the possibility to record a single lead ECG. This ECG is stored to be reviewed by a medical professional.

The project is divided into the following subgroups:

- Signal acquisition
- Signal processing
- Classification

A block diagram of the interactions between these subgroups can be seen in Figure 1.1. The objectives of the signal acquisition subgroup are to acquire continuous PPG and accelerometer recordings in a digital format. An ECG recording should be requested when an AF episode is detected from the PPG monitoring. The signal processing subgroup is responsible for filtering and processing of these digital signals. Here the PPG signal is enhanced by reducing noise and motion artefacts to output to the classification subgroup. Furthermore, an estimate of the beat-to-beat heart rate is extracted. The objective of the classification subgroup is to extract features from the filtered PPG signal and the accompanying beat-to-beat heart rate. A classification algorithm is designed that uses these signal features (e.g. heart rate variability) to detect AF episodes in the PPG recordings.

This subdivision of the project has a minimal interdependence of the subgroups, which is favourable for a successful proof of concept of each subgroup. The signal acquisition is a logical subgroup as there is a clear necessity for specific research on where and how to measure the heart related signals from the body. This includes the design choices from the analog front end up to the analog-to-digital conversion. Next, the signal is processed on the device itself as seen in Figure 1.1. The need for a processing subgroup was also evident, since the PPG signal can be severely distorted by noise, rendering obsolete the classification of this data. The design choice for a classification subgroup is justified by having automated AF detection, which limits the amount of tedious work for clinical experts. Alternatively, a subgroup for wireless transmission of the data could be chosen, in which case further signal processing could take place on a PC or a cloud based application.

1.2. Thesis outline

This thesis describes the work done by the signal processing subgroup. In chapter 2 the program of requirements describes the requirements for the proposed wearable AF detection device and the specific requirements for the signal processing subgroup. Chapter 3 gives background information about the working of the heart and atrial fibrillation, photoplethysmogram signals and electrocardiogram signals. Chapter 4 describes the design process for a noise reduction algorithm. It is based on using a fundamental frequency estimator to acquire a reference signal for adaptive filtering, and subsequently peak detection to retrieve the peaks in the PPG signal under noisy conditions. Chapter 5 compares the accuracy of the peak detection after filtering for different adaptive filtering algorithms. The accuracy is determined by comparing peak locations from detected peaks in the PPG signal for NSR and AF segments to detected peaks in the reference ECG signal in those segments. Chapter 6 discusses the results of the peak detection comparison and discusses the usability of the algorithm for AF classification in a wearable device. Chapter 7 concludes on the usability of the algorithm by summarizing the validation method and the results and compares this to the requirements for the device.

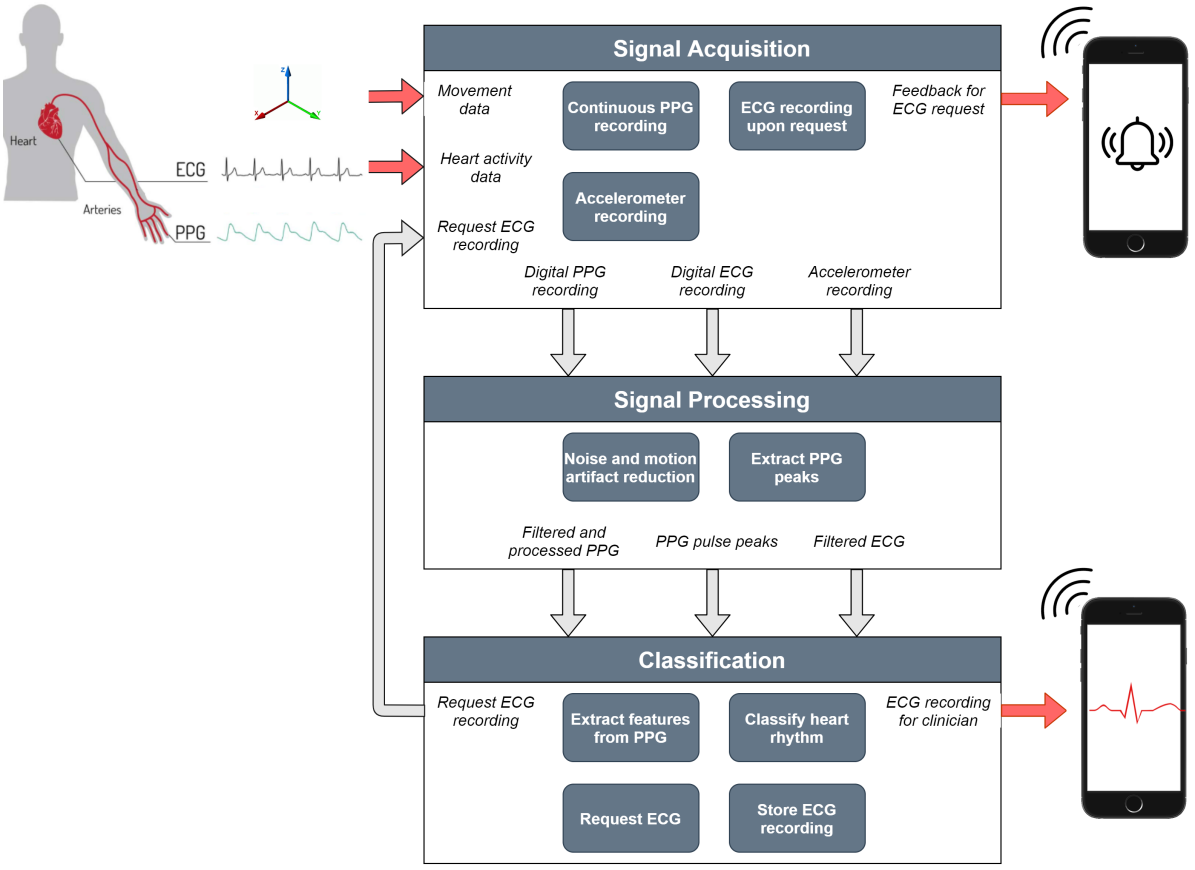

Figure 1.1: Overview of the complete project, the subgroups and their interdependencies. The proposed system runs on a wearable device (e.g. worn on the wrist), and communicates wirelessly with a smartphone or PC

2

Program of Requirements

This chapter contains the requirements for the project. The requirements for a prototype of a wearable AF detection device as a complete system are listed. The necessity of a requirement is explained in the accompanying paragraphs. The objectives for the signal processing part of the project are listed afterwards. The requirements result from interviews with professionals in the field of cardiology and biomedical engineering and a review of current literature on research into atrial fibrillation, photoplethysmograms and electrocardiograms.

2.1. Requirements for the complete system

2.1.1. Functional requirements

1. **The system must measure the user's heart activity:**

   - It continuously measures PPG to make 30 second long PPG recordings.
   - It is able to make a single lead 30 seconds ECG recording on request by the user.

   The system must measure the user's heart activity to acquire information to detect atrial fibrillation. PPG signals are used in the classifying algorithm, the ECG signals are provided to a medical professional. Medical professionals are not trained to analyse PPG recordings for arrhythmia. The ECG can help a professional decide if an AF detection by the device warrants further medical intervention. Recordings must be of at least 30 seconds since AF episodes are defined to be at least of that length.

2. **The system must be able to detect AF episodes:**

   - Detect AF episodes using machine learning based classification algorithm.
   - The classification must have an accuracy of at least +0.8 Correlation Coefficient score.
   - Type II (False Negatives) are more important to minimize, reducing the number of times an AF episode is missed.
   - Detected episodes last at least 30 seconds, shorter episodes do not need to be detected.

   The requirement to use machine learning results from the aim of the graduation project to have automated AF detection. The accuracy requirement is chosen to be strict enough that false positives do not overburden the user and the medical system.

3. **The system must be able to output a report that is interpretable by a medical professional in order to aid in the diagnosis of Atrial Fibrillation:**

   - Provide the time of the AF episode with visual ECG recording corresponding to that AF period.

   When a user is warned about the possibility of having AF the help of a medical professional is needed for a complete diagnosis and treatment matching with the diagnosis. When a user can only present the warning given by the system it does not provide any information to the medical professional to

warrant further investigation, the professional has to blindly trust the detection system. When more people start using a system for early detection the medical system can be overburdened with request to diagnose AF. In this case it would be helpful if the device provides evidence on when it detected AF and show a corresponding ECG for that period. This can be reviewed by an expert to decide if further diagnoses is required or if it is a false alarm.

4. **The system must be able to give feedback to the user if an AF episode is detected**
   The aim of the prototype is to automatically detect AF based on continuous PPG recordings, when an AF episode is detected the user should receive a warning and the advice to contact a medical professional. After a user is warned on the possibility of AF it also has the option to make a manual ECG recording with the device. To know when to make an ECG recording the user needs to receive feedback in a fast and urgent way.

5. **The system must be able to be used during day-to-day activities while maintaining its accuracy:**

   - Noise and artefacts due to motion should be reduced in the recorded signals.
   - Segments that are too noisy to be recovered should be rejected.
   - Segments that are enhanced should correctly detect 85% of the peaks.

   During activities where the system and its sensors move, the recorded signals are prone to be corrupted with motion artefacts, these artefacts can have an effect on the extracted features used in machine learning and result in misclassification of NSR as AF.

2.1.2. Non-functional requirements

1. **The device must be wearable as an accessory:**

   - The used technology should be implementable in a device with a maximum area size of 20 cm^2 and a height of 1 cm.
   - The device should use a battery that lasts at least 12 hours on one charging cycle.
   - The device should be rechargeable.

   An accessory like a watch or jewelry is a convenient form for a detection system so that it can be used over long periods or even permanently if the user should choose so.

2. **The system must only provide information and advice but never a diagnosis**
   The system cannot provide a diagnosis since it will not undergo clinical testing and machine learning based on PPG signals is not an approved method of diagnosing AF.

3. **The system must be affordable to the general public, therefore the cost of hardware/development must be no more than €30,- per device**
   The system is designed to provide a way to detect the presence of AF in the early stages. The burden to use the device on individuals that not necessarily have any symptoms or risk factors for AF should be as low as possible. Using the device in a general population can maximize the prevention of damage from AF in later stages. The device should therefore be cheap and commercially available.

2.2. Objectives for the signal processing subgroup

The signal processing subgroup focuses on the requirements to enable the system to be used during daily activity. From the problem description and the requirements the choice was made to use continuous PPG recordings as input signal for the classification algorithm. Activities cause the PPG signal to be corrupted with noise and artefacts. Using corrupted data can reduce the accuracy of classification algorithm. When features extracted from noisy data with NSR correspond with features from noise free data with AF, this will lead to segments incorrectly being classified as positive for AF.

The PPG signal needs to be enhanced to reduce the effects of noise on extraction of features. The intervals between peaks in the PPG signal are used to determine the heart rate and heart rate variability statistics. These are important features in classification of AF.

It is possible that the incoming signal does not represent a true PPG signal anymore due to the amount of noise or lack of signal present. This is possible when the sensor does not make proper contact with the skin.

These segment should be excluded from classification since the algorithm can not properly extract features when the incoming signal does not represent a PPG signal.

The PPG signal cannot be deformed too much by the filtering process since this can result in the possibility of features not properly extracting from the signal. Listing requirements for PPG signal quality has proven difficult since there is not a standardized signal quality index. There is no measure of determining what is too much deformation by analysing the signal. The quality of the output signal can be validated by running the classification process and comparing its accuracy for different output signals.

The acquired ECG signal is only used as an output for a medical professional. Processing of the this signal will be limited to bandpass filtering. Motion artefacts in the ECG signal can be manually recognized by the medical professional and ignored where applicable.

3

Background

In this chapter background information is provided on atrial fibrillation, which is the heart condition this project is addressing. The main methods for monitoring heart activity are electrocardiogram (ECG) and Photoplethysmogram (PPG). The basics of PPG and ECG signals are described to come to a better understanding on how to detect AF. To this end a commonly used method of classification using machine learning algorithms is introduced. Finally, the dataset used for this thesis is described to provide context for the developed methods.

3.1. Atrial Fibiliration (AF)

The heart consists of four chambers, the upper two chambers of the heart are called the atria and the lower two chambers of the heart are called the ventricles. Figure 3.1 shows the chambers in the heart and the electrical nodes. The atria collect blood from the blood circulation when they are relaxed and pump this into the ventricles when they contract. When the ventricles contract, they pump the blood through the body causing circulation back to the atria. The sinoatrial (SA) node is a group of cells in the right atrium that generate an electrical impulse that causes the atria to contract. The electrical impulse travels from the SA node along the right atrium through the atrioventricular (AV) node and also causes the ventricles to contract. The AV node is a junction in the electrical pathway that connects the atria with the ventricles. The AV node delays the electrical signal to prevent the ventricles to contract before they are filled with blood by the atria. In a healthy heart this process results in a regular heartbeat also called a normal sinus rhythm (NSR).

During an AF episode the atria receive chaotic electrical signals that disrupt the regular impulses from the SA node, this causes the atria to beat chaotically and irregular. During AF the AV node also receives the chaotic electrical impulses, however not all impulses are conducted through the node. The ventricles contract at irregular and higher than normal rate, but not as fast as the atria since not all impulse pass the AV node. This results in the ventricles contracting with a different rhythm than the atria.

AF does not cause an acute health problem. In early stages of the disease a patient does not notice any symptoms or it notices heart palpitations due to the ventricles beating irregularly. However, AF episodes can cause damage to the heart and blood vessels. The damage increases the severity of future episodes and blood clots are more likely to form. Due to the damage done by AF, patients have an increased risk for strokes and heart failure.

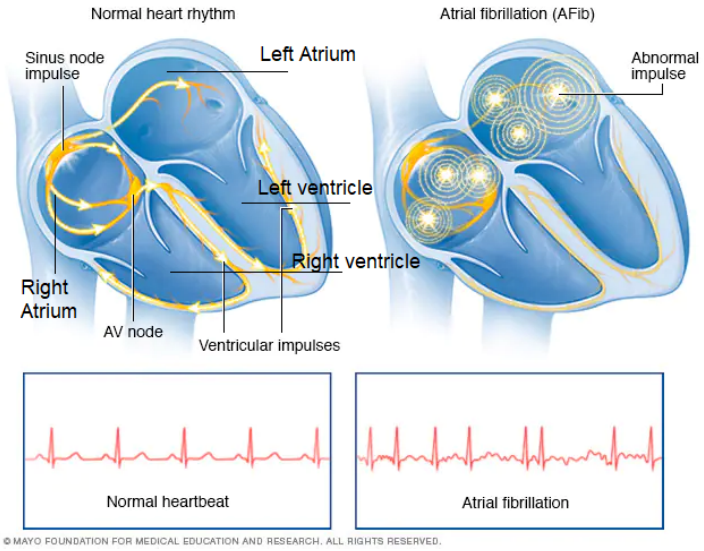

Figure 3.1: Image of the heart and ECG waveform for NSR and AF. Image adapted from [7].

3.2. Photoplethysmogram (PPG)

A Photoplethysmogram (PPG) is a non-invasive technique that measures the blood flow at the surface of the skin. As the heart pumps blood to the periphery, a change in blood volume occurs. This is measured by illuminating the skin with the light (most often infrared or green light) of an LED and then measuring the amount of light that is either transmitted or reflected to a photo diode. The PPG signal that is measured comprises of signal components from the absorption of skin, bones and tissue, and that of the blood flow itself. A visualization of this can be seen in Figure 3.2. Here a single pulse in the clean PPG signal consists of a primary peak (systolic peak) and a secondary peak (diastolic peak). Common applications of PPG are the calculations of the arterial oxygen saturation and heart rate, found for example in medical pulse oximeters. Other upcoming applications are smartwatches that incorporate PPG for health monitoring during day-to-day activities. In these devices the heart rate is being monitored during daily activities where motion is present. However, modern wearable devices are also capable of measuring features like blood pressure, heart rate variability and other health-related information [8]. This is especially interesting in early detection of Atrial Fibrillation, where heart rate variability serves as a key indicator. For a very long time ECG has been used as the dominant method for detecting cardiovascular abnormalities and arrhythmia. However, PPG shows to be a potential alternative due to its' great wearing comfort, portability and cost effectiveness.

3.2.1. Noise sources and filtering

One of the major difficulties is that a PPG signal is highly susceptible to noise. Motion artefacts pose the most problematic source of noise as this can lead to an inaccurate interpretation of the PPG signal. For these reasons an algorithm for enhancing the signal quality in presence of noise is of great importance. Reduction of motion artefacts is commonly done by using adaptive noise cancellation. These are adaptive filtering methods that involve an FIR filter structure with an adaptive algorithm for tuning the filter coefficients to converge to a minimal error. Adaptive algorithms include Least Mean Squares (LMS), Normalized Least Mean Squares (NLMS), Recursive Least Squares (RLS) and state estimators like a Kalman Filter [9]. Accelerometer data is often taken as a noise reference signal to use in adaptive noise cancellation [10]. The drawback of using accelerometer data is the additional hardware and computational complexity. Additionally, accelerometer data reflects the motion present in the signal, as opposed to motion induced noise. Other methods include a synthetic noise reference by means of Singular Value Decomposition (SVD), Fast Fourier Transformation (FFT) and Independant Component Analysis (ICA) [11]. Another solution focuses on heart rate estimation by means of a peak detection algorithm without the use of additional hardware [12]. This method extracts heart rate variability using a peak detection algorithm that reliable extracts peaks from a PPG signal that is corrupted by noise and motion artefacts. The peak detection is based on computing the autocorrelation on semi-periodic segments.

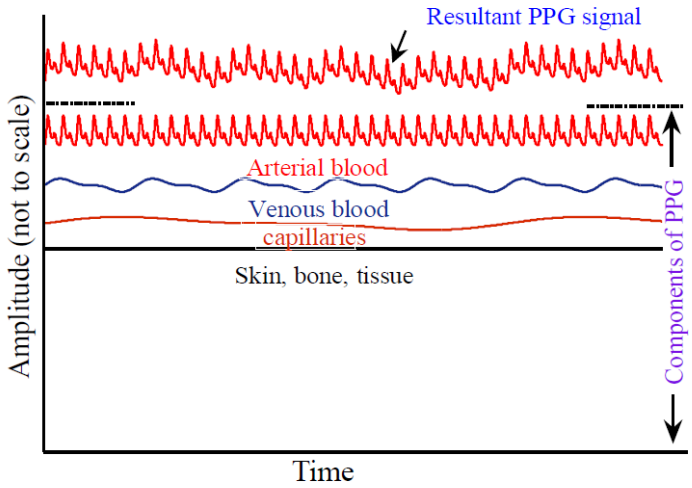

Figure 3.2: Components of a typical PPG signal [13].

3.3. Electrocardiogram (ECG)

The electrocardiogram (ECG) is an important measurement in the process of diagnosis of cardiovascular health problems. A standard 12-lead ECG records potential differences between specified body surfaces, which reflect the electrical activity of the heart muscles. In the ECG measurements of a normal sinus rhythm several segments are of interest. These segments are included in the PQRST complex as shown in Figure 3.3. The P wave is the electrical signal corresponding to the contraction of the atrium. The QRS complex corresponds to the contraction of the ventricles. The T wave corresponds to the relaxation of the ventricles. Deviations in the PQRST complex can be indications of arrhythmia or other cardiovascular problems. AF is visible in an ECG as chaotic high-frequency electrical activity in the atrial waveform and an absence of the P wave. Due to the chaotic rhythm of the atrium, the ventricular rate is also irregular. This irregular rhythm can be observed in the interval between QRS complexes [14]. Accurate detection of atrial activity and P waves can be difficult due to noise in the signal. Therefore, using only information on the QRS complex can cause other arrhythmia to be misclassified as AF.

In [16] information is given about how a medical standard ECG is performed and recommendations are given on what digital signal processing is required. The ECG signal contains relevant information in a frequency band from 0.67 Hz to 150 Hz for adults. The fundamental frequency of the QRS complex is 10 Hz and most diagnostic information is below 100 Hz, but can be as high as 500 Hz. A frequency of 0.67 Hz corresponds to a heart rate of 40 BPM, lower heart rates are possible but are uncommon.

3.3.1. Noise sources and filtering

Multiple forms of artefacts present as noise in the ECG measurement are mentioned in [17]. This includes internal and external electrical interference by devices in proximity of the ECG sensors, motion artefacts caused by movements or muscle activity and baseline wandering of the ECG signal. To obtain a noise free ECG signal power line interference, base line wandering and motion artefacts need to be removed from the sensor signal. This is done by either analog or digital filtering.

To remove the base line wandering, a multitude of filtering methods are used. A common way to remove base line wandering is high-pass filtering with cutoff frequency of 0.7 Hz [18]. Some other techniques include adaptive filters, notch filters, LMS algorithm, wavelet transform, projection operator-based approach, empirical mode decomposition (EMD) methods, robust sparse signal decomposition, cubic spline method. [18] Compares the Fourier decomposition method (FDM) to the widely used empirical mode decomposition (EMD) for both base line wandering and power line interference removal.

Motion artefacts are caused by movements the body. These movements can result in movement of the sensor electrodes, which causes change in the measurements. Movements also cause a change in the skin impedance due to deformation of the skin, which changes the signal. The electrodes that are used to measure the heart signal can also measure electrical activity in the muscles. Muscle contractions that occur during movement result in a noise signal in the measurement. Independent Component Analysis (ICA), Adaptive

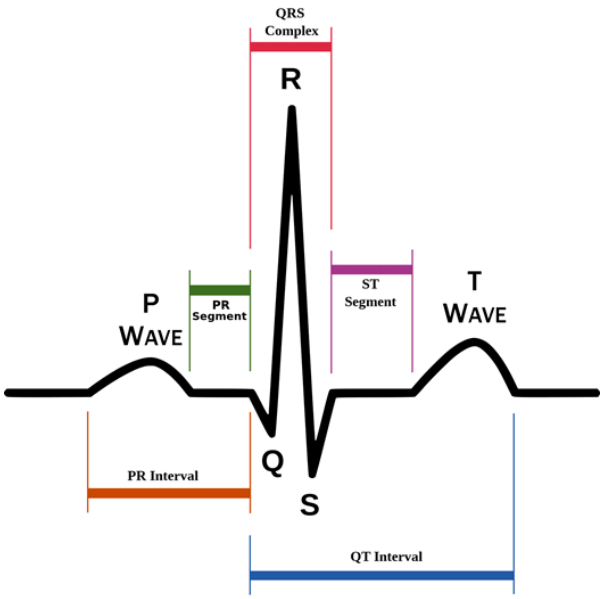

Figure 3.3: PQRST complex in an ECG signal of a NSR [15]

Filtering, Singular Value Decomposition (SVD) and Tensor Decomposition are reviewed in [19] as options to reduce the effects of motion artefacts in ECG signals.

ICA separates two or more independent source signals. In the case of ECG: the noise free ECG signal and the motion artefacts. These signals are detected by two independent detectors which output a noisy ECG signal and a motion signal. All signals are assumed to be realizations of random variables with their own probability density. For the working of ICA it is assumed that the output signals are a linear combination of the input signals.

Adaptive filtering is a suitable technique to achieve reduction of motion artefacts. This technique requires a reference signal that is correlated with the noise signal that is estimated by the filter. Skin impedance measurements are a suitable reference signal as shown in [20][21][22]. Another option is to use three axis accelerometer signals as a reference signal correlating with the motion artefacts. Ottenbacher et al [22] compare adaptive filtering using a skin impedance reference signal to similar filters using a accelerometer reference signal. Both are suitable to improve QRS detection when motion artefacts are present. Combining the two might result in further improvements of the QRS detection

SVD separates features into independent feature spaces. This can be used to separate features that contain information on the motion artefacts from features that contain information on the noise free ECG signal. The SVD gives no information on how to interpret the independent features. Domain knowledge is required to recognize the patterns of the PQRST complex in the features and patterns of motion artefacts. This makes SVD less suitable for noise removal than adaptive filtering.

Tensor decomposition is a higher dimensional decomposition technique. Tensors can be uniquely decomposed under less strict conditions than ICA and SVD, making it suitable when limited different measurements are available. Dargie and Lilienthal [19] applied tensor decomposition, to extract MA features from ECG recordings. In [23][24][25][26] tensor decomposition is used to separate features containing atrial activity. This is important for AF classification, since the absence of P waves is an indicator of AF. A drawback of BTD is that proper initialisation of all parameters, needed for the tensor decomposition, is difficult. Tensor decomposition also requires multiple lead ECG measurements to construct the tensor, similar to how the matrix is constructed for SVD.

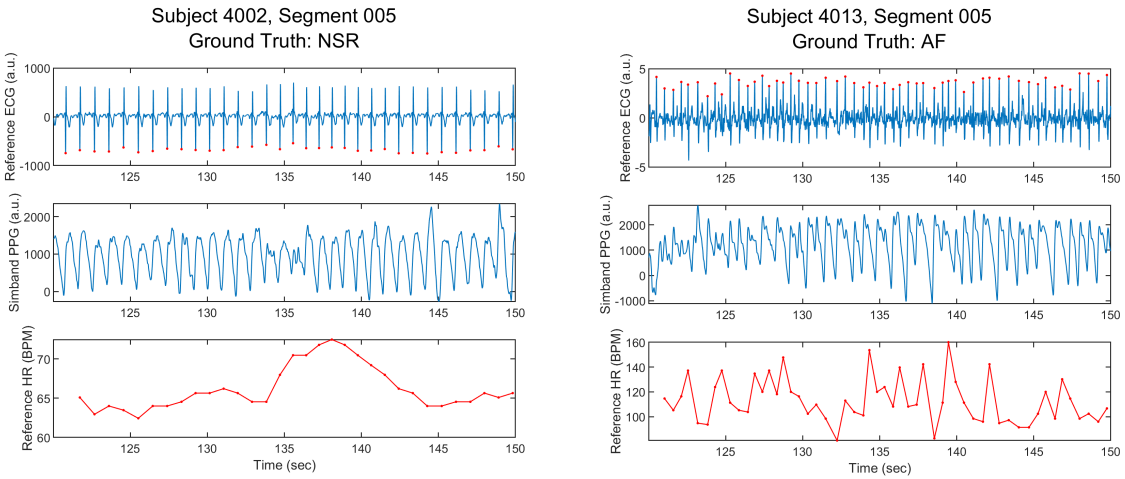

Figure 3.4: 30 second segments of PPG and ECG signals. Detected peaks in the ECG signal are marked with a red dot. The left figure shows a subject with NSR. The right figure shows a subject with AF. Peaks in the NSR segment have a regular interval and the heartrate is only changing slightly beat to beat. Peaks in the AF segment have an irregular interval, the heart rate in the AF segment is also higher than usual therefore peaks in both the ECG and PPG signal lie close together.

3.4. Machine learning for AF classification

Classical machine learning classification algorithms require feature extraction on input signals. The extracted features contain information that can separate the different classes in the set of recordings. A machine learning algorithm optimizes the way it interprets the information from the features to separate classes. In our case the classes that we are interested in are AF segments and NSR segments. The heart rate of a subject and the heart rate variability (HRV) are important features that separate a heart signal with NSR from a heart signal with AF. The heart rate and HRV are derived from the interval between peaks. For ECG signals the intervals between R peaks in the QRS complex are used and for PPG the intervals between the pulse peaks in the signal are used. AF patients have an irregular heart rate and the HRV measures differ from a NSR. Figure 3.4 plots both PPG and ECG signals during a 30 second segment of data and the heart rate derived from the ECG peaks. The left plot shows a subject with AF and the right plot shows a subject with NSR. In the plots of both the ECG and PPG signal it is noticeable that the interval between peaks is very irregular for the AF segment and the derived heart rate is both higher and rapidly changing for the segment with AF compared to the NSR segment. The accuracy of peak detection is very important to properly separate AF from NSR. Recordings for NSR can be misclassified as AF when the interval between peaks is irregular due to incorrect detection of peaks in a noisy signal. In chapter 4 a filtering approach is proposed that enables peak detection in segments where the PPG signal is distorted due to noise.

For AF classification not only the peak intervals are interesting, but also other features can be extracted from the PPG signal. Features relying on the shape of the waveform can only be extracted from good quality PPG signals, without distortion from noise and artefacts and without deformation due to aggressive filtering. Elgendi [27] separates PPG signal waveforms in three groups: excellent for diagnosis (G1), where systolic and diastolic waves are both visible, acceptable for diagnosis (G2), where the systolic and diastolic waves can not be recognized but heart rate tracking is possible, and unfit for diagnosis (G3), where the systolic and diastolic waves can not be recognized and heart rate tracking is impossible. Figure 3.5 shows example waveforms for the three categories. For PPG there is not a widely used signal quality index (SQI) that indicates whether a signal is of good quality for heart rate tracking and AF detection. The perfusion Index (PI) is used as SQI to determine the suitability of PPG recordings when PPG is used for derivation of oxygen saturation, but is not necessarily useful for heart rate tracking. In [27] the skewness of the PPG signal outperformed other SQIs in its ability to separate excellent recordings from acceptable and unfit one.

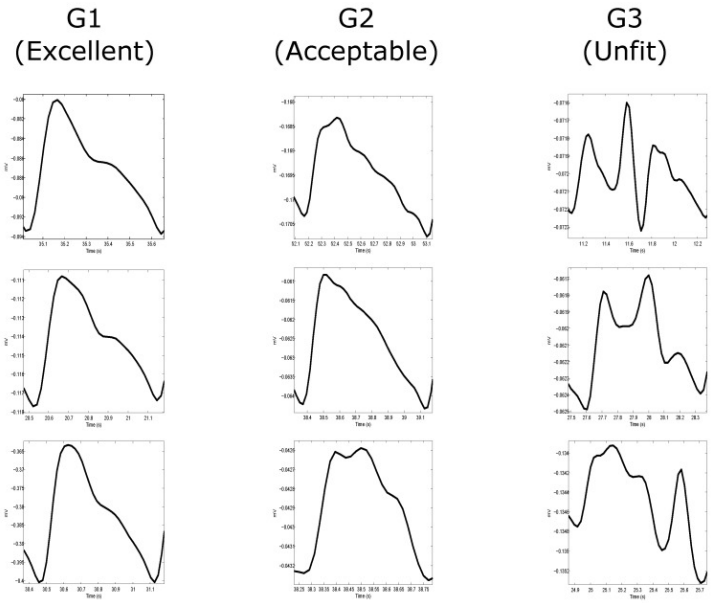

Figure 3.5: Examples of excellent(G1), acceptable(G2) and unfit(G3) PPG signal waveforms. Signals in G1 show both the systolic and diastolic wave. Signals in G2 do not have a clear separate systolic and diastolic wave, but heart rate tracking using pulse peaks is possible. Signals in G3 do not have a clear separate systolic and diastolic wave and heart rate tracking is impossible [27] .

3.5. UMMC Simband dataset

To aid the design of the AF detection wearable prototype we make use of the UMMC Simband dataset. This dataset was created by Ding et al. [28] with the objective to test the performance of an AF detection algorithm using pulse data from a smartwatch among an elderly population. This dataset is useful to us since it contains multiple PPG signals with different light wavelengths, an ECG reference signal and an acceleration signal. The experiment includes conditions and activities that resemble daily use of a wearable device and the dataset is annotated with labels for AF and NSR for short segments.

Dataset description

The dataset consist of 40 subjects. The subjects wore the Samsung Simband 2 and a 7-lead Holter monitor to record their heart activity. The Samsung Simband records PPG with 8-channel sensor layout. The different channels use light with different wavelengths. The Samsung Simband also records acceleration with a three axis accelerometer. The Samsung Simband can also make a single lead ECG recording when the user touches the electrode on the watch with the opposing arm, thereby closing an electrical loop around the heart. The ECG recordings of the Simband were not used in the experiment. The Holter device was used to record 3-lead ECG recordings. The ECG recordings were divided into 30 second segments and analysed with a highly accurate AF detection algorithm[29] to indicate if AF was present in the 30 second recording. About 10% of the recordings containing NSR and 10% of the recordings containing AF were blindly, without knowledge of the algorithms results, reviewed by a board-certified cardiac electrophysiologist. The findings of the cardiac electrophysiologist were 100% in agreement with the algorithm results[28]. Nine of the subjects have segments containing AF, 6 of the subjects have segments containing premature atrial contractions(PAC) or premature ventricular contractions (PVC).

Experiment protocol

The subjects performed several actions while wearing the recording devices. The actions were chosen to mimic daily activities. The recordings contain the following movement protocol: sitting still for 2 minutes, slowly walking for 2 minutes, about 2 miles per hour, 30 seconds of standing, 2 minutes of quickly walking, about 4 miles per hour, standing for 1 minute, 1 minute of of vertical movements with the watch wearing arm, 1 minute of wrist movements with the watch wearing arm, 30 seconds standing still, 1 minute of repeatedly sitting down and standing up from a chair, climbing stairs for 2 minutes, 1 minute sitting down performing breathing exercises.

Signals used for further processing

The loading code provided by [30],[31] loads a single lead from the 3 lead ECG, green light with a wavelength of 520-535nm, from the Samsung Simband PPG sensor, and the average magnitude of the 3 axis accelerom-

eter(ACC) calculated from the individual axes, a_x, a_y, a_z by equation: 3.1. The ECG signal in the dataset is down-sampled to 128 Hz, the PPG signal to 50 Hz and the ACC signal to 30 Hz. The signals are divided into 30 second segments by the loading code. In addition to the signals the labels of the 30 second segments are provided and the extracted locations of the R peaks in the ECG signal. The segment labels contain the following labels: 0 for NSR, 1 for AF, 2 for PAC/PVC, 3 for possible PAC/PVC but considered as noisy, 5 for PPG too noisy and NaN in case the reference ECG was incomplete.

$$ACC = \sqrt{a_x^2 + a_y^2 + a_z^2} \tag{3.1}$$

4

Design of signal enhancement algorithm

The performance of wearable PPG sensors is highly influenced by noise and movement artefacts. This substantially affects the usefulness of PPG sensors when a subject is moving. In this chapter a method for reducing these noise and artefact sources is described, thereby enhancing the signal. First the correlation is investigated between the accelerometer data and motion induced noise, to determine the need for accelerometer data. Thereafter, an adaptive filtering structure design is proposed in which a reference signal is generated from the PPG signal itself. The full system can be seen as a block diagram in Figure 4.7 and is implemented using several functions written in Matlab as seen in Appendix (A.2.2).

4.1. Analysis of PPG signals in relation to motion

Acceleration is considered as a suitable reference signal in adaptive filtering applications to reduce motion induced noise[32],[33]. In this section we investigate how the PPG signal is correlated with the acceleration signal. To further investigate the signals the segments of the PPG signal are first bandpass filtered with a 6th order Butterworth filter with cut-off frequencies of 0.5 Hz and 20 Hz. The filtered PPG signal and de ACC signal is then normalized to zero mean and unit variance.

Looking at the acceleration signal it is clear that the actions from the experiment protocol result in a high magnitude acceleration signal, and an almost zero magnitude signal when the subject is standing or sitting still. Figure 4.1 contains 30 second segments of this subject when it is sitting still, walking fast and during the random wrist movements. A simple peak detection algorithm based on the work in [34] is used to indicate the peaks in the PPG signal. The algorithm finds the indices of the maxima in the regions where the signal is above the moving average. Maxima that are within a threshold from previous maximum are ignored. When the subject is sitting still, the ACC signal is low and the maximum amplitude in a PPG pulse is mostly constant, when the subject is walking the peak magnitudes of the ACC signal are larger and the PPG signal is somewhat distorted compared to an ideal PPG waveform. The peak detection algorithm detects all peaks in this segment but also detects some extra peaks. When the subject is moving it's wrist the peak magnitudes of the ACC signal are even larger and the PPG signal is further distorted. The peak detection misses some peaks and also detects some extra peaks.

In previous work by Bashar et al. [30] all segments containing ACC signals above a threshold are rejected for AF classification. This leaves many of the recordings during day-to-day activities unusable, only recordings where the subject is sitting still are usable for AF classification with this approach. Even segments where the subject is standing still can contain too much distortion even if the acceleration is low. Figure 4.2 shows a segment from subject 2 where it is standing still after having done a fast walk. This segment was considered too noisy to classify by the variable frequency complex demodulation (VFCDM) based noise detection from [30]. In the figure it is visible that not all peaks are detected correctly by the simple peak detection algorithm.

4.1.1. Coherence between acceleration and PPG signals

To examine the relation between the acceleration and PPG signals, wavelet coherence can be used. Wavelet coherence is a measure of the correlation of two signals over different frequencies in the time-frequency plane. Figure 4.3 shows the coherence plot for the acceleration signal and PPG signal of subject 2. The red boxes mark the periods where there is high acceleration due to the activities in the experiment. There is

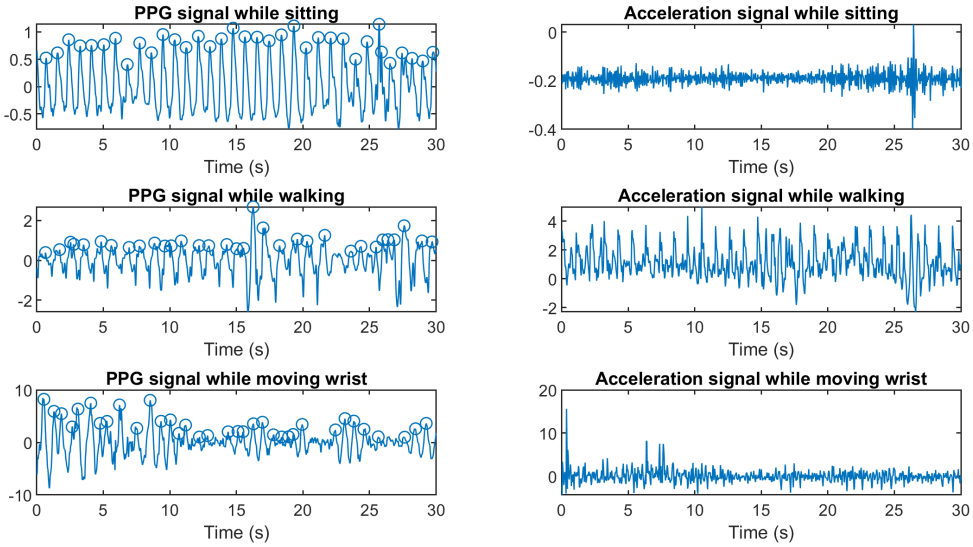

Figure 4.1: 30 second segments of PPG and ACC signals of subject 2. Detected peaks in the PPG signal are marked with a circle. The top 2 plots are from a segment where the subject is sitting still, the middle 2 plots are from a segment where the subject is walking fast, the bottom 2 plots are from a segment where the subject is moving it's wrists.

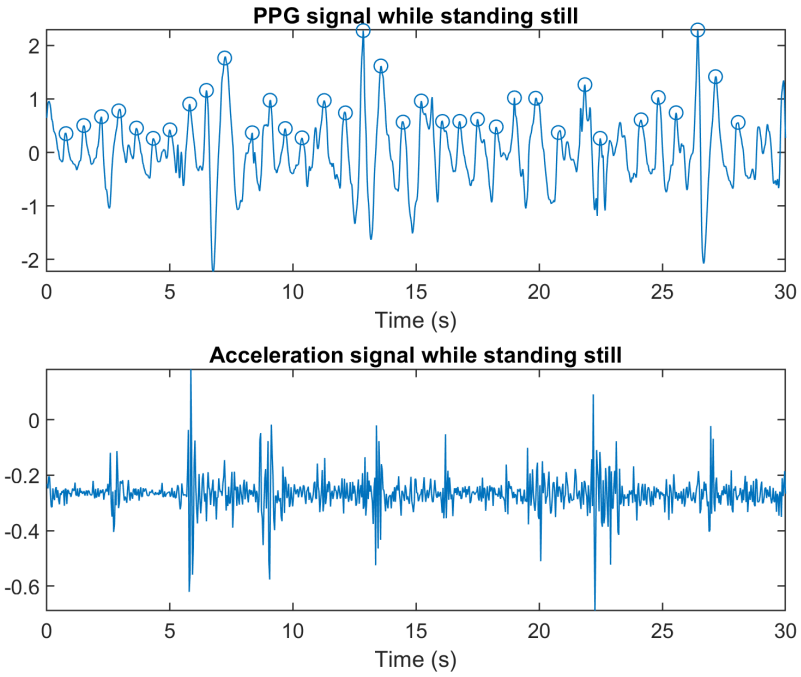

Figure 4.2: 30 second segment of PPG and ACC signals of subject 2. Detected peaks in the PPG signal are marked with a circle. The plots contain a segment where the subject is standing still for a minute after walking fast for 2 minutes.

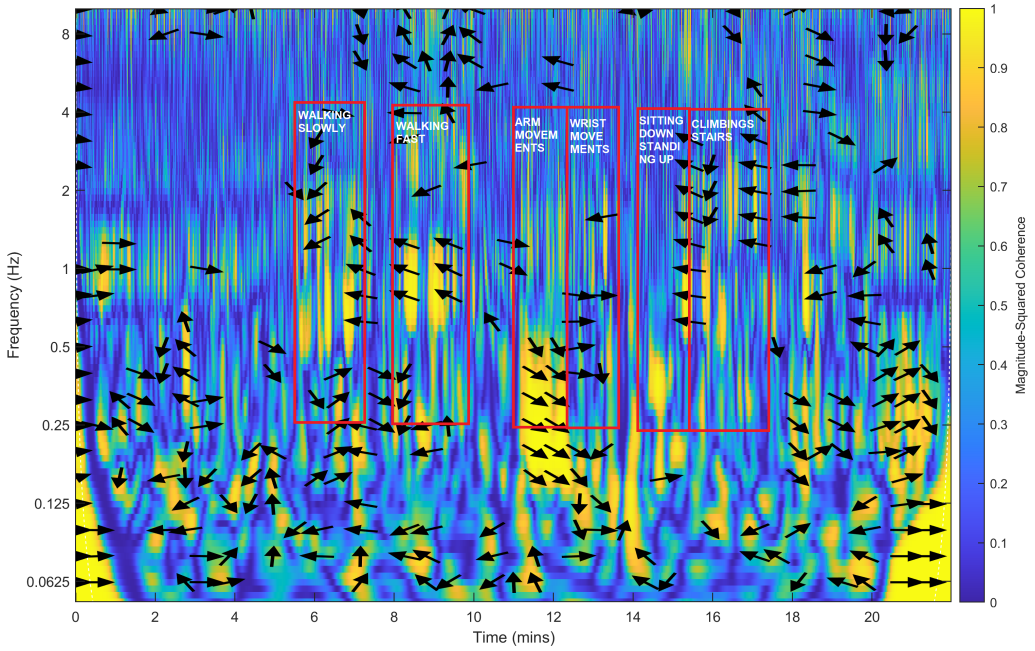

Figure 4.3: The wavelet coherence of the PPG and ACC signals of subject 2. The red boxes encompass the periods the acceleration signal has a high magnitude due to actions in the experiment. In these segments the wavelet coherence is the most interesting since the movement from the actions lead to low quality PPG.

more coherence inside these periods than elsewhere in the plot, but the coherence still varies a lot. The acceleration in these periods is fairly constant in average magnitude, but the correlation with the PPG signal is variable. Figure 4.4 shows how often coherence occurs in a defined frequency band. This figure plots the amount of times a coherence higher than 0.7 is present for longer than 5 seconds in 0.5 Hz frequency bins from 0.5 Hz to 8 Hz in the left bar chart. The right bar chart further zooms in on the low frequencies that contained coherence more often than the higher frequencies. It shows the amount of times a high coherence occurs for 0.1 Hz bins from 0.3 Hz to 1.8 Hz. Only the low frequency components between 0.3 and 1.0 Hz show coherence that could correspond to motion induced artefacts.

4.1.2. Acceleration as a reference signal in adaptive filtering

To create a reference signal from the acceleration, the average magnitude calculated with equation 3.1 of the three-axis accelerometer is bandpass filtered with the same filter as used for the pre-processing of the PPG signal in 4.2. This signal can be used as the reference signal r(n) in the adaptive filter structure. However, using the filtering structure with this reference signal did not lead to satisfactory results. The amplitude of the filtered signal was influenced in an unpredictable manner and prevented peak detection to find the correct maxima. This unpredictable result can be explained by the limited correlation between the PPG and ACC signals. From the analysis of the wavelet coherence we found that the signals were only correlated for short periods of time and for limited frequencies. To obtain a useful reference signal from the acceleration more manipulation would be required to only use correlated parts of the signal, but this not further investigated in this thesis.

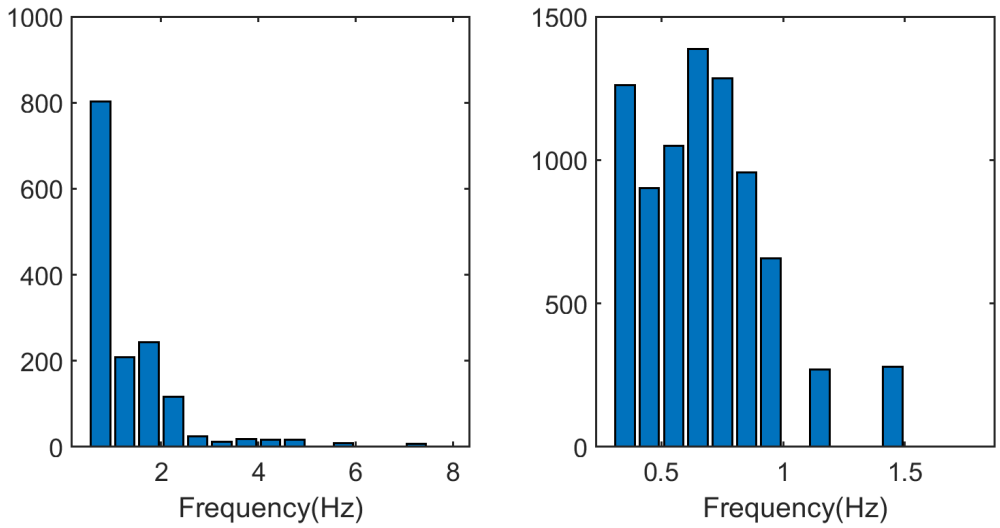

Figure 4.4: This figure shows the occurrence of high coherence between the ACC and PPG signal. In the left bar graph segments when the average coherence for the frequency bins is higher than 0.7 for a period of 5 seconds are summed and displayed for frequency bins from 0.5 - 8 Hz with a bin size of 0.5 Hz. In the right bar graph segments when the average coherence for the frequency bins is higher than 0.7 for a period of 5 seconds are summed and displayed for frequency bins from 0.3 - 1.8 Hz with a bin size of 0.1 Hz.

4.2. Pre-processing

The raw PPG signal that is measured is first passed through a band-pass filter to remove baseline wandering and high frequency noise like power line interference. The useful information in the signal is roughly contained in the range of: 0.4 to 5 Hz [35], which fixes the lower bound of the heart rate to 24 BPM. The specifications for the corresponding bandpass filter are given below:

- Stop band frequencies of:F_{stop1} = 0.1 Hz and F_{stop2} = 5.0 Hz.
- Cut-off frequencies of F_{c1} = 0.4 Hz and F_{c2} = 4.7 Hz.
- Stop band attenuation of 60 dB.

For the design of the bandpass filter an equiripple filter design is chosen, because it results in a shorter FIR filter when compared to windowing structures like Chebyshev and Hamming. The resulting filter order is 329, which is the minimum order for satisfying the filter specifications. In Figure 4.5 it can be verified that the designed filter meets the listed requirements. In Figure 4.6 a plot is given as an example and visualization of the PPG signal plus the effect of bandpass filtering it. This signal is considered to be clean, meaning that there are no motion artefacts or other significant noise sources present except for high frequency noise.

4.3. Motion artefact reduction

Apart from high frequency noise, the signal quality can be influenced by motion artefacts. In this section the algorithm for reducing motion artefacts is proposed. A block diagram of the basic structure of the algorithm is shown in Figure 4.7. The first step in processing the signal is the bandpass filter that is already discussed. Next a reference signal is used for the adaptive filter, which is generated from the bandpass filtered PPG signal. This signal should be correlated with the PPG signal without motion artefacts. The first step is estimating the heart rate as the fundamental frequency. This is done by analysing the autocorrelation of the PPG signal. The fundamental frequency is then used for a more narrow bandpass filtering, from which only specific periodic information of the signal remains. This is used in the adaptive filtering as a reference signal to filter towards. The output of the filter, which is being fed back to the adaptive algorithm, is the enhanced PPG signal.

The signal is processed using a sliding window approach of 2 seconds wide and 0.4 seconds increment per window. These values are found as optimal for finding the fundamental frequency. A too narrow time window leads to not enough periodic information. On the other hand, a too wide time window leads to an inaccurate fundamental frequency as the heart rate varies across time. Having an accurate fundamental frequency is key to good adaptive filtering, especially since the goal is to detect AF, where distances between

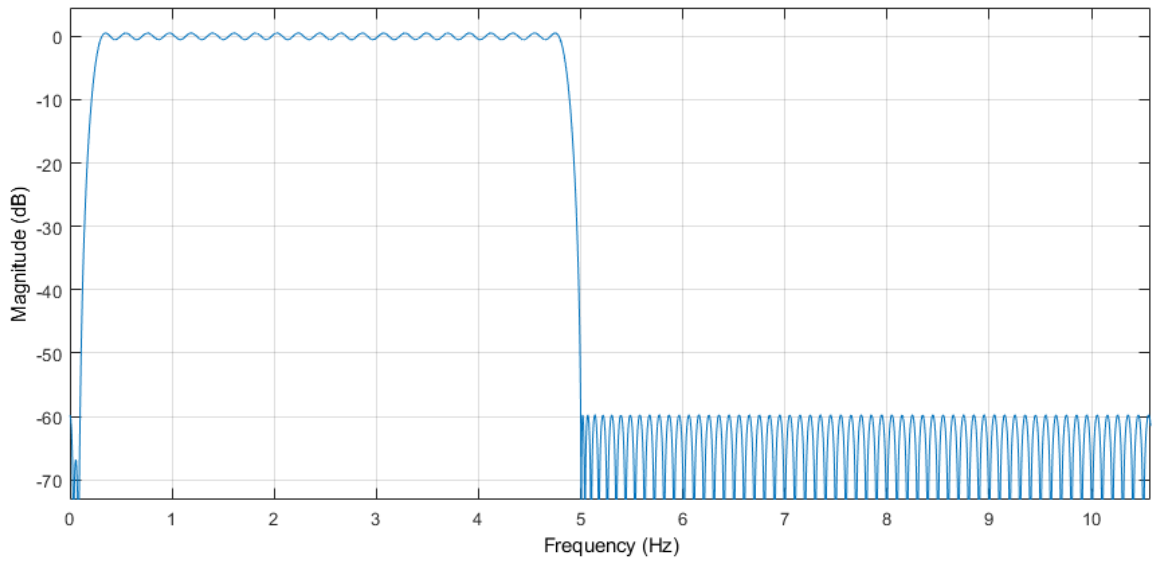

Figure 4.5: Magnitude response of the designed equiripple bandpass filter for pre-processing of the PPG signal.

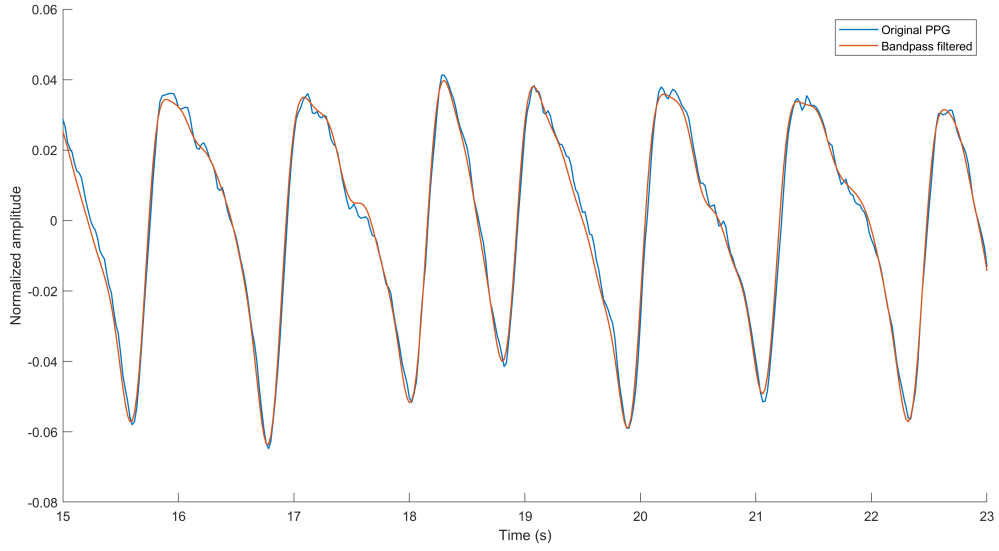

Figure 4.6: Extracted section of PPG data from the Simband smartwatch dataset [31].

consecutive systolic pulses can vary significantly. The 0.4 second increment ensures an overlap of 80% of the data in between two windows. This ensures that there is plenty of repetition of the systolic peaks across the windows, which will later be shown to be important for the selection of valid peaks. In the following subsections the proposed algorithm will be discussed in more detail.

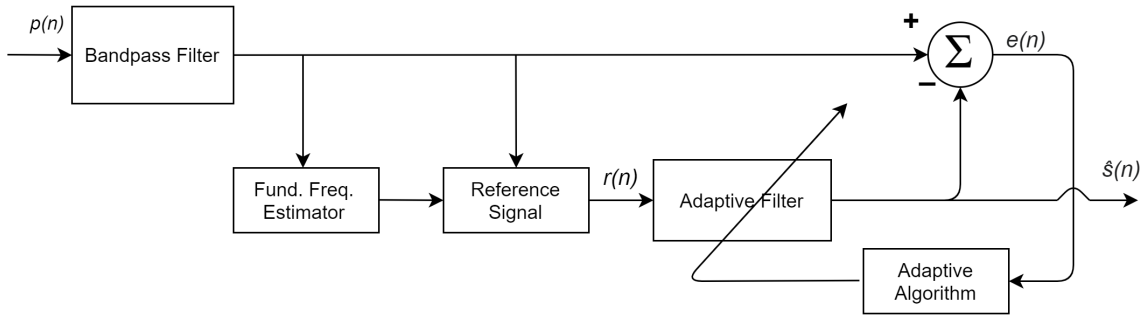

Figure 4.7: Block diagram of adaptive filtering system. $p(n)$ is the input signal from the PPG sensor. After applying a bandpass filter, the fundamental frequency is extracted from the signal, reference signal $r(n)$ is generated. The output $\hat{s}(n)$ is the enhanced signal, obtained by minimizing the error signal $e(n)$ towards this reference.

4.3.1. Reference Signal

The first step in generating the reference signal from the signal properties and waveform is the estimation of the fundamental frequency. The autocorrelation function preserves periodicity information of the input signal. The fundamental frequency is therefore estimated by computing the autocorrelation of the signal as seen in Equation (4.1). In this equation $R_p(n)$ is the autocorrelation of a PPG segment $p(n)$. When applied on the filtered data segment (Figure 4.6), the autocorrelation as seen in Figure 4.8 is obtained. Since the signal is assumed to be periodic, the first peak will indicate one period of the signal and thus gives an indication of the fundamental frequency [36]. The beat-to-beat heart rate is estimated when the fundamental frequency of the signal is calculated on a sufficiently small time segment. This is important for tracking sudden changes in the heart rate and not filter out the signal component. Hereafter, the signal is filtered with a bandpass having a bandwidth of half the estimated fundamental frequency. This bandwidth is found as an optimal value, narrow enough to ensure the heart pulse is tracked in a noisy signal, while being wide enough to allow for heart rate variability. The output is used as a reference signal for the adaptive filtering stage.

$$R_p(n) = \frac{\sum_{n=0}^{N-1} p(n)p(n+k)}{\sum_{n=0}^{N-1} p(n)^2}. \tag{4.1}$$

4.3.2. Adaptive algorithms

The following signal model is proposed for the PPG signal [33]:

$$p(n) = s(n) + m(n) + v(n). \tag{4.2}$$

Here $p(n)$ is the measured PPG signal, $s(n)$ is the desired noise-free PPG signal, $m(n)$ is the noise caused by motion and $v(n)$ is the high frequency noise caused, among other sources, by electrical interference. $v(n)$ is assumed to be normally distributed with a mean of zero [33]. The goal is to estimate $s(n)$ as accurate as possible from observing $p(n)$. The proposed method to do this is accomplished by an adaptive filter. The general topology for such an adaptive filter is based on minimizing the power of the reference signal components $r(n)$ in the error signal $e(n)$. There are several adaptive algorithms that can achieve this including: Least Mean Squares (LMS), Normalized Least Mean Squares (NLMS) and Recursive Least Sqaures (RLS) [9]. These algorithms use similar FIR structures, however the algorithm that tunes the filter coefficients differs in, for example, rate of convergence and computational complexity. For low complexity a NLMS filter was recommended by Schäk et al [33], which has a computational complexity of $\boldsymbol{O}(L)$ as compared to $\boldsymbol{O}(L^2)$ for RLS and Kalman. In addition, NLMS allows for a better control of the step-size μ of the Least Squares filter, which reduces the number of computations. NLMS however tends to converge much slower than RLS. Other, more optimal solutions, are based on a Kalman filter in combination with a fixed-interval smoother [9].

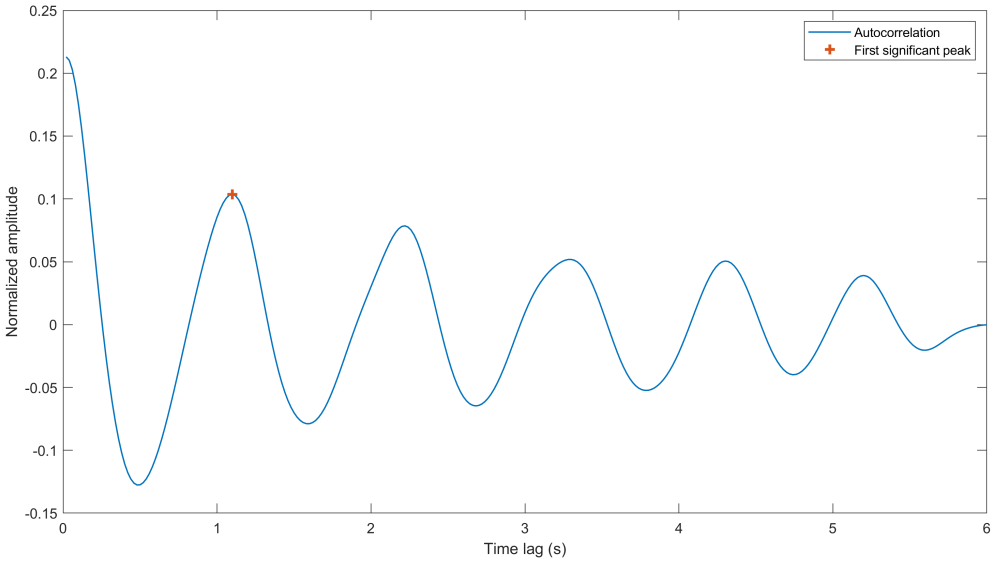

Figure 4.8: Plot of the computed autocorrelation of a noise free Simband dataset segment and the estimation of the first significant peak location.

This is however only suitable for off-line processing, as a large memory capacity and computational power is needed [37]. For RLS and Kalman based adaptive filtering computational power can be decreased significantly by processing the data in smaller blocks to avoid large matrix multiplications. This can be achieved by processing the data in a sliding window.

Based on an LMS algorithm the filter coefficients are given by the following equation[1]:

$$\hat{\boldsymbol{h}}(n+1) = \hat{\boldsymbol{h}}(n) + \mu \boldsymbol{r}(n) e(n), \tag{4.3}$$

where $\hat{\boldsymbol{h}}(n)$ are the estimated filter coefficients, $r(n)$ is the reference signal and $e(n)$ is the estimation error at time n obtained by:

$$e(n) = p(n) - \hat{\boldsymbol{h}}^T(n)\boldsymbol{r}(n), \tag{4.4}$$

Generally a value for the step-size μ is taken between: $0 < \mu < 2$ [38]. The filter coefficients for NLMS are similar but normalized by a factor $||\boldsymbol{r}(n)||^2 + \delta$, where δ is an arbitrary small non-zero constant (e.g. $\delta = 10^{-10}$) included for numerical stability reasons (avoiding division by zero):

$$\hat{\boldsymbol{h}}(n+1) = \hat{\boldsymbol{h}}(n) + \frac{\mu}{\boldsymbol{r}^T(n)\boldsymbol{r}(n) + \delta} \boldsymbol{r}(n) e(n). \tag{4.5}$$

Compared to the NLMS algorithm, the RLS algorithm has a faster convergence but also involves more mathematical operations. The standard RLS algorithm performs the following steps to update the filter coefficients:

$$\hat{\boldsymbol{h}}(n) = \hat{\boldsymbol{h}}(n-1) + \boldsymbol{K}(n) e(n), \tag{4.6}$$

here the error signal for a given time index is:

$$e(n) = p(n) - \hat{\boldsymbol{h}}(n-1)\boldsymbol{r}(\boldsymbol{n}). \tag{4.7}$$

The gain of the RLS algorithm is given by:

$$\boldsymbol{K}(n) = \frac{\boldsymbol{P}(n-1)\boldsymbol{r}(\boldsymbol{n})}{\lambda + \boldsymbol{r}^T(n)\boldsymbol{P}(n-1)\boldsymbol{u}(n)}, \tag{4.8}$$

[1] Note that in the following equations a bold symbol is used for vector/matrix and non-bold for scalar notation, e.g., the dot product of two vectors: $\boldsymbol{a}^T(n)\boldsymbol{b}(n)$, will result in a scalar.

here λ is the forgetting factor, generally between $0.9 < \lambda < 1$ [9], which determines the overall performance of the RLS algorithm. This parameter is a compromise between stability on the one hand, and fast convergence rate on the other hand. A low value of λ means that previous samples have less contribution to the covariance matrix, which makes the algorithm more sensitive to recent samples. $\boldsymbol{P}(n)$ is the inverse correlation matrix of the input signal which takes an initial value:

$$\boldsymbol{P}(0) = \begin{bmatrix} \delta^{-1} & 0 & \dots & 0 \\ 0 & \delta^{-1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \delta^{-1} \end{bmatrix}. \tag{4.9}$$

The regularization factor δ^{-1} is an arbitrary small constant for initialization. The inverse correlation matrix is updated by:

$$\boldsymbol{P}(n) = \lambda^{-1}\boldsymbol{P}(n-1) - \lambda^{-1}\boldsymbol{K}(n)\boldsymbol{r}^{T}(n)\boldsymbol{P}(n-1). \tag{4.10}$$

The other proposed adaptive algorithm is based on a Kalman filter. The Kalman filter representation is a (state space) generalization of the LMS and RLS filters [39]. The difference is that the Kalman filter is an optimal filter in mean square sense, achieving the shortest path of convergence. Whereas the gradient path of the other filter solutions may be longer and thus non-optimal. A drawback of the Kalman filter is the tracking lag present in the estimated parameters. This is disadvantageous if there are abrupt changes in the signal. However, the tracking lag of the filter parameters can be solved using a smoother. The fixed-interval Kalman smoother (a combination of a fixed-interval smoother and a Kalman filter) was found to be the most effective in reducing motion artefacts in PPG signals [9]. Starting from the general description of the state space representation:

$$\boldsymbol{x_k} = \boldsymbol{A_{k-1}}\boldsymbol{x_{k-1}} + \boldsymbol{n_{k-1}}, \tag{4.11}$$

$$y_k = \boldsymbol{C_k}\boldsymbol{x_k} + e_k, \tag{4.12}$$

where the state vector $\boldsymbol{x_k}$ is expressed as a random walk model, which makes the state transition matrix $\boldsymbol{A_k}$ an indentity matrix [9]:

$$\boldsymbol{x_k} = \boldsymbol{x_{k-1}} + \boldsymbol{n_{k-1}}. \tag{4.13}$$

Equation (4.13) is the state equation where state $\boldsymbol{x_k}$ is dependent on the previous state times the transition matrix plus the noise $\boldsymbol{n_{k-1}}$ of that state. y_k represents the observation data, $\boldsymbol{C_k}$ the observation matrix and e_k the observation noise. Here $\boldsymbol{x_k}$ correspond to the filter coefficient and y_k to the desired signal component $s(n)$. Both the state noise vector and the observation noise is assumed to be zero-mean Gaussian noise processes, $\boldsymbol{n_k} \sim N(0, \boldsymbol{Q_k})$ and $\boldsymbol{e_k} \sim N(0, \sigma_k^2)$, respectively. These are assumed to be independent, where $\boldsymbol{Q_k}$ is the state noise covariance matrix and σ_k^2 is the measurement noise variance [9]. The Kalman recursion algorithm provides an estimate for the state vector $\boldsymbol{x_k}$. According to Simon et al [40] the following recursion equations can be obtained from this model:

$$\hat{\boldsymbol{x}}_{\boldsymbol{k}}^{-} = \hat{\boldsymbol{x}}_{\boldsymbol{k-1}}^{+}, \tag{4.14}$$

here the minus sign indicates before and the plus sign indicated the state vector after y_k is observed.

$$\boldsymbol{P_k^-} = \boldsymbol{P_{k-1}^+} + \boldsymbol{Q_k}, \tag{4.15}$$

$$\boldsymbol{K_k} = \boldsymbol{P_k^-}\boldsymbol{C_k}^{T}(\boldsymbol{C_k}\boldsymbol{P_k^-}\boldsymbol{C_k}^{T} + \boldsymbol{R_k})^{-1}, \tag{4.16}$$

Here $\boldsymbol{K_k}$ is the Kalman filter gain, which updates the previous state vector estimate $\boldsymbol{x_k^-}$ and its covariance matrix $\boldsymbol{P_k^-}$ after an observation y_k according to:

$$\hat{\boldsymbol{x}}_{\boldsymbol{k}}^{+} = \hat{\boldsymbol{x}}_{\boldsymbol{k}}^{-} + \boldsymbol{K_k}(y_k - \boldsymbol{C_k}\hat{\boldsymbol{x}}_{\boldsymbol{k}}^{-}), \tag{4.17}$$

$$\boldsymbol{P_k^+} = (\boldsymbol{I} - \boldsymbol{K_k}\boldsymbol{C_k})\boldsymbol{P_k^-}. \tag{4.18}$$

$\boldsymbol{P^k}$ being the covariance matrix of the state vector and $\boldsymbol{C_k}$ being replaced by the reference signal tap-input $\boldsymbol{r(k)}$ and replacing $\boldsymbol{P_k^-}$ by $\boldsymbol{P_{k-1}}$, gives the Kalman gain:

$$\boldsymbol{K_k} = \frac{\boldsymbol{P_{k-1}}\boldsymbol{r}(k)}{(\boldsymbol{r}^T(k)\boldsymbol{P_{k-1}}\boldsymbol{r}(k) + \sigma_k^2)}. \tag{4.19}$$

This sets the filter coefficients as:

$$\hat{\boldsymbol{h}}(k) = \boldsymbol{h}(k-1) + \boldsymbol{K_k}(p(k) - \boldsymbol{r}^T(k)\hat{\boldsymbol{h}}(k-1)). \tag{4.20}$$

.

The autocorrelation matrix takes an initial value of:

$$\boldsymbol{P}_0 = \begin{bmatrix} \sigma_{kf}^2 & 0 & \dots & 0 \\ 0 & \sigma_{kf}^2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \sigma_{kf}^2 \end{bmatrix}, \tag{4.21}$$

and the autocorrelation matrix for the next iteration is given by:

$$\boldsymbol{P_k} = (\boldsymbol{P_{k-1}} - \boldsymbol{K_k}\boldsymbol{r}^T(k)\boldsymbol{P_{k-1}}). \tag{4.22}$$

The measurement noise variance σ_k^2 can be set to 1 and $\boldsymbol{Q_k} = 0$. This can be done since only the ratio of σ^2/σ_{kf}^2 influences the estimates [41]. The value of σ_{kf}^2 is the value that should be optimized. Tarvainen et al found an optimal value of $\sigma_{kf}^2 = 0.0003$.

When immediate update of the estimate is not necessary, future observations can be used for the estimation as well. If we continue to take in samples after an observation for a fixed period, this estimator is called a fixed-interval smoother [41]. An efficient implementation of this method is the Rauch-Tung-Striebel (RTS) smoother [40]. In this implementation the conventional Kalman filter must be executed initially, and then the backward RTS smoothing can be applied from sample $k = T$ to $k = 1$ iteratively:

$$\boldsymbol{K_k'} = \boldsymbol{P_k^+}(\boldsymbol{P_{k+1}^-})^{-1}, \tag{4.23}$$

$$\hat{\boldsymbol{x}}_{\boldsymbol{k}}' = \hat{\boldsymbol{x}}_{\boldsymbol{k}}^+ + \boldsymbol{K_k'}(\hat{\boldsymbol{x}}_{\boldsymbol{k+1}}' - \hat{\boldsymbol{x}}_{\boldsymbol{k+1}}^-). \tag{4.24}$$

Here the smoother gain matrix $\boldsymbol{K_k'}$ includes the error covariance matrices obtained during the Kalman filtering process of this interval. The resulting filter coefficients after applying the RTS smoothing are then formulated by:

$$\hat{\boldsymbol{h}}'(k) = \hat{\boldsymbol{h}}(k) + \boldsymbol{K_k'}(\hat{\boldsymbol{h}}'(k+1) - \hat{\boldsymbol{h}}(k+1)). \tag{4.25}$$

The matrices $\boldsymbol{P_k}$, $\boldsymbol{P_k^+}$ and coefficients $\hat{h}(k)$ must be stored in the forward Kalman process before applying the smoother. Real time computing can still be performed if the available computational power allows for this. However this is done in data blocks, since the smoother implementation requires a forward and backward computation path on a given interval of samples.

4.3.3. Peak detection

Since the signal is filtered to extract the systolic peaks from the waveform, a peak detection algorithm has been developed. This algorithm uses the output of the adaptive filtering stage. The method for finding the peaks in a specific data window is a straightforward approach:

1. Subtract the mean from the signal,
2. Find the positive regions in between two zero crossings,
3. Calculate the index of the maximum value on this segment as the peak location.

Since the peak detection is done on a sliding window with overlap, many of the peaks are observed across multiple data windows. For a sliding window of 2 seconds with 0.4 seconds increment, there is an overlap of 1.6 seconds in adjacent windows. This means that every data sample is observed across 5 different windows. This method improves the performance of the peak detection algorithms as it possible to compare the detected peaks across the windows. The following method is implemented to combine the detected peaks from the segmented data windows:

1. The detected peak locations of the data windows are combined in one array and then sorted by index,
2. The distance of each peak from the other peaks is computed. Peaks that have no surrounding peaks will be discarded to remove outliers,
3. Adjacent peaks are clustered by computing the average of the peak indexes that are within 10 samples distance of other peaks. The final peak location of this set of peaks is computed as this average index.

An example of the implementation of this method is seen in Figure 4.9. The fragment shows that the detected peaks in this time frame are observed within separate data windows. From this figure it can be confirmed that averaging across different windows leads to a better estimation of the peak locations and can effectively deal with outliers. The Figure in Appendix A.1 shows the result of this peak detection algorithm for a clean, noisy, AF and severe AF PPG segment of 30 seconds. From this figure it can be seen that for a clean PPG the beat-to-beat heart rate can be directly recovered from a peak detection of the bandpass filtered PPG signal. However, for noisy, AF and severe AF, peak detection is not reliable and further signal enhancement is required.

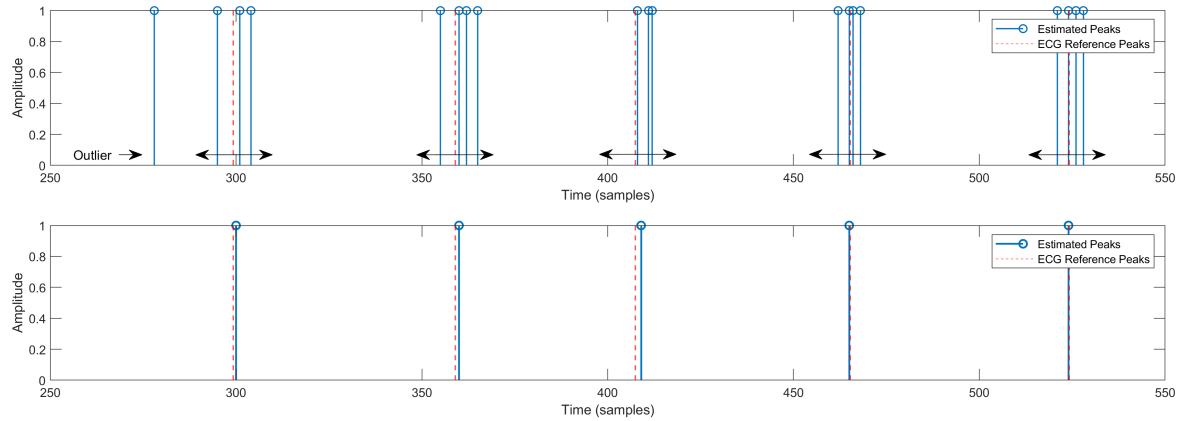

Figure 4.9: A 6 seconds data segment is plotted with the indicated detected peaks. What can be seen is that the heart beat is detected in multiple consecutive data windows. In each window the detected peak has a different location as seen in the top graph. The first peak is an outlier as it is the only peak being so far from another peak. The peaks that are close to each other are placed in a cluster. The final peak is calculated as the average of this cluster as seen in the bottom graph.

4.4. Discussion

For the pre-processing of the signal an equiripple bandpass filter is designed, because of the lowest filter order and flat frequency response in the stop-frequency range. This is interesting for low computational power implementations, however if this is not the case then a Chebyshev, Hamming or other filter solutions might be more viable options. These give an advantage if a very flat frequency response in the pass-band frequencies is required. It was not further investigated whether this would lead to significant improvement in the filtering of the signal. The adaptive filtering method proposed in this section involves generating a reference signal from the PPG signal itself. This proved to be a more robust method to enhance the signal quality as compared to using accelerometer data as a reference signal. Nevertheless, this is only a first attempt, and further improvements of statistical methods for generating the reference signal are to be made. These will most likely have great influence on the effectiveness of the adaptive filtering stage. For example, multiple PPG sensors using different frequencies can be used to generate a reference signal. A method proposed by Yousefi et al. [42] uses the difference signal of red and infrared PPG as a synthetic reference for NLMS adaptive filtering. This is interesting because different frequencies of light are expected to reflect from a different depth of tissue. This changes the effect of motion artefacts for these different frequency signals, while the heart signal remains similar. This would for example make estimation of the fundamental frequency more robust, which in the currently used method is still hard in the presence of high noise levels. This method does not involve filtering with a narrow bandpass for generating the reference signal, which in the current implementation could remove heart rate variability if this is severely present. The use of multiple PPG frequencies is therefore likely to be a more viable method in the context of AF detection, in which heart rate variability serves as a key feature for AF detection.

5

Results

To validate the results quantitatively, the final detected peaks that are computed based on the peaks in the individual windows are compared to the peaks that have been recorded by the ECG. Either the S or R peaks detected in the ECG signal are taken as reference, dependent on the peaks chosen by the Simband dataset. The ECG peaks occur with the same interval as the PPG peaks such that a time shift will lead to synchronized peak locations. Peaks that are located within an absolute range of 10 samples from the ECG reference peaks (corresponding to 0.2 seconds) are considered to be correct. The value of 10 samples is considered to be a good estimation considering the average change of a few samples in the tracking of a peak in the signal when applying adaptive filtering. On the other hand, the error caused by detection of a peak should be well below half the distance in between two pulses to accurately detect which pulse the peak belongs to. To further support the results a normalized heart rate error is computed for the bandpass filtered PPG and the enhanced PPG signals using the ECG heart rate as a reference. Since the heart rate is given for each interval between two pules, which can vary between ECG and PPG, the ECG reference is interpolated by re-sampling the ECG heart rate. The normalized heart rate error is computed as seen in equation (5.1). Here N is the number of beats, $HR(j)$ is the heart rate from the PPG signal and $Ref(j)$ is the heart rate from the ECG reference. The validation method is applied for the 30 seconds fragments provided by the Simband dataset. The code for generating the results and their validation can be found in Appendix (A.2.1).

$$Error = \sqrt{\frac{1}{N}\sum_{j=0}^{N}\left(\frac{HR(j)-Ref(j)}{Ref(j)}\right)^2} \tag{5.1}$$

5.1. Normal sinus rhythm

The following figures show the results of adaptive filtering using a NLMS, Kalman and RLS algorithm. The results are validated using a peak detection algorithm and the peaks detected from an ECG reference signal is used as ground truth data. The results shown are for a low noise (Figure 5.1) and high noise environment (Figure 5.2). The dataset annotates that the patients have a normal sinus rhythm, so no AF is present in these segments. From a visual inspection it is seen that all three algorithms are suitable for tracking the PPG signal in a low noise environment. In these scenarios the fundamental frequency of the signal is easy to track and the adaptive filtering works efficiently to remove abrupt changes in the signal that might be caused by noise. In a high noise environment NLMS and RLS show the most promising results when applied for peak detection. The results of the adaptive filtering structure combined with the peak detection method are seen for a low noise (Figure 5.3), and high noise (Figure 5.4) signal. The results for the detected peaks are seen as compared to the ECG reference peaks with the corresponding confusion matrix. Here the top left square is true positive, which means that at the location of the ECG peak there is one detected PPG peak within the allowed range. The top right square shows the number of false negatives, where there is a peak present in the ECG signal but is not detected. The bottom left square shows the number of false positives, here there is a peak detected in the PPG signal that is not in the ECG signal. The beat-to-beat heart rate error for a low and high noise 30 second segment is seen in table (5.1). Here the original signal is the bandpass filtered input signal and NLMS, Kalman and RLS are the output of the corresponding adaptive filtering structure. The original signal shows to have the highest error and RLS the lowest. The result was consistent for other time segments, showing

that the filtering structure reduces the heart rate error for clean and noisy segments. To further support these results, the peak detection after the adaptive filter is compared to the results of peak detection of the same segments without adaptive filtering, as seen in Figure A.1. From the confusion matrix it is seen that the results for peak detection are significantly improved after adaptive filtering, particularly in a noisy environment.

Table 5.1: Beat-to-beat HR error (Normal sinus rhythm)

Signal type	Original	NLMS	Kalman	RLS
Clean	0.059	0.034	0.010	0.021
Noisy	2.104	0.324	0.698	0.195

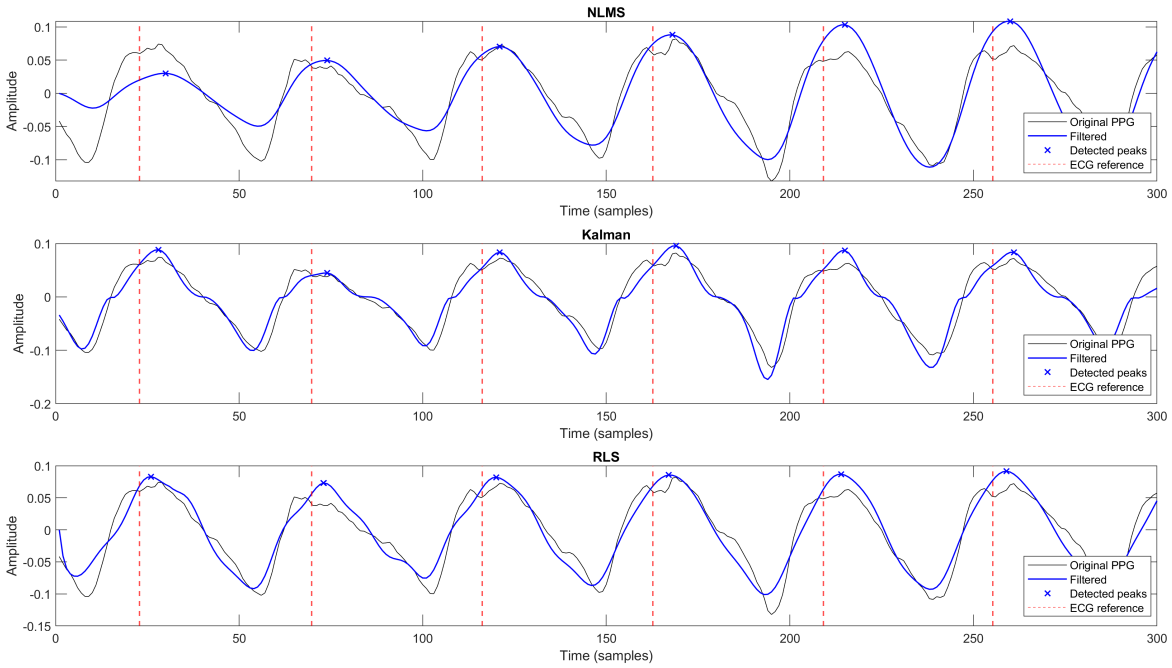

Figure 5.1: Plot of 6 seconds segment of low noise PPG data processed by the NLMS, Kalman and RLS adaptive filter. The dark line shows the original PPG signal, before the bandpass and adaptive filtering stages. The signal quality is very high, therefore the three different algorithms do not show remarkable differences in peak detection. The shape of the Kalman filtered signal seems to be slightly altered. Further observations are that the NLMS algorithm takes some time to track the amplitude of the signal, whereas the RLS and Kalman signal converge rapidly.

5.2. Atrial Fibrillation

Figure 5.5 shows a low noise signal window of 6 seconds and Figure 5.6 shows a medium noise level window both for a patient having an AF episode. For both NLMS, Kalman and RLS the adaptive filtering shows efficient noise reduction in the PPG signal measured during an AF episode. An important observation is that while the fundamental frequency is estimated for this segment, the adaptive filtering does not remove the variability information caused by the AF period, as this is the exact feature sought after. The quantitative validation of the filtering structure for the AF dataset is seen in Figure 5.7 for low noise and in Figure 5.8 for a medium noise environment. What is seen from comparing the two is that Figure 5.8 possesses a higher heart rate and a higher heart rate variability. This most likely indicates the severity of the AF episode. This fact is also reflected in the confusion matrix, where significantly more false negatives are present for all three algorithms. The algorithm therefore performs less during severe AF episodes, which could be problematic as the goal is to detect AF. The reason for this is most likely because the PPG signal during severe AF episodes is hard to differentiate from noisy PPG. The high frequency component introduced by these periods can lead to a signal that appears to have multiple systolic peaks within a single PPG pulse, after the first peak the signal does not drop below the mean before rising a second time. One would see only two small peaks within

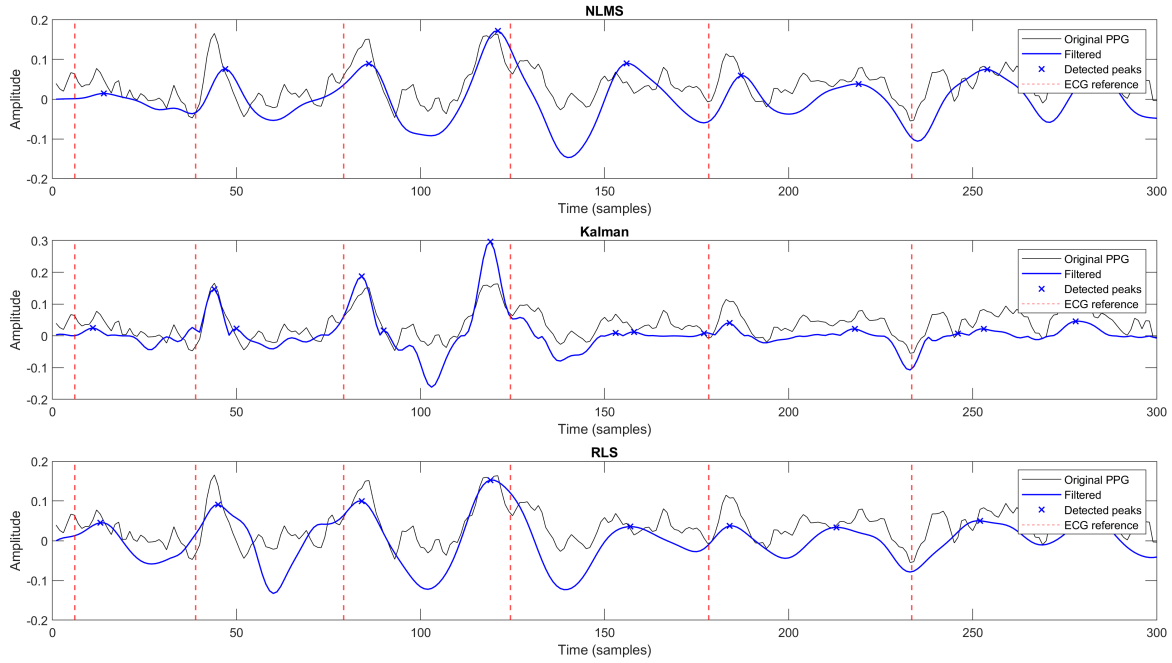

Figure 5.2: Plot of 6 seconds segment of high noise PPG data processed by the NLMS, Kalman and RLS adaptive filter. The dark line shows the original PPG signal, before the bandpass and adaptive filtering stages. The signal is plotted before bandpass filtering to show the level of noise present in the signal. In the left half of the signal all three algorithms seem to recover the signal, but for the right half the signal quality is too low to correctly track the pulse, which will induce false detections. The NLMS and RLS algorithms seem to induce less falsely detected peaks than the Kalman.

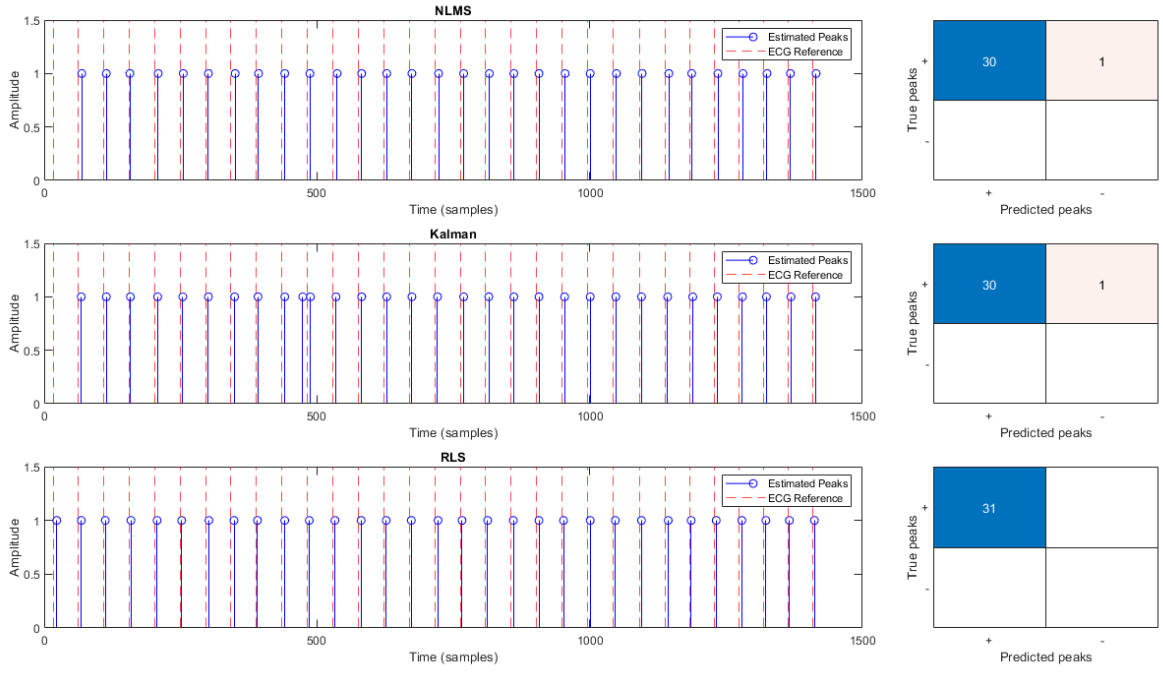

Figure 5.3: Results of the peak detection after adaptive filtering as compared to the ECG reference signal together with the derived confusion matrix. This result is obtained from a low noise PPG signal measured during a normal sinus rhythm. NLMS, Kalman and RLS show similar performance as all peaks are correctly detected except the first peak for NLMS and Kalman. This is because the RLS algorithm converges fast enough to detect the first peak.

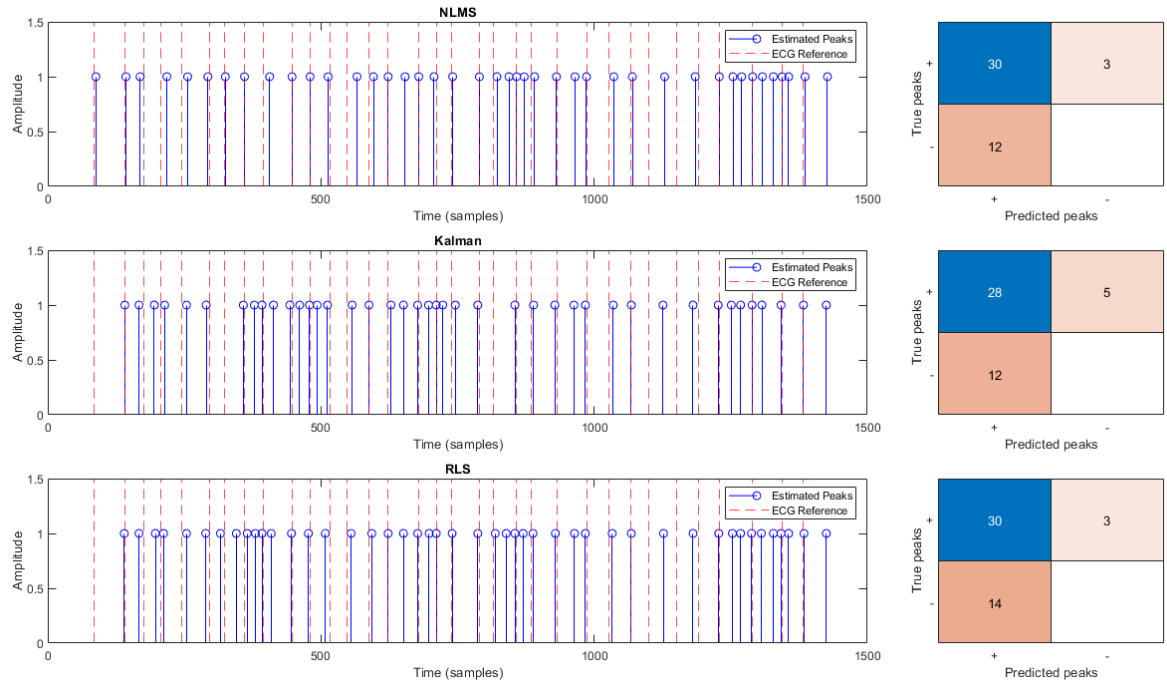

Figure 5.4: Results of the peak detection after adaptive filtering as compared to the ECG reference signal together with the derived confusion matrix. This result is obtained from a high noise PPG signal in which no AF is annotated. This however does not exclude the possibility of AF for this segment as there is a lot of noise present. This result shows that 12 false positives for NLMS and Kalman and 14 for RLS, which means many excess peaks are detected. From the confusion matrix it is concluded that NLMS seems to perform slightly better for this segment.

a larger periodic pulse which would be smoothed out by the adaptive filter to produce only one peak. The beat-to-beat heart rate error for a low and high noise 30 second segment is seen in Table (5.2). The result was consistent with other time segments, showing that the filtering structure reduces the heart rate error for AF and severe AF segments. The Kalman filter showed good performance in low noise and moderate AF situations, however not significantly better than NLMS and RLS, which show the best results for higher heart rate and more severe AF episodes. To support these results the peak detection after the adaptive filter is compared to the results of peak detection of the same AF segments without adaptive filtering, as seen in Figure A.1. For both the AF and severe AF segment the proposed adaptive filter shows a significant improvement in the peak detection for all three algorithms.

Table 5.2: Beat-to-beat HR error (AF period)

Signal type	Original	NLMS	Kalman	RLS
AF	0.186	0.020	0.013	0.044
Severe AF	2.019	0.246	0.764	0.269

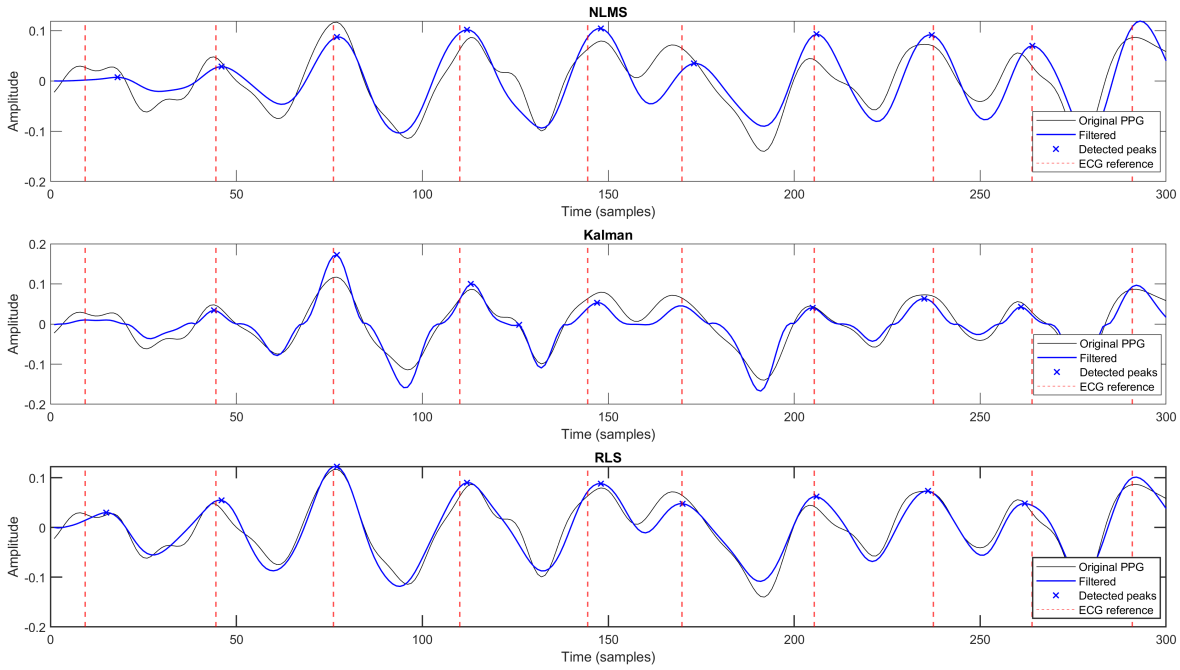

Figure 5.5: Plot of a 6 seconds segment of low noise PPG data measured during an AF episode. The signal is again processed by the NLMS, Kalman and RLS adaptive filter and compared to the ECG reference peaks. The dark line shows the original PPG signal after bandpass filtering. All three algorithms show equal capability of tracking the irregular heart beat in this segment, however Kalman gives some falsely detected peaks.

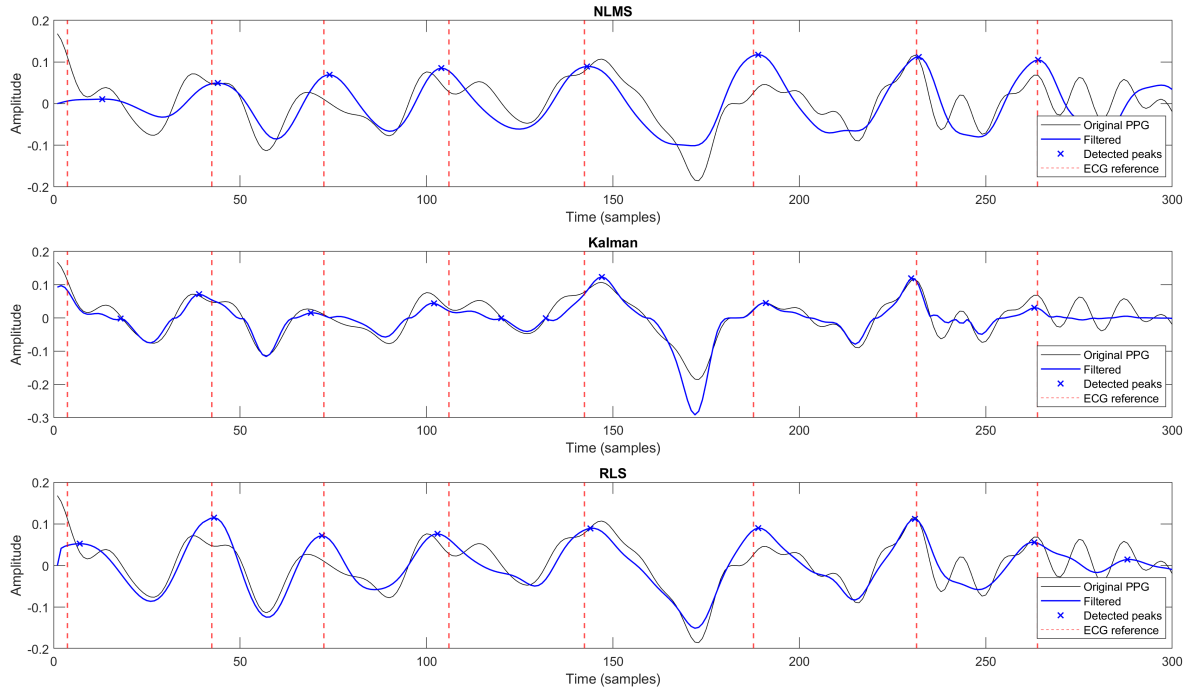

Figure 5.6: Plot of a 6 seconds segment measured during an AF episode in a medium noise level environment. NLMS and RLS display better enhancement of the signal for this segment, particularly as seen from comparing the detected peaks to the ECG reference signal.

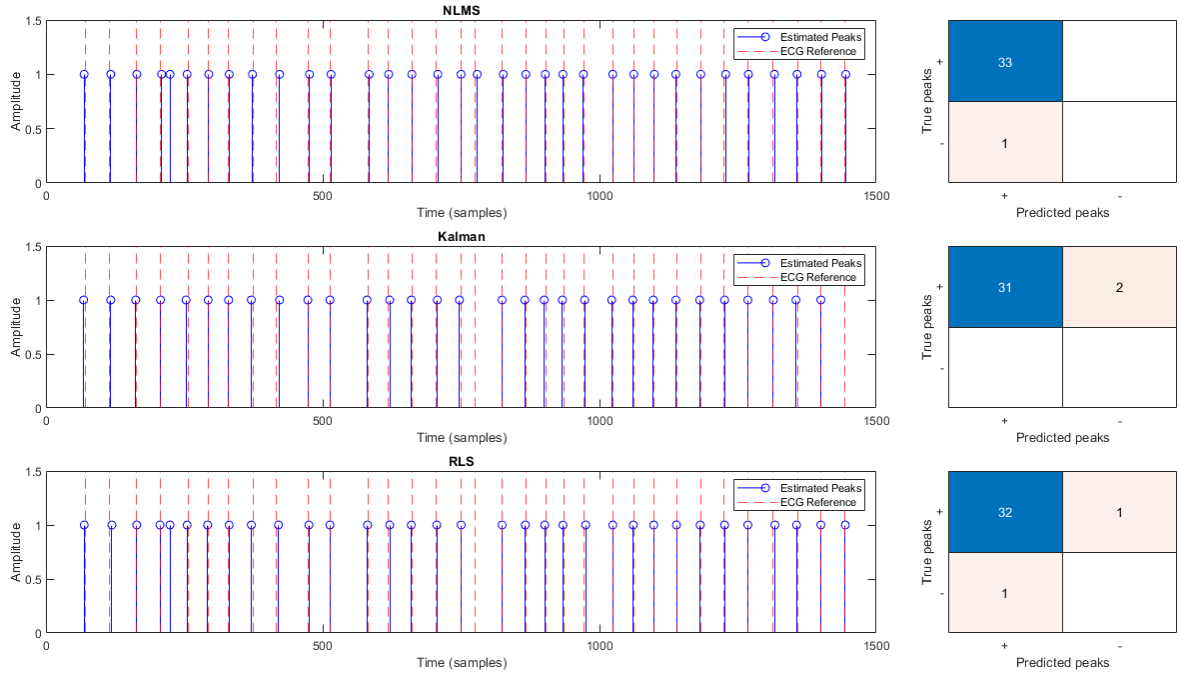

Figure 5.7: Results of the peak detection after adaptive filtering a low noise segment measured during an AF episode. NLMS, Kalman and RLS show comparable performance in peak detection. As seen from the confusion matrix, NLMS performs better than RLS by only one false negative.

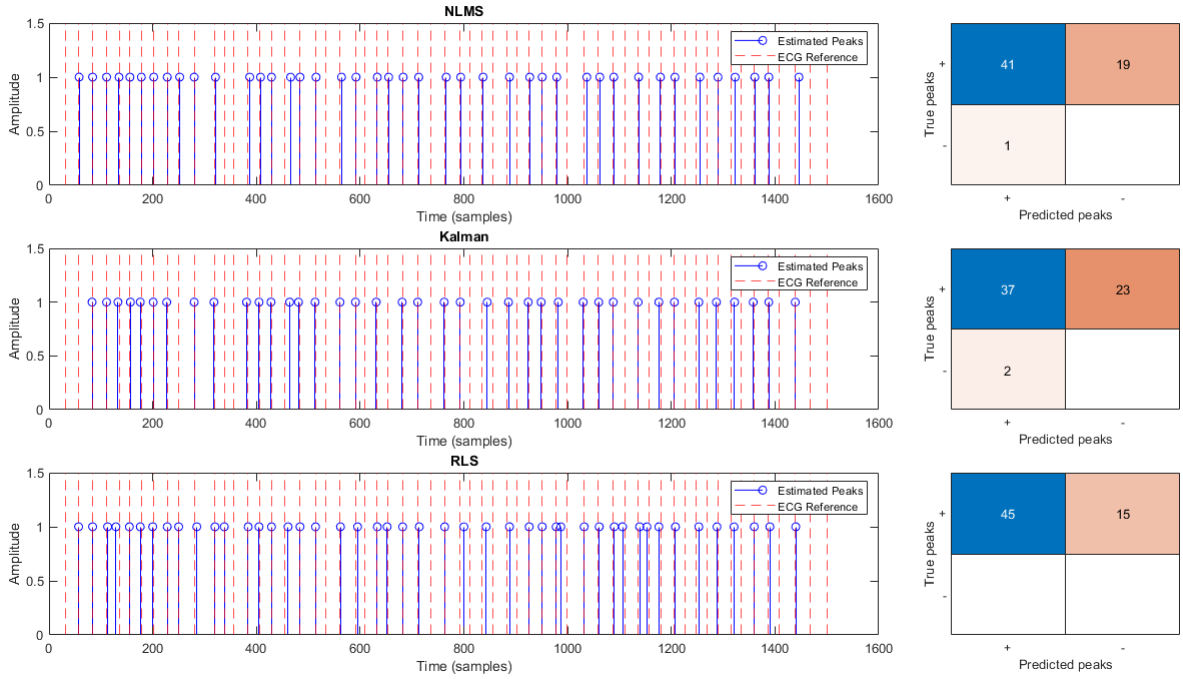

Figure 5.8: Results of the peak detection after adaptive filtering segment measured during an AF episode. It can be seen that the AF episode is more severe and the heart rate is higher than that seen in Figure (5.7). This causes a significant number of false negatives in the peak detection, as seen from the confusion matrix. RLS seems to outperform NLMS and Kalman in terms of detected peaks as compared to the ECG reference.

6

Discussion

6.1. Interpretation of results

The primary objective of this research was to evaluate methods for reduction of noise in PPG signals. This aim has been achieved as seen from both visual and quantitative validation, provided in chapter 5. The importance of computing both heart rate error and confusion is that the heart rate error does not provide information on the number of detected heart beats, but only on the error of the ones detected. These results are therefore complementary. For non-AF data segment it can be seen that the signal can be recovered in low to medium high noise environments. From a peak detection method it appears that the NLMS and RLS adaptive algorithm perform equally, however NLMS achieving this result at a filter order of 300, while RLS only has an order of 30. Based on the beat-to-beat heart rate error, the Kalman algorithm appears to perform the best in low noise environments ($Error = 0.010$), however the error for every signal of this type is relatively small ($Error < 0.060$). In high noise environments the RLS shows significantly better performance ($Error = 0.195$) than the other algorithms. For AF data segments a distinction between AF and severe AF periods is made, where severe AF often goes paired with a raised heart rate, heart rate variability and noise. For low noise AF the Kalman filter shows the smallest heart rate error ($Error = 0.013$), comparable to that of NLMS ($Error = 0.020$). For a severe AF period NLMS ($Error = 0.246$) and RLS ($Error = 0.269$) showed equal performance based on the heart rate error, but RLS performs best on peak detection as seen from the confusion matrix in Figure 5.8. Based on the accuracy of the peak detection and the heart rate error, the RLS algorithm showed the best performance overall, closely followed by NLMS. Given the lower filter order and faster rate of convergence, RLS is considered the most viable option for the adaptive filtering structure presented in this thesis.

6.2. Future work

Currently the validation of the algorithm is done by indicating the detection error and accuracy of specific 30 seconds data segments. The developed method of using peak detection can influence the result, as this can perform better for a certain algorithm based on the produced waveform. It is therefore advisable to improve the method for determining the beat-to-beat heart rate, either by improving or replacing the current peak detection, as this directly impacts the accuracy and error of our outcome. It is also not clear whether the output of the filtering structure is viable for AF detection. It is therefore advisable to develop a (machine learning) algorithm for evaluation of the the enhanced signal and the detected pulses.

Currently there is no system in place to detect segments that are too noisy to recover features. If the accuracy of machine learning classification is lower than expected, the miss classified segments can be further reviewed to learn about what types of noise are still present in the enhanced signal and how they can be recognized in a future noise detecting system. Furthermore, alternative features to heart rate variability may be extracted from the enhanced PPG signal in future work.

7

Conclusion

An algorithm for PPG signal enhancement in the presence of noise has been developed for the evaluation of Atrial Fibrillation. The methods have been validated by using the presented peak detection mechanism for reliable beat-to-beat heart rate extraction. The results are compared to ECG ground truth heart rate data, which is provided by the used dataset. The adaptive filtering structure that is developed tests the performance of NLMS, RLS and Kalman as compared to no adaptive filtering. The result was a significant reduction in the heart rate error and improved peak detection for all three adaptive algorithms. Moreover, the RLS algorithm proved to be the most viable option for reducing movement artefacts and AF detection. The result also shows that artefacts can be reduced without the need of additional hardware such as an accelerometer. Shortcomings of the algorithm were evident for high noise and severe AF periods, in which a substantial portion of the signal is not correctly recovered. This is noticed from the peak detection and heart rate error in these segments. For these reasons further research is recommended on the topic of adaptive filtering in context of AF, especially in the robust generation of a reference signal. Furthermore, additional validation (e.g. using machine learning) is required to make conclusion about the ability to correctly classify AF from the enhanced signal and systolic peak locations given by our algorithm.

7.1. Reflection on the program of requirements

The algorithm that is designed in chapter 4 and tested in chapter 5 shows good results in extracting the beat-to-beat heart rate. It achieves the requirement of detecting 85% of peaks in the signal when calculating this as the number of correctly detected peaks divided by the number of total reference peaks in the segment. However, in high noise environment the false detection of extra peaks can skew the performance for classification. The improved peak detection should improve the results of the AF detection algorithm designed by the classification subgroup. A method to assess the suitability of the enhanced signal for the extraction of other features has not been developed, as a lack of ground truth data for other features prevents a direct evaluation. We propose to compare the classification accuracy for the enhanced PPG signal and a bandpass filtered PPG signal, where both methods use the peaks extracted from the enhanced signal. This way the influence of the adaptive filter on the other features can be verified without the influence of better peak detection. Furthermore, the filtering algorithm currently does not perform any segment rejection. This leaves the system prone to errors due to segments that contain signals that are unrecoverable due to heavy distortion or lack of signal, that should not be processed. This segments rejection system turned out to be a large topic on its own and was therefore left out of the scope. It is suggested to test the current system in a fully implemented prototype, estimate the impact of the lack of noise rejection and then design a system based on the results. The provided Matlab algorithm is suitable to be implemented on the ARM micro controller that is selected for the prototype of the AF detection system, using available C++ signal processing libraries.

A

Appendix

A.1. Additional results

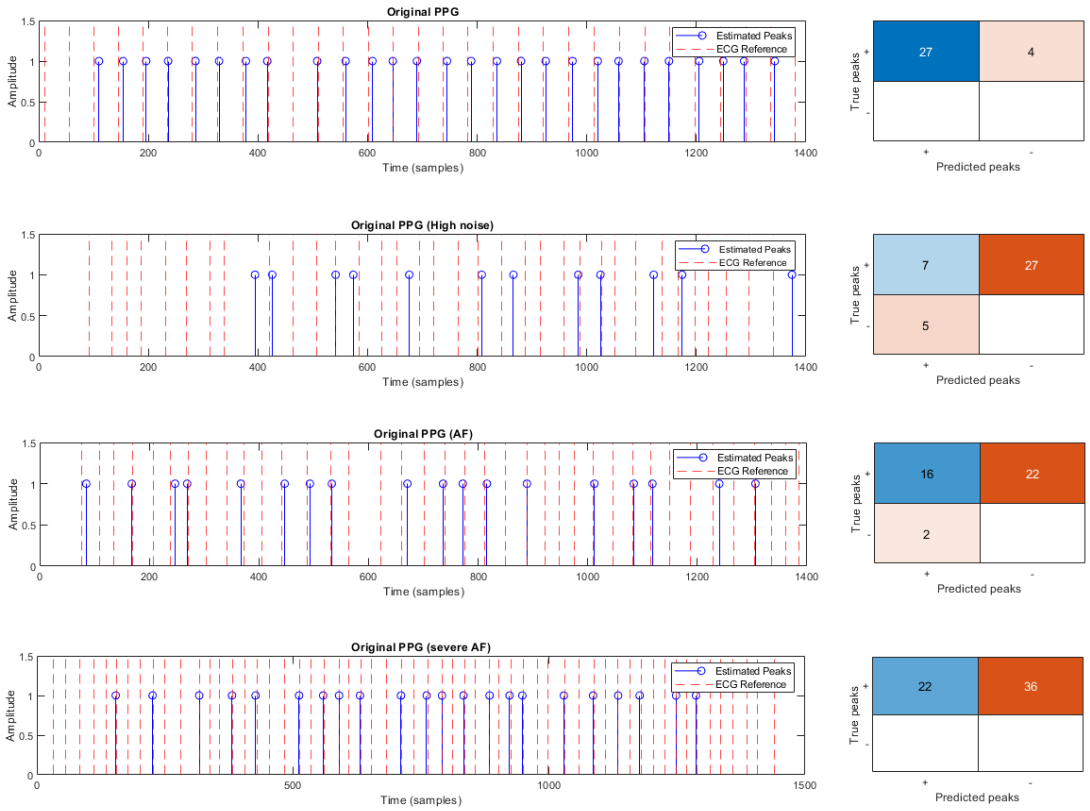

Figure A.1: Results from the peak detection of the PPG signal without adaptive filtering (after bandpass filtering). The first and second plot show a clean and noisy PPG segment, corresponding to the same segments seen in Figure 5.3 and Figure 5.4, respectively. The third and fourth plot show the results for AF and severe AF PPG segments, corresponding to the same segments seen in Figure 5.7 and Figure 5.8, respectively. From a comparison of the results it is clear that the unfiltered signal performs worse than the filtered signal for all the signal types.

A.2. Matlab scripts

A.2.1. Main file

```
close all;

%% Adaptive filtering structure
fs_ACC = 30; % Hz, downsampled ACC frequency.
fs_PPG = 50; % Hz, downsampled PPG frequency.
Fs = fs_PPG;

%Downsample with Q, upsample with P to obtain same size for acc/ppg if needed.
P = 3; Q = 3;
ppg = resample(PPG_buffer,P,Q);
ecg_peaks = ((refECG_pkloc)/2.56+8);

%Remove mean level from signal
p = (ppg- mean(ppg));
a = (ACC_buffer - mean(ACC_buffer));

%Bandpass filtering: 30th order. 30/2 Hz*(0.01 <->0.2) = 0.15<->5 Hz.
load('Bandpass_60dB.mat')
band= Bandpass60dB;
p_f = conv(p,band,'same');
a_f = conv(a,band,'same');

%Time axis for plotting
L_ppg = length(ppg);
t = (1:1:L_ppg)/Fs;

%plot the bandpass section:
% hold on
% plot(t,p/norm(p))
% plot(t,p_f/norm(p_f))
% xlim([15,21])
% legend('Original PPG','Bandpass filtered')
%
% xlabel('Time (s)')
% ylabel('Normalized amplitude')

%Sliding windows approach to filter data segments

%Uncomment either 1 or 2:
%1. For optimal peak detection, can comment out signal plots:
Lwindow = 100;      %corresponding to 2 sec of signal
factor = 5;         %window step size of 2/5 = 0.4 sec

%2. For plotting of time segments uncomment signal plots and comment result plots:
%   Lwindow = 300;      %corresponding to 2 sec of signal
%   factor = 1;         %window step size of 2/5 = 0.4 sec

[Nw,Xppg] = slidingWindow(p_f,Lwindow,factor);
[Nwx,Xppg_n] = slidingWindow(p,Lwindow,factor);

%initializations for fundamental freq estimation:
Rx = zeros(Nw,Lwindow);
P_optimal = zeros(Nw,1);
```

```
t_window = (1:1:Lwindow)/Fs;
w0 = zeros(Nw,1);

%init vectors for detected peaks:
stored_peaks_kalman = [];
stored_peaks_NLMS = [];
stored_peaks_RLS = [];
stored_peaks_normal = [];

for i_w = 1:Nw      %for each window from matrix Xppg
  %print progress
    fprintf('%d%%\n', round(i_w/Nw*100));
  %start and final sample of each window as function of width and increment
    start_index = (i_w-1)*Lwindow/factor;
    final_index = start_index + Lwindow;

  %autocorrelations of PPG signal with sliding window
    Rx(i_w,:) = autoCorr(Xppg(i_w,:));

  %find the first significant peak in the autocorrelation
    Rx_n = Rx(i_w,:);
    [peaks,ipeaks] = findpeaks(Rx_n);    %find all peaks in autocorrelation
    [p_val,p_index] = maxk(peaks,4);     %find 4 most prominent peaks
    p_index = min(p_index);              %take the first significant peak index
    period = ipeaks(p_index);
    P_optimal(i_w,1) = Fs/(period);

%     figure(i_w)
%     plot(t_window,Rx_n);
%     hold on
%     plot(t_window(period),Rx_n(period),'x');
%     xlabel('Time lag (s)')
%     ylabel('Normalized amplitude')
%     legend('Autocorrelation','First significant peak')

%Computes the reference signal as a sharp bandpassed version of the
%original
    w0(i_w) = (P_optimal(i_w,1)/(Fs/2));  %mid frequency of notch filter
    bw(i_w) = w0(i_w);
    bandpass_ref = fir1(100,[w0(i_w) - bw(i_w)/2, w0(i_w)+bw(i_w)/2],'bandpass');
    band_ref = conv(Xppg(i_w,:),bandpass_ref,'same');

%Adaptive filter algorithms:
  %adaptive filtering of signal using a NLMS algorithm
    FIRord_nlms = 300;
   [p_nlms,e_nlms] = nlms_matlab(band_ref,Xppg(i_w,:),FIRord_nlms);

  %adaptive filtering using a RLS algorithm
    lambda = 0.9996;
    FIRord_rls = 30;
   [p_rls,h_rls] = rls_matlab(band_ref,Xppg(i_w,:),lambda,FIRord_rls);

  %adaptive filtering using a Kalman Filter
   [p_kalman,p_smooth,h_kalman,h_smooth] = Adaptive_Kalman(band_ref,Xppg(i_w,:));

  %finding local maxima of the filtered signal for peak retrieval
```

```
    [idx_normal] = detectPeaks(Xppg_n(i_w,:));
    [idx_kalman] = detectPeaks(p_kalman);
    [idx_nlms] = detectPeaks(p_nlms);
    [idx_rls] = detectPeaks(p_rls);

     idx_kalman = idx_kalman(:)';
     idx_nlms = idx_nlms(:)';
     idx_rls = idx_rls(:)';
     idx_normal = idx_normal(:)';

   %store all found peak locations in a vector to compare
     stored_peaks_kalman = [stored_peaks_kalman, idx_kalman+start_index]; %#ok<AGROW>
     stored_peaks_NLMS = [stored_peaks_NLMS, idx_nlms+start_index]; %#ok<AGROW>
     stored_peaks_RLS = [stored_peaks_RLS, idx_rls+start_index]; %#ok<AGROW>
     stored_peaks_normal = [stored_peaks_normal, idx_normal+start_index]; %#ok<AGROW>

     i_pk = find(ecg_peaks >= start_index & ecg_peaks <= final_index);
     peaks_in_window = ecg_peaks(i_pk)-start_index;


%COMMENT FROM HERE TO IGNORE SIGNAL PLOTS:

%    %plot the filtered data together with original data
%      figure(i_w)
%      subplot(311)
%      plot(Xppg_n(i_w,:)/norm(Xppg_n(i_w,:)),'color','black')
%      hold on
%      plot(p_nlms/norm(p_nlms),'color','blue')
%      plot(idx_nlms,p_nlms(idx_nlms)/norm(p_nlms),'x','color','blue')
%      hold off
%      xlabel('Time (samples)')
%      ylabel('Amplitude')
%      for i = 1:length(peaks_in_window)
%         xline(peaks_in_window(i),'--r');
%      end
%      hold off
%      legend('Original PPG','Filtered','Detected peaks','ECG reference')
%      title('NLMS')
%
%      subplot(312)
%      plot(Xppg_n(i_w,:)/norm(Xppg_n(i_w,:)),'color','black')
%      hold on
%      plot(p_kalman/norm(p_kalman),'color','blue')
%      plot(idx_kalman,p_kalman(idx_kalman)/norm(p_kalman),'x','color','blue')
%      hold off
%      xlabel('Time (samples)')
%      ylabel('Amplitude')
%      for i = 1:length(peaks_in_window)
%         xline(peaks_in_window(i),'--r');
%      end
%      hold off
%      legend('Original PPG','Filtered','Detected peaks','ECG reference')
%      title('Kalman')
%
%      subplot(313)
%      plot(Xppg_n(i_w,:)/norm(Xppg_n(i_w,:)),'color','black')
```

```
%     hold on
%     plot(p_rls/norm(p_rls),'color','blue')
%     plot(idx_rls,p_rls(idx_rls)/norm(p_rls),'x','color','blue')
%     xlabel('Time (samples)')
%     ylabel('Amplitude')
%     for i = 1:length(peaks_in_window)
%        xline(peaks_in_window(i),'--r');
%     end
%     hold off
%     legend('Original PPG','Filtered','Detected peaks','ECG reference')
%     title('RLS')

%TO HERE

end

%% Results and Evaluation from peak detection

max_dist = 10;     %maximum allowed distance for a peak to have from GT

%final selection of peaks from all peaks detected:
selpeaks_NLMS = cluster_mean(stored_peaks_NLMS,max_dist);
selpeaks_NLMS = selpeaks_NLMS(2:end-1);

selpeaks_kalman = cluster_mean(stored_peaks_kalman,max_dist);
selpeaks_kalman = selpeaks_kalman(2:end-1);

selpeaks_RLS = cluster_mean(stored_peaks_RLS,max_dist);
selpeaks_RLS = selpeaks_RLS(2:end-1);

selpeaks_normal = cluster_mean(stored_peaks_normal,max_dist);
selpeaks_normal = selpeaks_normal(2:end-1);

%COMMENT FROM HERE TO SUPRESS VALIDATION ALGORITHM:

%remove first and last two peak from ecg since this is not recovered
ecg_peaks = ecg_peaks(1:end-2,1);

%check distances between ecg peak and detected peak:
  %initializations:
  N_ecg_pks = length(ecg_peaks);
  dist_i = zeros(N_ecg_pks,1);

  close_peaks_kalman = zeros(N_ecg_pks,1);
  close_peaks_NLMS = zeros(N_ecg_pks,1);
  close_peaks_RLS = zeros(N_ecg_pks,1);


  %True Positives and False Negatives:
  for i=1:N_ecg_pks

      dist_i_kalman = abs(selpeaks_kalman - ecg_peaks(i));
      dist_i_NLMS = abs(selpeaks_NLMS - ecg_peaks(i));
      dist_i_RLS = abs(selpeaks_RLS - ecg_peaks(i));

      close_peaks_kalman(i) = length(find(dist_i_kalman < max_dist));
```

```
        close_peaks_NLMS(i) = length(find(dist_i_NLMS < max_dist));
        close_peaks_RLS(i) = length(find(dist_i_RLS < max_dist));
    end
    TP_NLMS = sum(close_peaks_NLMS>0);
    FN_NLMS = sum(close_peaks_NLMS==0);

    TP_kalman = sum(close_peaks_kalman>0);
    FN_kalman = sum(close_peaks_kalman==0);

    TP_RLS = sum(close_peaks_RLS>0);
    FN_RLS = sum(close_peaks_RLS==0);

    %initializations for the three algorithms:
    N_pks_NLMS = length(selpeaks_NLMS);
    N_pks_kalman = length(selpeaks_kalman);
    N_pks_RLS = length(selpeaks_RLS);

    dist_j_NLMS = zeros(N_ecg_pks,1);
    dist_j_kalman = zeros(N_ecg_pks,1);
    dist_j_RLS = zeros(N_ecg_pks,1);

    close_peaks_NLMS = zeros(N_pks_NLMS,1);
    close_peaks_kalman = zeros(N_pks_kalman,1);
    close_peaks_RLS = zeros(N_pks_RLS,1);

    %False Positives:
    for j=1:N_pks_NLMS
        dist_j_NLMS = abs(selpeaks_NLMS(j)-ecg_peaks);
        close_peaks_NLMS(j) = length(find(dist_j_NLMS < max_dist));
    end

    for j=1:N_pks_kalman
        dist_j_kalman = abs(selpeaks_kalman(j)-ecg_peaks);
        close_peaks_kalman(j) = length(find(dist_j_kalman < max_dist));
    end

    for j=1:N_pks_RLS
        dist_j_RLS = abs(selpeaks_RLS(j)-ecg_peaks);
        close_peaks_RLS(j) = length(find(dist_j_RLS < max_dist));
    end

    FP_NLMS = sum(close_peaks_NLMS == 0);
    FP_kalman = sum(close_peaks_kalman == 0);
    FP_RLS = sum(close_peaks_RLS == 0);

    accuracy_NLMS = sum(close_peaks_NLMS)/N_ecg_pks;
    accuracy_kalman = sum(close_peaks_kalman)/N_ecg_pks;
    accuracy_RLS = sum(close_peaks_RLS)/N_ecg_pks;

    Conf_matrix_NLMS = [TP_NLMS, FN_NLMS; FP_NLMS, 0];
    Conf_matrix_kalman = [TP_kalman, FN_kalman; FP_kalman, 0];
    Conf_matrix_RLS = [TP_RLS, FN_RLS; FP_RLS, 0];

%%plot of all peaks detected
% figure()
%   subplot(211)
```

```
%   stem(stored_peaks,ones(length(stored_peaks),1))
%   for i = 1:length(ecg_peaks)
%       xline(ecg_peaks(i),'--r');
%   end
%   xlabel('Time (samples)')
%   ylabel('Amplitude')
%   legend('Estimated Peaks','ECG Reference Peaks')


%plot of final selection of peaks for NLMS algorithm:
figure()
  subplot(3,4,[1 2 3])
  stem(selpeaks_NLMS,ones(1,N_pks_NLMS),'color','blue')
  for i = 1:N_ecg_pks
     xline(ecg_peaks(i),'--r');
  end
  xlabel('Time (samples)')
  ylabel('Amplitude')
  ylim([0,1.5])
  legend('Estimated Peaks','ECG Reference')
  title('NLMS')

  subplot(3,4,[5 6 7])
  stem(selpeaks_kalman,ones(1,N_pks_kalman),'color','blue')
  for i = 1:N_ecg_pks
     xline(ecg_peaks(i),'--r');
  end
  xlabel('Time (samples)')
  ylabel('Amplitude')
  ylim([0,1.5])
  legend('Estimated Peaks','ECG Reference')
  title('Kalman')

  subplot(3,4,[9 10 11])
  stem(selpeaks_RLS,ones(1,N_pks_RLS),'color','blue')
  for i = 1:N_ecg_pks
     xline(ecg_peaks(i),'--r');
  end
  xlabel('Time (samples)')
  ylabel('Amplitude')
  ylim([0,1.5])
  legend('Estimated Peaks','ECG Reference')
  title('RLS')

  %plot the confusion matrix
  subplot(3,4,4)
  confusionchart(Conf_matrix_NLMS,{'+','-'})
  xlabel('Predicted peaks')
  ylabel('True peaks')
  subplot(3,4,8)
  confusionchart(Conf_matrix_kalman,{'+','-'})
  xlabel('Predicted peaks')
  ylabel('True peaks')
  subplot(3,4,12)
  confusionchart(Conf_matrix_RLS,{'+','-'})
  xlabel('Predicted peaks')
```

```
  ylabel('True peaks')

%TO HERE

%Evaluation of accuracy expressed in heart rate error:
%Error = sqrt(1/N sum [(HR(j) - Ref(j))/Ref(j)]^2
%Beat-to-beat heart rate and sample index.

%makes the reference signal 'continuous' by resampling to length 1500.
Ref_ECG = 60./(diff(ecg_peaks)./Fs);
Ref_ECG = resample(Ref_ECG,1500,length(Ref_ECG));

HR_PPG_NLMS = 60./(diff(selpeaks_NLMS)./Fs);
HR_NLMS_i = selpeaks_NLMS(2:end);

HR_PPG_kalman = 60./(diff(selpeaks_kalman)./Fs);
HR_kalman_i = selpeaks_kalman(2:end);

HR_PPG_RLS = 60./(diff(selpeaks_RLS)./Fs);
HR_RLS_i = selpeaks_RLS(2:end);

HR_PPG = 60./(diff(selpeaks_normal)./Fs);
HR_i = selpeaks_normal(2:end);

%Beat-to-beat normalized heart rate error:
Error = sqrt(1/length(HR_PPG).*sum(((HR_PPG-Ref_ECG(HR_i)')./(Ref_ECG(HR_i))'))^2);
Error_NLMS = sqrt(1/length(HR_PPG_NLMS).*sum(((HR_PPG_NLMS-Ref_ECG(HR_NLMS_i)')./(Ref_ECG(HR_NLMS_i)
Error_Kalman = sqrt(1/length(HR_PPG_kalman).*sum(((HR_PPG_kalman-Ref_ECG(HR_kalman_i)')./(Ref_ECG(HR
Error_RLS = sqrt(1/(length(HR_PPG_RLS)).*sum(((HR_PPG_RLS-Ref_ECG(HR_RLS_i)')./(Ref_ECG(HR_RLS_i)'))
```

A.2.2. Functions

```
%Sliding window function
function[N,X] = slidingWindow(x,M,factor)
 %X: output matrix with overlapping data as column vectors
 %x: input data array
 %M: number of samples in a window
 %N: amount of arrays of M samples
 %factor: M/factor number of samples increment per adjacent window
 %e.g. 1500 samples (30s) -> M = 300, factor = 3, gives 6s + 2s per window.

 Nx = length(x(:,1));                    %input length
 while mod(Nx,factor) ~= 0               %to make M devisible by our factor
     x(end) = [];                        %delete last sample if not devisible
     Nx = Nx - 1;
 end

 incr = floor(M/factor);                 %number of samples increment per window
 N = floor(1 + (Nx-M)/incr);             %amount of rows for a factor of overlap

 %check if there are enough samples to fill our matrix
 if N*M > Nx
    N = N - 1;  %decrease number of rows if not
 end

 X = zeros(N,M);                         %preallocating matrix

 %Vector x to matrix X with overlapping column vectors
 a = 1; b = M; k = 1;
 for i = 1:N
     while k <= M
         for j = a:b
             X(i,k) = x(j);
             k = k + 1;
         end
     end
     a = a + incr;
     b = b + incr;
     k = 1;
 end

end
```

```
%Autocorrelation function
function [C] = autoCorr(s)
%this function calculates the autocorrelation of equal length to input
s = s(:);
N = length(s);
C = zeros(N,1);

for k = 1:N
    s_l = [zeros(k-1,1); s(1:N-k+1)]; %lagged version of s, padded with zeros

    for n = 1:N                        %C(k) summing for all n
        C(k) = C(k) + s(n)*s_l(n);
    end
end
C = (C/norm(C));
```

```
end

%NLMS algorithm
function [ynlms,enlms,wnlms] = nlms_matlab(p,a,L)
nlms_filter = dsp.LMSFilter(L,'Method','Normalized LMS');
p = p'/norm(p);
a = a'/norm(a);
[~,mumaxmsenlms] = maxstep(nlms_filter,p);
nlms_filter.StepSize = mumaxmsenlms/80;
[ynlms,enlms,wnlms] = nlms_filter(a,p);
end

%RLS algorithm
function [y,e,w] = rls_matlab(p,a,lambda,L)
p = p'/norm(p);
a = a'/norm(a);
rls = dsp.RLSFilter(L, 'ForgettingFactor', lambda);
[y,e] = rls(a,p);
w = rls.Coefficients;
end

%Adaptive algorithm based on Kalman filter
function [s_kf,s_ks,h_kf,h_ks] = Adaptive_Kalman(x,u)

%x and u must be column vector x is the desired signal, u the reference
x = x(:)/norm(x);
u = u(:)/norm(u);

N = length(x);
h = zeros(N,N);

%values found from literature
sigma = 1;
sigma_kf = 0.0003;

%initial covariance identity matrix
P = zeros(N,N,N);
P(:,:,1) = (sigma_kf^2)*eye(N);

%forward path for adaptive kalman gain filtering:
for i = 2:N

 %update Kalman gain vector
 Kgain = P(:,:,i-1)*u/(transpose(u)*P(:,:,i-1)*u+sigma^2);
 %update state vector of filter coefficients
 h(:,i) = h(:,i-1)+Kgain*((x(i)-transpose(u)*h(:,i-1)));

 %update covariance matrix
 %save previous P for smoothing
 P(:,:,i) = (eye(N) - Kgain*transpose(u))*P(:,:,i-1);

end

h_prime = h;

% %backwards path for RLS smoothing algorithm:
```

```
% for i=N-1:-1:1
%
%  %Kprime for smoothing
%  K_prime = P(:,:,i)/(P(:,:,i+1));
%  h_prime(:,i) = h(:,i) + K_prime*(h_prime(:,i+1)-h(:,i+1));
% end

%filter output: Kalman filter and Kalman smoother (RLS):
 s_kf = abs(h(:,N)).*x;
 h_kf = h;

 s_ks = abs(h_prime(:,2)).*x;
 h_ks = h_prime;

end
```

```
%% custom r peaks detection
% use with parameters: x = ppg signal, averaging_window = 0.9, low_amp_window =20, low_amp_tres
% =0.6, f = sampling rate ppg signal,

function local_max = detectPeaks(x)
    local_max = [];
    x = x - mean(x);

    pos_zero_crossings = find(diff(sign(x))>0);
    neg_zero_crossings = find(diff(sign(x))<0);

        if (isempty(neg_zero_crossings) || isempty(pos_zero_crossings)) == 0
            if neg_zero_crossings(1) < pos_zero_crossings(1)
                neg_zero_crossings(1) = [];
            end
        else
            local_max = [];
            return
        end

        if (isempty(neg_zero_crossings) || isempty(pos_zero_crossings)) == 0
            if neg_zero_crossings(end) < pos_zero_crossings(end)
                pos_zero_crossings(end) = [];
            end
        else
            local_max = [];
            return
        end

    start_stop = cat(1, pos_zero_crossings.', neg_zero_crossings.');
    L = size(start_stop,2);
    size_int = 300;
    x_segments =zeros(size_int,L);

    for i = 1:L
        segment = x(start_stop(1,i):start_stop(2,i));
        segment(end:size_int) = 0;
        x_segments(:,i) = segment;
    end
    [~, Indices] = max(x_segments);
    local_max = Indices + start_stop(1,:)-1;
```

```
end

%This function takes all peak locations as input (sample index or time) and outputs the clustered av

function [peaks] = cluster_mean(peaks_loc,interval)
    %sort all the peaks chronologically
    peaks = sort(peaks_loc);
    Npeaks = length(peaks);
    number_of_friends = zeros(1,Npeaks);
    weight_av = 1;

    %1st iteration: Removing outliers from peak set by counting number of
    %friends
    for i = 1:Npeaks
        for j = i:Npeaks
            dx = abs(peaks(i)-peaks(j));
           if (dx < round(interval/2)) && (i ~= j)
              number_of_friends(i) = number_of_friends(i) + 1;
           end
        end
        if (number_of_friends(i) < 1)
            peaks(i) = 0;
        end
    end

    %2nd iteration: averaging clusters of peaks with weigthed average
    for i = 1:Npeaks
        for j = i:Npeaks
            dx = abs(peaks(i)-peaks(j));
           if (dx < interval) && (i ~= j)
              peaks(i) = round((peaks(i)*weight_av + peaks(j))/(weight_av+1));
              peaks(j) = 0;
              weight_av = weight_av + 1;
           else
              weight_av = 1;
           end
        end
    end
    %select only nonzero values from array
    i_sel = peaks>0;
    peaks = peaks(i_sel);
end
```

Bibliography

[1] A. D. Ceornodolea, R. Bal, and J. L. Severens, "Epidemiology and management of atrial fibrillation and stroke: Review of data from four european countries," vol. 2017. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5467327/

[2] I. C. Van Gelder and M. E. Hemels, "The progressive nature of atrial fibrillation: a rationale for early restoration and maintenance of sinus rhythm," vol. 8, no. 11, pp. 943–949. [Online]. Available: https://doi.org/10.1093/europace/eul107

[3] A. Hersi and D. G. Wyse, "Management of atrial fibrillation," vol. 30, no. 4, pp. 175–233. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0146280604001458

[4] L. M. Eerikäinen, L. Dekker, A. G. Bonomi, R. Vullings, F. Schipper, J. Margarito, H. M. de Morree, and R. M. Aarts, "Validating features for atrial fibrillation detection from photoplethysmogram under hospital and free-living conditions," in *2017 Computing in Cardiology (CinC)*, pp. 1–4, ISSN: 2325-887X.

[5] M. Lemay, S. Fallet, P. Renevey, J. Solà, C. Leupi, E. Pruvot, and J.-M. Vesin, "Wrist-located optical device for atrial fibrillation screening: A clinical study on twenty patients," in *2016 Computing in Cardiology Conference (CinC)*, pp. 681–684, ISSN: 2325-887X.

[6] V. D. A. Corino, R. Laureanti, L. Ferranti, G. Scarpini, F. Lombardi, and L. T. Mainardi, "Detection of atrial fibrillation episodes using a wristband device," vol. 38, no. 5, pp. 787–799. [Online]. Available: https://doi.org/10.1088/1361-6579/aa5dd7

[7] Atrial fibrillation - symptoms and causes. [Online]. Available: https://www.mayoclinic.org/diseases-conditions/atrial-fibrillation/symptoms-causes/syc-20350624

[8] D. Castaneda, A. Esparza, M. Ghamari, C. Soltanpur, and H. Nazeran, "A review on wearable photoplethysmography sensors and their potential future applications in health care," vol. 4, no. 4, pp. 195–202. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6426305/

[9] B. Lee, J. Han, H. J. Baek, J. H. Shin, K. S. Park, and W. J. Yi, "Improved elimination of motion artifacts from a photoplethysmographic signal using a Kalman smoother with simultaneous accelerometry," *Physiological Measurement*, vol. 31, no. 12, pp. 1585–1603, Oct. 2010. [Online]. Available: https://doi.org/10.1088/0967-3334/31/12/003

[10] S. H. Kim, D. W. Ryoo, and C. Bae, "Adaptive noise cancellation using accelerometers for the PPG signal from forehead," in *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 2564–2567, ISSN: 1558-4615.

[11] M. R. Ram, K. V. Madhav, E. H. Krishna, N. R. Komalla, and K. A. Reddy, "A Novel Approach for Motion Artifact Reduction in PPG Signals Based on AS-LMS Adaptive Filter," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 5, pp. 1445–1457, May 2012.

[12] T. Bhowmik, J. Dey, and V. N. Tiwari, "A novel method for accurate estimation of HRV from smartwatch PPG signals," in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Jul. 2017, pp. 109–112, iSSN: 1558-4615.

[13] K. Reddy, B. George, M. Mohan, and J. Kumar, "A novel calibration-free method of measurement of oxygen saturation in arterial blood," vol. 58, pp. 1699–1705.

[14] S. Babaeizadeh, R. E. Gregg, E. D. Helfenbein, J. M. Lindauer, and S. H. Zhou, "Improvements in atrial fibrillation detection for real-time monitoring," *Journal of Electrocardiology*, vol. 42, no. 6, pp. 522–526, Nov. 2009. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0022073609002507

[15] P. Ganesan, "Characterization of Cardiac Electrogram Signals During Atrial Fibrillation," Ph.D. dissertation, Jun. 2015.

[16] "Recommendations for the Standardization and Interpretation of the Electrocardiogram | Circulation." [Online]. Available: https://www.ahajournals.org/doi/10.1161/CIRCULATIONAHA.106.180200

[17] A. R. Pérez-Riera, R. Barbosa-Barros, R. Daminello-Raimundo, and L. C. de Abreu, "Main artifacts in electrocardiography," *Annals of Noninvasive Electrocardiology : The Official Journal of the International Society for Holter and Noninvasive Electrocardiology, Inc*, vol. 23, no. 2, Sep. 2017. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6931710/

[18] P. Singh, I. Srivastava, A. Singhal, and A. Gupta, "Baseline Wander and Power-Line Interference Removal from ECG Signals Using Fourier Decomposition Method," in *Machine Intelligence and Signal Analysis*, ser. Advances in Intelligent Systems and Computing, M. Tanveer and R. B. Pachori, Eds. Singapore: Springer, 2019, pp. 25–36.

[19] W. Dargie and J. Lilienthal, "Review of Motion Artifacts Removing Techniques for Wireless Electrocardiograms," in *2020 IEEE 23rd International Conference on Information Fusion (FUSION)*, Jul. 2020, pp. 1–8.

[20] J. Ottenbacher, M. Kirst, L. Jatobá, U. Großmann, and W. Stork, "An approach to reliable motion artifact detection for mobile long-term ECG monitoring systems using dry electrodes," in *IV Latin American Congress on Biomedical Engineering 2007, Bioengineering Solutions for Latin America Health*, ser. IFMBE Proceedings, C. Müller-Karger, S. Wong, and A. La Cruz, Eds. Berlin, Heidelberg: Springer, 2008, pp. 440–443.

[21] S. Kim, R. F. Yazicioglu, T. Torfs, B. Dilpreet, P. Julien, and C. Van Hoof, "A 2.4μA continuous-time electrode-skin impedance measurement circuit for motion artifact monitoring in ECG acquisition systems," in *2010 Symposium on VLSI Circuits*, Jun. 2010, pp. 219–220, iSSN: 2158-5636.

[22] J. Ottenbacher, M. Kirst, L. Jatoba, M. Huflejt, U. Grossmann, and W. Stork, "Reliable motion artifact detection for ECG monitoring systems with dry electrodes," in *2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Aug. 2008, pp. 1695–1698, iSSN: 1558-4615.

[23] L. N. Ribeiro, A. R. Hidalgo-Muñoz, and V. Zarzoso, "Atrial signal extraction in atrial fibrillation electrocardiograms using a tensor decomposition approach," in *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Aug. 2015, pp. 6987–6990, iSSN: 1558-4615.

[24] H. Moghaddasi, A.-J. van der Veen, N. M. S. de Groot, and B. Hunyadi, "Tensor-based Detection of Paroxysmal and Persistent Atrial Fibrillation from Multi-channel ECG," in *2020 28th European Signal Processing Conference (EUSIPCO)*, Jan. 2021, pp. 1155–1159, iSSN: 2076-1465.

[25] P. M. R. de Oliveira, V. Zarzoso, and C. A. R. Fernandes, "Coupled Tensor Model of Atrial Fibrillation ECG," in *2020 28th European Signal Processing Conference (EUSIPCO)*, Jan. 2021, pp. 915–919, iSSN: 2076-1465.

[26] P. M. R. de Oliveira and V. Zarzoso, "Löwner-Based Tensor Decomposition for Blind Source Separation in Atrial Fibrillation ECGs," in *2019 27th European Signal Processing Conference (EUSIPCO)*, Sep. 2019, pp. 1–5, iSSN: 2076-1465.

[27] M. Elgendi, "Optimal Signal Quality Index for Photoplethysmogram Signals," *Bioengineering*, vol. 3, no. 4, Sep. 2016. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5597264/

[28] E. Y. Ding, D. Han, C. Whitcomb, S. K. Bashar, O. Adaramola, A. Soni, J. Saczynski, T. P. Fitzgibbons, M. Moonis, S. A. Lubitz, D. Lessard, M. T. Hills, B. Barton, K. Chon, and D. D. McManus, "Accuracy and Usability of a Novel Algorithm for Detection of Irregular Pulse Using a Smartwatch Among Older Adults: Observational Study," *JMIR Cardio*, vol. 3, no. 1, p. e13850, May 2019. [Online]. Available: https://cardio.jmir.org/2019/1/e13850

[29] J. Lee, Y. Nam, D. D. McManus, and K. H. Chon, "Time-Varying Coherence Function for Atrial Fibrillation Detection," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 10, pp. 2783–2793, Oct. 2013.

[30] S. K. Bashar, D. Han, S. Hajeb-Mohammadalipour, E. Ding, C. Whitcomb, D. D. McManus, and K. H. Chon, "Atrial Fibrillation Detection from Wrist Photoplethysmography Signals Using Smartwatches," *Scientific Reports*, vol. 9, no. 1, p. 15054, Dec. 2019. [Online]. Available: http://www.nature.com/articles/s41598-019-49092-2

[31] D. Han, S. Bashar, F. Mohagheghian, E. Ding, C. Whitcomb, D. McManus, and K. Chon, "Premature atrial and ventricular contraction detection using photoplethysmographic data from a smartwatch," vol. 20, no. 19. [Online]. Available: https://escholarship.umassmed.edu/oapubs/4442

[32] H. Asada, H.-H. Jiang, and P. Gibbs, "Active noise cancellation using MEMS accelerometers for motion-tolerant wearable bio-sensors," in *The 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 1, pp. 2157–2160.

[33] T. Schäck, C. Sledz, M. Muma, and A. M. Zoubir, "A new method for heart rate monitoring during physical exercise using photoplethysmographic signals," in *2015 23rd European Signal Processing Conference (EUSIPCO)*, pp. 2666–2670, ISSN: 2076-1465.

[34] P. v. Gent, H. Farah, N. v. Nes, and B. v. Arem, "Analysing Noisy Driver Physiology Real-Time Using Off-the-Shelf Sensors: Heart Rate Analysis Software from the Taking the Fast Lane Project," *Journal of Open Research Software*, vol. 7, no. 1, p. 32, Oct. 2019. [Online]. Available: http://openresearchsoftware.metajnl.com/articles/10.5334/jors.241/

[35] J. L. Moraes, M. X. Rocha, G. G. Vasconcelos, J. E. Vasconcelos Filho, V. H. C. de Albuquerque, and A. R. Alexandria, "Advances in photopletysmography signal analysis for biomedical applications," vol. 18, no. 6.

[36] P. Boersma, "Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound," in *IFA Proceedings 17*, 1993, pp. 97–110.

[37] "Optimal Smoothers," in *Kalman Filtering*. John Wiley & Sons, Ltd, 2008, pp. 183–223. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/9780470377819.ch5

[38] I. Mh, L. Sy, P. Ts, K. Ts, C. Mh, and A. Yb, "Ballistocardiogram artifact removal from EEG signals using adaptive filtering of EOG signals," *Physiological measurement*, vol. 27, no. 11, Nov. 2006. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/17028414/

[39] D. P. Mandic, S. Kanna, and A. G. Constantinides, "On the intrinsic relationship between the least mean square and kalman filters [lecture notes]," *IEEE Signal Processing Magazine*, vol. 32, no. 6, pp. 117–122, 2015.

[40] "The discrete-time Kalman filter," in *Optimal State Estimation*. John Wiley & Sons, Ltd, 2006, pp. 121–148. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/0470045345.ch5

[41] M. Tarvainen, J. Hiltunen, P. Ranta-aho, and P. Karjalainen, "Estimation of nonstationary EEG with Kalman smoother approach: an application to event-related synchronization (ERS)," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 3, pp. 516–524, Mar. 2004.

[42] R. Yousefi, M. Nourani, S. Ostadabbas, and I. Panahi, "A Motion-Tolerant Adaptive Algorithm for Wearable Photoplethysmographic Biosensors," *Biomedical and Health Informatics, IEEE Journal of*, vol. 18, pp. 670–681, Mar. 2014.