



TRABALHO DE CONCLUSÃO DE CURSO

**Development of a Healthcare
Wearable Platform Using a
Photoplethysmography Sensor**

Gabriel Passos Moreira Pinheiro

Brasília, 19 de junho de 2018

UNIVERSIDADE DE BRASÍLIA

FACULDADE DE TECNOLOGIA

TRABALHO DE CONCLUSÃO DE CURSO

**Development of a Healthcare
Wearable Platform Using a
Photoplethysmography Sensor**

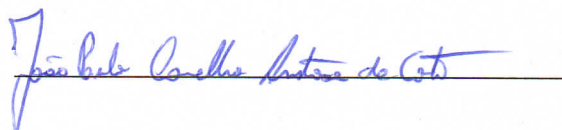
Gabriel Passos Moreira Pinheiro

*Trabalho de Conclusão de Curso submetido ao Departamento de Engenharia
Elétrica como requisito parcial para obtenção
do grau de Engenheiro Eletricista*

Banca Examinadora

Prof. João Paulo Carvalho Lustosa da Costa,
Dr.-Ing., ENE/UnB

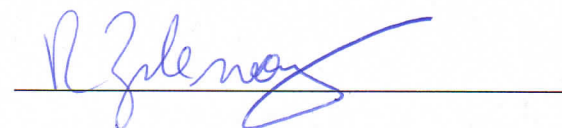
Orientador



Ricardo Kehrle Miranda, Dr.-Ing., ENE/UnB
Examinador interno



Prof. Ricardo Zelenovsky, Dr., ENE/UnB
Examinador interno



FICHA CATALOGRÁFICA

PINHEIRO, GABRIEL PASSOS MOREIRA

Development of a Healthcare Wearable Platform Using a Photoplethysmography Sensor [Distrito Federal] 2018.

xvi, 67 p., 210 x 297 mm (ENE/FT/UnB, Engenheiro, Engenharia Elétrica, 2018).

Trabalho de Conclusão de Curso - Universidade de Brasília, Faculdade de Tecnologia.

Departamento de Engenharia Elétrica

1. wearables

2. photoplethysmography

3. hear rate estimation

4. healthcare

I. ENE/FT/UnB

II. Título (série)

REFERÊNCIA BIBLIOGRÁFICA

PINHEIRO, G.P.M. (2018). *Development of a Healthcare Wearable Platform Using a Photoplethysmography Sensor*. Trabalho de Conclusão de Curso, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 67 p.

CESSÃO DE DIREITOS

AUTOR: Gabriel Passos Moreira Pinheiro

TÍTULO: Development of a Healthcare Wearable Platform Using a Photoplethysmography Sensor.

GRAU: Engenheiro Eletricista ANO: 2018

É concedida à Universidade de Brasília permissão para reproduzir cópias deste Trabalho de Conclusão de Curso e para emprestar ou vender tais cópias somente para propósitos acadêmicos e científicos. Os autores reservam outros direitos de publicação e nenhuma parte desso Trabalho de Conclusão de Curso pode ser reproduzida sem autorização por escrito dos autores.

Gabriel Passos Moreira Pinheiro

SQS 314 - Bl. F - Ap. 106

Asa Sul

CEP 70383-060 - Brasília - DF - Brasil

Dedicatória

A todos os engenheiros da família, em especial ao meu avô Trajano, pelo encorajamento à curiosidade e criatividade.

Gabriel Passos Moreira Pinheiro

Agradecimentos

Agradeço a Deus por todas as bênçãos e graças em minha vida. Agradeço aos meus pais, Alexandre e Thaís, pelo amor e apoio incondicional, e também por todos os conselhos e incentivo aos estudos. Agradeço também a minhas irmãs pelo carinho e companheirismo.

Agradeço ao Prof. João Paulo Carvalho Lustosa da Costa pela oportunidade da realização deste trabalho e por todo o apoio e confiança; ao Dr. Ricardo Kehrle Miranda por todo o auxílio e orientações; ao Juliano Barbosa Pretz pela parceria nos projetos de pesquisa que culminaram neste texto; a todos os meus professores de graduação pelos grandes ensinamentos; e a todos os muitos colegas de pesquisa durante a graduação pelas valiosas colaborações durante essa etapa acadêmica que se encerra.

Agradeço especialmente à Gabriela Araújo por todo o carinho e apoio durante a realização desse trabalho; ao Caio Fábio Oliveira por todo o auxílio; ao Lucas Lopes pela produção das imagens para abrilhantar o trabalho; ao Rafael Zamarion pela indicação de referência; ao Danilo Gameiro pela contribuição com equipamento para o bom andamento dos experimentos realizados. Também agradeço a meus colegas de curso pelo companheirismo nos momentos de seriedade e de descontração e por todo apoio.

Agradeço a toda a comunidade acadêmica pelos valiosos conhecimentos utilizados durante essa pesquisa, e à comunidade open-source pelas importantes contribuições aplicadas nesse trabalho.

Gabriel Passos Moreira Pinheiro

ABSTRACT

In order to implement a remote monitoring system for post-surgical patients using wearable devices, research on photoplethysmography measuring systems for heart beat detection is done so as to verify requirements for said systems and project processing algorithms for heart rate estimation. A performance comparison is carried out between implemented algorithms based on ESPRIT, FFT, autocorrelation, zero crossing and peak detection for evaluating the efficiency of each one under various operational conditions on a photoplethysmography sensor. Results are obtained for distinct values of LED current level, LED pulse width and sample rate. Additionally, a prototype for an online monitoring platform is proposed for health care data gathered with wearable devices. The implemented prototype contain features such as user accounts, chart visualization, text reminders and team sharing.

RESUMO

De modo a implementar um sistema de monitoramento remoto de pacientes pós-cirúrgicos com dispositivos vestíveis, é feito um estudo sobre sistemas de medição de batimentos cardíacos por fotoplethysmografia para verificar requisitos de tais sistemas e esquematizar algoritmos de processamento para estimação da frequência cardíaca. Uma comparação de desempenho é feita entre implementações de algoritmos baseados em ESPRIT, FFT, autocorrelação, cruzamento de zero e detecção de picos para avaliar a eficiência de cada um sob diferentes condições operacionais com um sensor de fotoplethysmografia. São obtidos resultados para diferentes valores de corrente de LED, largura de pulso do LED e frequência de amostragem. Ainda, é proposto um protótipo de plataforma online de monitoramento para dados de saúde aplicado à dispositivos vestíveis. Tal protótipo possui funcionalidades de contas de usuário, visualização de gráficos, lembretes de texto e compartilhamento com equipes.

SUMMARY

1	INTRODUCTION	1
1.1	MOTIVATION	1
1.2	OBJECTIVES	2
1.3	OUTLINE	2
1.4	RELATED WORK	3
2	HEART-RATE MEASURING	4
2.1	PROBLEM STATEMENT	4
2.2	NON-INVASIVE METHODS	4
2.2.1	ELECTROCARDIOGRAPHY	5
2.3	PHOTOPLETHYSMOGRAPHY	6
2.3.1	THE PPG SIGNAL	6
2.3.2	PPG MODES	9
2.3.3	MEASURING PPG	10
2.3.4	SOURCES OF NOISE IN PPG	12
3	HEART-RATE DETECTION ALGORITHMS	14
3.1	PROBLEM OUTLINE	14
3.2	ESPRIT	15
3.2.1	DEFINITION	15
3.2.2	COMPLEXITY ANALYSIS	17
3.3	FREQUENCY ANALYSIS USING FFT	18
3.3.1	DEFINITION	18
3.3.2	COMPLEXITY ANALYSIS	21
3.4	AUTOCORRELATION	22
3.4.1	DEFINITION	22
3.4.2	COMPLEXITY ANALYSIS	27
3.5	ZERO-CROSSING	27
3.5.1	DEFINITION	27
3.5.2	COMPLEXITY ANALYSIS	30
3.6	PEAK-DETECTION	30
3.6.1	DEFINITION	30
3.6.2	COMPLEXITY ANALYSIS	32
4	METHODOLOGY	33
4.1	PPG SENSOR	33
4.1.1	MAX30100	33

4.1.2	RCWL-0530	34
4.2	DATA ACQUISITION	34
4.2.1	ARDUINO PRO MINI	34
4.2.2	RECORDING DATA	35
4.3	EXPERIMENTAL SETUP	35
4.3.1	CIRCUIT ASSEMBLY	35
4.3.2	MEASUREMENTS REFERENCES	36
4.3.3	EXPERIMENTAL PROCEDURE	36
4.4	SIGNAL PREPROCESSING	36
4.4.1	MOVEMENT ARTIFACT REMOVAL	36
4.4.2	DC FILTERING	37
4.4.3	ADDITIONAL FILTERING	38
5	MONITORING PLATFORM	39
5.1	REQUIREMENTS	39
5.2	DEVELOPMENT OF THE APPLICATION	40
5.3	FEATURES	40
5.3.1	LOGIN AND REGISTRATION	41
5.3.2	PATIENTS LIST	42
5.3.3	NEWS FEED	43
5.3.4	USER PROFILE	43
5.3.5	TEXT REMINDERS	43
5.3.6	DATA VISUALIZATION	44
5.3.7	TEAM SHARING	44
5.4	FUTURE WORK	44
6	RESULTS AND DISCUSSION	54
6.1	LED CURRENT LEVEL	54
6.1.1	DISCUSSION	55
6.2	LED PULSE WIDTH	57
6.2.1	DISCUSSION	58
6.3	SAMPLE RATE	59
6.3.1	DISCUSSION	60
7	CONCLUSION	62
	BIBLIOGRAPHY	64
	APPENDICES	67

LIST OF FIGURES

2.1	An illustration of an ECG waveform and the main points of interest noted.	5
2.2	Diagram illustrating the absorption coefficients of the main components of human tissue (H_2O , Hb, HbO_2 , and Melanin) along the visible and near-infrared spectra, with the main wavelengths used for PPG (green, red and near-infrared) in evidence.	7
2.3	Illustration of a PPG pulse waveform, the main points are denoted along with their corresponding time of occurrence.	8
2.4	A schematic representation of both transmissive and reflective mode in PPG. The illustration shows pathways of light in both scenarios, as well as positioning of the devices and sources of ambient light noise.	9
2.5	A transimpedance amplifier construction, commonly used to interface current input from photodiodes and amplifying it to a voltage signal output.	11
2.6	A proposed topology for an active low-pass filter, allowing the PPG signal to be amplified while high-frequency noise is filtered. Component values are chosen according to the desired gain and bandwidth.	12
4.1	Circuit assembled for gathering PPG data from MAX30100, consisting of (from left to right) an YwRobot power source, a decoupling ceramic capacitor, an RCWL-0530 breakout board, an Arduino Pro and an FTDI breakout board. This illustration was produced using the software Fritzing.	35
4.2	Example of an unprocessed PPG signal recorded using the experimental setup. In red, the highlighted part is plotted separately in order to show the PPG waveform. In the detected signal, the valleys represent systolic peaks, followed by diastolic peaks and signal feet, which are the highest values of each cycle, as described in Subsection 2.3.1. Motion artifacts are visible as abrupt variations at the beginning and end of the signal.	37
5.1	Login component of the web application, where registered users can access their accounts.	41
5.2	New users must register to use the application, entering their information using this component.	45
5.3	To add new supervised patients, users must register their corresponding information as indicated in this component.	46
5.4	Patients supervised personally by users are displayed in a list from the most recently registered, allowing for users to access each patient's profile.	47
5.5	Patient profiles contain relevant information and a menu to access components to visualize health data, add a text reminder or share with a team.	48
5.6	The latest user activity is show in the news feed component from the most recent, informing of recently registered patients or a new text reminder.	49

5.7	Users' profiles allow editing account information and changing password.	49
5.8	Password can be updated by inputting the current password, choosing a new one and confirming it.....	50
5.9	Reminders are listed from the most recent, showing the included text information, date added and author. Clicking a reminder allows users to check exam values and to edit it.	50
5.10	Reminders are text messages that can also include exams results form BMP and CBC, allowing users to record information relevant to each patient.	51
5.11	Health data measured with wearable devices can be visualized through charts in this component. The prototype includes information for calories, steps, floors and hear rate.	52
5.12	Patients can be shared with the users' teams by adding them with this component. .	52
5.13	Users can access one of their teams profile by choosing it in this list, and also create a new team.	53
5.14	Team members can add new members through e-mail, change the team's name or delete it. Other members' contact information are also listed.	53
6.1	Estimated BPM values for each algorithm for different current level configurations. The reference BPM values are given by the oximeter during the start and at the end of the measurement.	55
6.2	Estimated BPM values for each algorithm for different current level configurations. The reference BPM values are given by the smartband during the start and at the end of the measurement.	56
6.3	Estimated BPM values for each algorithm for different pulse width and corresponding ADC resolution configurations. The reference BPM values are given by the oximeter during the start and at the end of the measurement.	58
6.4	Estimated BPM values for each algorithm for different pulse width and corresponding ADC resolution configurations. The reference BPM values are given by the smartband during the start and at the end of the measurement.	59
6.5	Estimated BPM values for each algorithm for different sample rate configurations, with the maximum ADC resolution available. The reference BPM values are given by the oximeter during the start and at the end of the measurement.	60
6.6	Estimated BPM values for each algorithm for different sample rate configurations, with the maximum ADC resolution available. The reference BPM values are given by the smartband during the start and at the end of the measurement.	61

LIST OF TABLES

6.1	Time and space complexities considered for each algorithm tested.	54
6.2	LED pulse width configurations tested and the implied ADC resolution.	57
6.3	Maximum available ADC resolution for each sample rate configuration tested.....	60

LIST OF ACRONYMS

AC	Alternating current
ADC	Analog-to-digital converter
BPM	Beats-per-minute
DC	Direct current
DFT	Discrete Fourier Transform
DSP	Digital signal processing
ECG	Electrocardiography
ESPRIT	Estimation of signal parameters by rotational invariance techniques
FFT	Fast Fourier Transform
FWHM	Full width at half maximum
GPB	Gain-bandwidth product
Hb	Deoxyhemoglobin
HbO ₂	Oxyhemoglobin
HR	Heart rate
HRV	Heart rate variability
IBI	Inter-beat intervals
IC	Integrated circuit
IoT	Internet of Things
IR	Infrared
ITU	Intensive treatment unit
LED	Light-emitting diodes
PCB	Printed circuit board
PPG	Photoplethysmography
PWM	Pulse width modulation
SpO ₂	Blood oxygen saturation
TIA	Transimpedance amplifier
UV	Ultraviolet
VCVS	Voltage-controlled voltage-source
WHO	World Health Organization

1 INTRODUCTION

1.1 MOTIVATION

Modern advancements in embedded systems technology allow for new applications in numerous fields. At the same time, breakthroughs in the Internet of Things (IoT) made possible for miniaturized and low-power sensors to be worn by users. Data can be then gathered with small form-factor components and transmitted wirelessly to computers, mobile devices or cloud servers. This technology has created additional requirements to enable and optimize use of such devices. Thus, new techniques in power saving, signal processing and wireless communications are being developed currently. One of the most recent epitomes of these technological advancements are wearable devices, gadgets that are worn by users. Available to be used on various parts of the body, such as wrists or forehead, wearables carry sensors that measure data and communicate via wireless protocols.

Wearable devices are ideal for constant and non-obtrusive monitoring. A large market for fitness wearable trackers formed in the last few years, with a global revenue of 21.66 billion dollars as of 2017, with projection of an increase to 93.69 billion dollars in 2027, according to [1]. These wearable devices are usually found in form of wristbands, being used seamlessly by users during physical activities and throughout their day. Users can then synchronize gathered data to their computers and smartphones to monitor information such as heartbeat, steps taken and calories spent during use of the device. General fitness information gathered in modern smartbands has been linked to improve users' commitment to their health [2]. Thus use of similar wearables directed towards healthcare applications is of interest.

In hospital environments, many patients are admitted to intensive treatment units (ITU) for emergence treatments and closer supervision. Having limited resources, a number of hospitals nationwide report usual lack of capacity for their ITU to admit new patients with various managements challenges [3]. This situation then creates obstacles to treat more serious cases and to admit new patients with urgent conditions. In the case of post-surgical cases, patients can be relocated from ITU to rooms or their homes, depending on their recovery conditions. Hence, by providing appropriate remote supervision, it is possible to unburden ITU and hospitals. Combining modern wearable technology with the given healthcare scenario, remote monitoring of post-surgical patients are of great significance for society.

On a personal scale, a real-time monitoring system allows medical professionals to assist multiple cases remotely. Doctors could then get simultaneous logs on their patients' conditions and focus their attention on more serious situations. Therefore, the health monitoring suite would consist of wearable devices – to gather data from patients – and a monitoring platform – delivering information to doctors and assistants. The devices would connect to the monitoring platform (such as a mobile application or an online website), allowing medics to check patients' health

information during the day and get alerts on anomalous events. Furthermore, doctors could interact with each other, collaborating in teams and leaving text reminders to each one regarding their shared patients. The link between wearable sensors and online platforms has the potential to further improve the effectiveness of remote monitoring of post-surgical patients.

1.2 OBJECTIVES

A study directed towards a non-obtrusive heart rate monitoring wearable device, comprising operational parameters and signal processing, is proposed. Alongside it, an online application for assessment of health-oriented data is developed. On the sensor hardware system, it is necessary to outline and validate optimal operational conditions in order to guarantee low power consumption and precise output data. Based on the experimental sensor measurements acquired with different operational parameters, digital signal processing is applied in order to filter and detect heart rate using different algorithms. Thus, precision and robustness are assessed for a number of methods.

The online platform is developed to enable monitoring of health information from multiple wearable devices users. It shall also enable collaboration among physicians, leaving text reminders in patients profiles and sharing them in teams.

1.3 OUTLINE

This work is presented in the following chapters:

Chapter 2 explores heart rate measuring in a wearable device context, analyzing various hardware requirements to achieve an appropriate system implementation. Chapter 3 then introduces frequency estimation algorithms directed towards heart rate detection. The characteristics of each algorithm is discussed comparatively, focusing on precision and feasibility in embedded systems. In sequence, Chapter 4 explains the experimental methodology to acquire, filter, process and analyze data from heart rate sensors. Also, comparative standards are defined to test suitability of each algorithm used to calculate heart rate.

Chapter 5 traces the technological and usability requirements for an online platform aimed at real-time monitoring of health conditions from users of wearable devices. Thus, a minimal viable prototype is designed to address initial specifications.

Chapter 6 presents experimental results of the project in order to give a deeper understanding of how each parameter affects the wearable system in terms of power and precision. Following, Chapter 7 concludes the work, discussing implications of results and proposing further development and improvement.

1.4 RELATED WORK

Signal processing techniques presented in this work have various related applications, as present in publications from the Laboratory of Array Signal Processing (LASP) in University of Brasília (UnB). The ESPRIT algorithm may be used for audio tampering detection and authentication for forensic analysis, as shown in [4] and [5] respectively, as well as applications in unmanned aerial vehicles (UAV), such as accurate attitude estimation presented in [6] and [7]. Additionally, a multiple-input and multiple-output (MIMO) architecture for use in mobile applications using wearable monitoring devices is proposed in [8], which is related to the monitoring platform build for this work.

2 HEART-RATE MEASURING

2.1 PROBLEM STATEMENT

General health of an individual is strongly related to its heart-rate, for instance [9] shows that mortality from both cardiovascular and non-cardiovascular causes generally increases with increasing heart rate. In specialized medical situations – for instance, post-surgical recovery – this information can be critical. Symptoms like tachycardia or sudden drops in heart rate can indicate serious health conditions or adverse effects of medication [10]. Therefore, constant monitoring of post-surgical patients' heart rate is a major necessity in ITU and hospitals. In many situations, the monitoring equipments are very costly and might not be available in sufficient quantities for all hospital inmates. Hence, these monitoring equipments can represent a strong burden on capabilities of hospitals to admit more patients.

Along with this context, different patients have varying needs of supervision for a successful recovery. Smaller surgical procedures usually require less time and medical attention than major interventions. Therefore, it is interesting to adapt the focus given to each case. Also, studies from the World Health Organization (WHO) show that more than 45% WHO Member States report to have less than 1 physician per 1000 population [11]. By choosing the correct approach to the needed supervision, it's possible to alleviate less serious patients of an uneasy stay in ITUs, simultaneously allowing physicians to pay closer attention to most severe cases.

The solution proposed in this work is to use wearable monitoring devices, that would drive down equipment costs and allow for remote medical supervision. Initially, it is imperative to discuss requirements for such devices and the approach necessary to achieve them. This chapter examines non-invasive methods of acquiring heart signals in embedded systems. In particular, the reflective photoplethysmography method has a deeper focus, as it is the better alternative for reasons shown next.

2.2 NON-INVASIVE METHODS

Medical procedures are deemed non-invasive when there is no break through the patient's skin and there is no contact with the mucosa or other unnatural body cavities [12]. This form of procedure is advantageous in a post-surgical context, since incisions and other invasive contacts should be avoided at all costs to prevent infections on recovering patients. The current main methods of visualizing a person's heart rate are electrocardiography (ECG or EKG) and photoplethysmography (PPG). Electrocardiography is briefly cited in the next subsection, showing characteristics and considerations made regarding this method during the present work.

2.2.1 Electrocardiography

The ECG was invented more than a century ago and is considered the gold standard for non-invasive heart diagnostics [13]. Since heartbeats are caused by natural electrical impulse in the heart's muscles, it is possible to measure the electrical activity of said heartbeat. An ECG is characterized by its waveform, in which each part of the signal is an electrical impulse resulting from muscle movements. In a simplified manner, as illustrated in Fig. 2.1, the heart's upper chambers make the first impulse – called a “P wave” – followed by a flat line when the electrical impulse goes to the bottom chambers. Next, the bottom chambers are contracted causing what is called a “QRS complex”. The last part of the signal is the “T wave”, representing a return to the resting state of the heart [14].

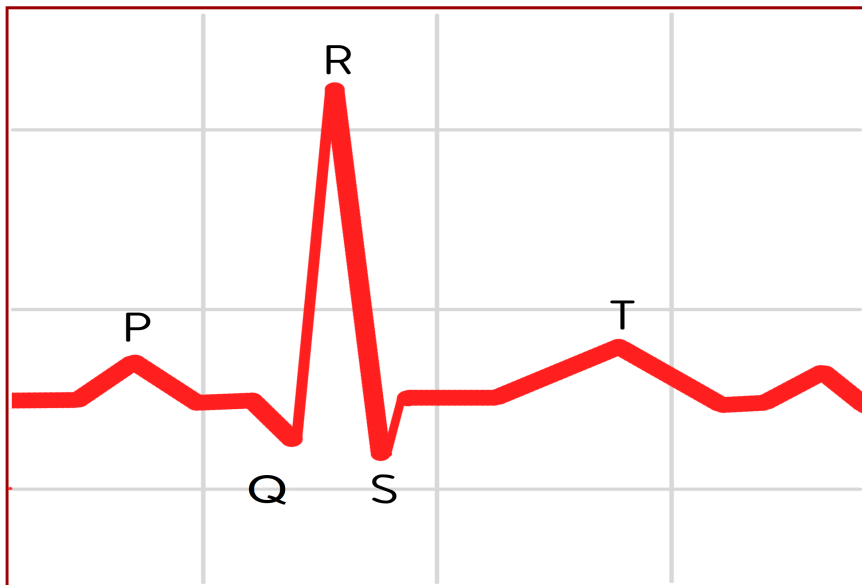


Figure 2.1: An illustration of an ECG waveform and the main points of interest noted.

The ECG exam is used to detect physical anomalies in the heart, e.g. the effects of drugs or devices used to control the heart, and also the size and position of heart's chambers [15]. Regarding heart rate, cardiac rhythm can be determined by the interval between two successive QRS complexes [16].

With all its advantages, the ECG is an indispensable test for correct diagnostics of heart dysfunctions and diseases. With that said, the standard 12-Lead ECG test is dependent on specialized equipment and the placement of electrical recording leads in various limbs of patients [17]. Therefore, ECG is mainly used in clinical settings and is rather unfit for daily or continuous usage under normal health circumstances. Hence, the PPG measurements are favored in wearable applications mainly due to its portability, as explained in the following section.

2.3 PHOTOPLETHYSMOGRAPHY

Photoplethysmography (PPG) is a technique that is used to detect blood volume changes in the microvascular bed of tissue [18]. In contrast to ECG, the PPG technique can be used to monitor heart rate non-invasively as well, but may use only one probe at the surface of the skin, leading to better acceptance of users for daily use. Based on the interaction between light and organic tissue, PPG measurements are achieved with simple constructions, such as an LED and a photodetector. Consequently, PPG has developed into a low-cost solution for primary care and specially for wearable health signal sensing. Its main applications are measuring heart rate, oxygen saturation, blood pressure and cardiac output [18].

2.3.1 The PPG signal

The interaction between light and skin tissue constituents – mainly water, melanin and blood tissue – can be absorptive, reflective or transmissive. These interactions are highly dependent on the wavelength used, due to tissue constituents' variant absorption and scattering coefficients. Relevant measured data and mathematical modeling of tissue interaction with light is shown in [19].

Given the theoretical background and relevant measurements cited, the main focus of this application is measuring blood volume changes related to heartbeats. Therefore, it is important to select the proper light signal used to probe the blood tissue. Light-emitting diodes (LED) are compact and reasonably low-powered devices that emit light. Their signal is distinguished by its wavelength and full width at half maximum (FWHM) parameter. Since light is an electromagnetic wave, its wavelength is given in terms of its frequency and the speed of light:

$$\lambda = \frac{c}{f}. \quad (2.1)$$

For describing light emissions in LEDs, it is preferred to use the central wavelength value λ_0 and its correspondent FWHM value. In photonics, the FWHM is given by the wavelength interval between the two extreme values of the light wave at which the its intensity is equal to half of its maximum value [20], centered in its central wavelength. Thus, the shorter FWHM is, sharper light wave emitted is.

Therefrom, investigating light-tissue interaction along the visible spectrum – but also including infrared (IR) and ultraviolet (UV) spectra – is fundamental. UV and longer IR light are strongly absorbed by water [18]. Simultaneously, melanin absorbs shorter wavelength more intensely, as presented in Fig. 2.2, and, since its concentration varies depending on skin pigmentation, different people will have distinct amounts of melanin absorption. Thus, an optical water window – as defined in [18] – exists in the interval that comprises red and near-infrared light, as is illustrated in Fig. 2.2 [21], allowing for a better quality of measurement in these wave-

lengths. However, it is interesting to note that an interval encompassed along green wavelengths has stronger absorption for blood tissues than that of melanin, displayed in Fig. 2.2, making green LED capable of measuring PPG signals correctly. Reflected green light has been shown to have an advantages over reflected infrared light [21], being a appropriate alternative depending on the application needed. However, since infrared light is invisible to the human eye, it is not perceived by users, as opposed to green light that can become uncomfortable.

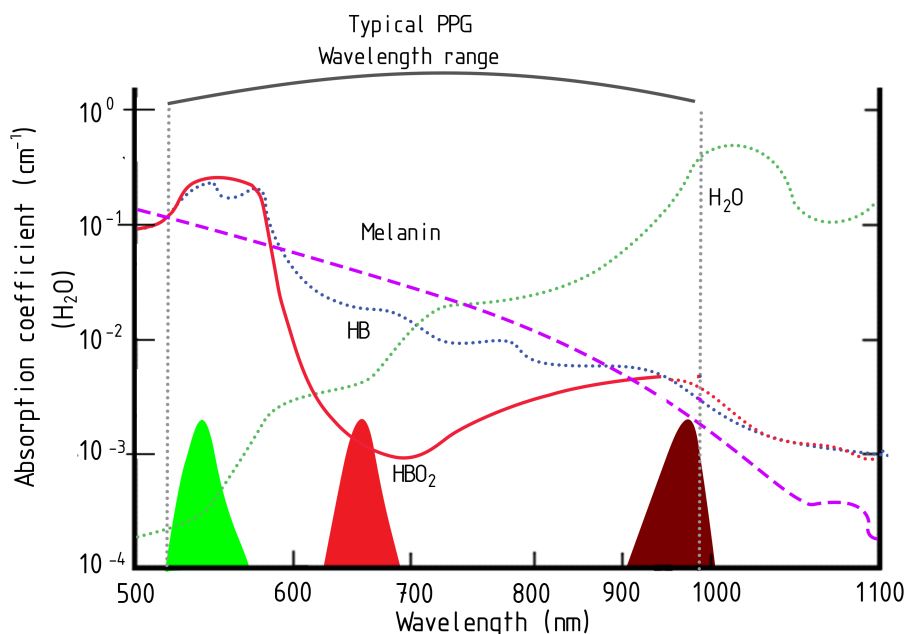


Figure 2.2: Diagram illustrating the absorption coefficients of the main components of human tissue (H_2O , Hb, HbO_2 , and Melanin) along the visible and near-infrared spectra, with the main wavelengths used for PPG (green, red and near-infrared) in evidence.

Following, it is significant to discuss differences in light absorption between oxygen-saturated hemoglobin (oxyhemoglobin or HbO_2) and desaturated hemoglobin (deoxyhemoglobin or Hb), which are also illustrated in Fig. 2.2. By binding itself to oxygen molecules during respiration, hemoglobin alters its chemical characteristics, showing stronger absorption coefficient for near-infrared (~ 940 nm) than for red wavelengths (~ 660 nm). On deoxyhemoglobin though, the inverse is observed, with stronger absorption in the red spectrum [19]. Hence, PPG is also capable of determining blood oxygen saturation in what is called pulse oximetry by combining red and infrared light emitters. Even though it is an interesting feature for a health-oriented wearable device, it is outside the scope of this projects, thus only the heart rate sensing application is considered.

The PPG waveform constitutes of a pulsating AC component – attributed to cardiac synchronous changes in the blood volume with each heart beat – superimposed by a slowly varying DC component with various lower frequency components attributed to respiration, sympathetic nervous system activity and thermoregulation [18].

The format of a PPG pulse is defined as two phases, as illustrated in Fig. 2.3 [22]: the anacrotic phase (beginning at the rising edge of the pulse at $T_{SP/2}$), and the catacrotic phase (until

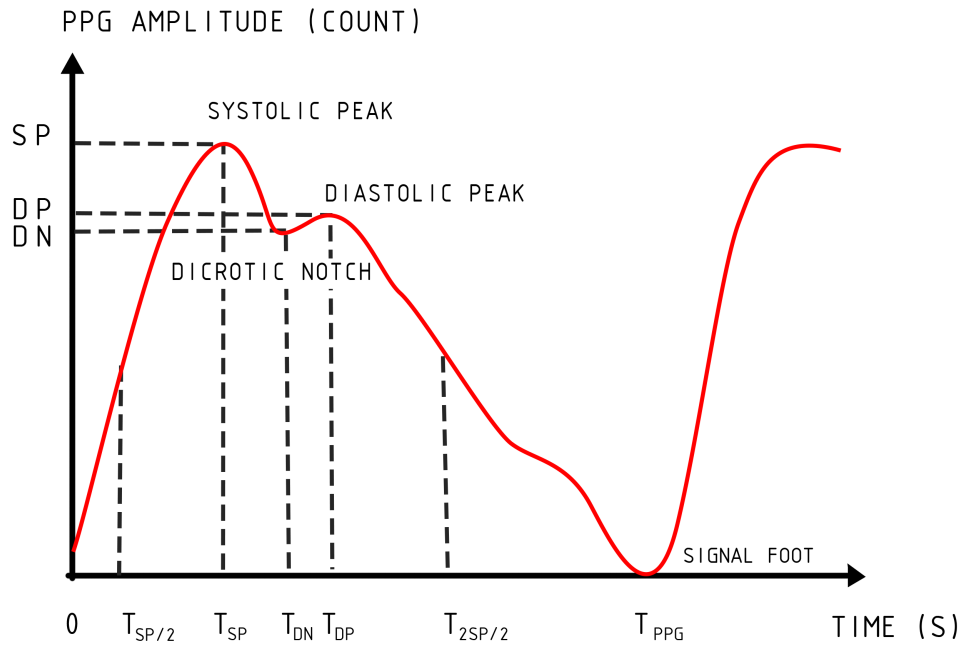


Figure 2.3: Illustration of a PPG pulse waveform, the main points are denoted along with their corresponding time of occurrence.

the falling edge of the pulse at $T_{2SP/2}$). The first phase is primarily due to systole, and the second phase results from diastole and wave reflections from the periphery [18]. In the former phase, the point of interest is a systolic peak (at T_{SP}), and in the latter phase, a diastolic peak is present at T_{DP} . Also, a dicrotic notch, at T_{DN} , is usually seen in between heartbeat phases of subjects with healthy compliant arteries [18]. A peak-to-peak interval, or the distance between two consecutive systolic peaks, represents a complete heart cycle [22]. A pulse may also be considered to start at the signal foot shown in Fig. 2.3. The systolic amplitude indicates the changes in blood volume caused by arterial blood flow and has been related to stroke volume, which could indicate vasoconstriction or vasodilation; it can also reflect cold sensitivity, reaction to anesthetics, blood loss, hypothermia, etc. Pulse width and area also correlate with the systemic vascular resistance, which could reflect drug interaction of the patient or blood viscosity [22].

An important aspect to consider is that the PPG wave is measured as the blood volume changes. Light detected by PPG systems is correspondent to the light emitted subtracted from the absorbed portion. Additionally, an increase in blood volume results in more light absorbed. Thus, the detected PPG signal corresponds to the inverse of blood volume changes, with measured waveform being a flipped version the one presented in Fig. 2.3.

2.3.2 PPG Modes

PPG measuring devices may function in two modes – reflection and transmission modes – depending on the placement of the photodetector in relation to the light emitter. If both are placed in the same side of a surface (e.g. the skin), it works on reflection mode. In this configuration, the photodiode captures light scattered back and reflected from the tissue. On the other side, if the photocoupler is placed on opposite sides of a surface (e.g. across a finger), transmission mode is attained [21]. Both mechanisms are illustrated in Fig. 2.4 [21]. The LED emits light through the skin tissue with added background light interference. Transmission mode detects light that passes through the tissue unabsorbed and its photodetector receives more ambient light interference. Light that is reflected or scattered back in the skin is detected in reflectance mode, also subject to ambient light interference.

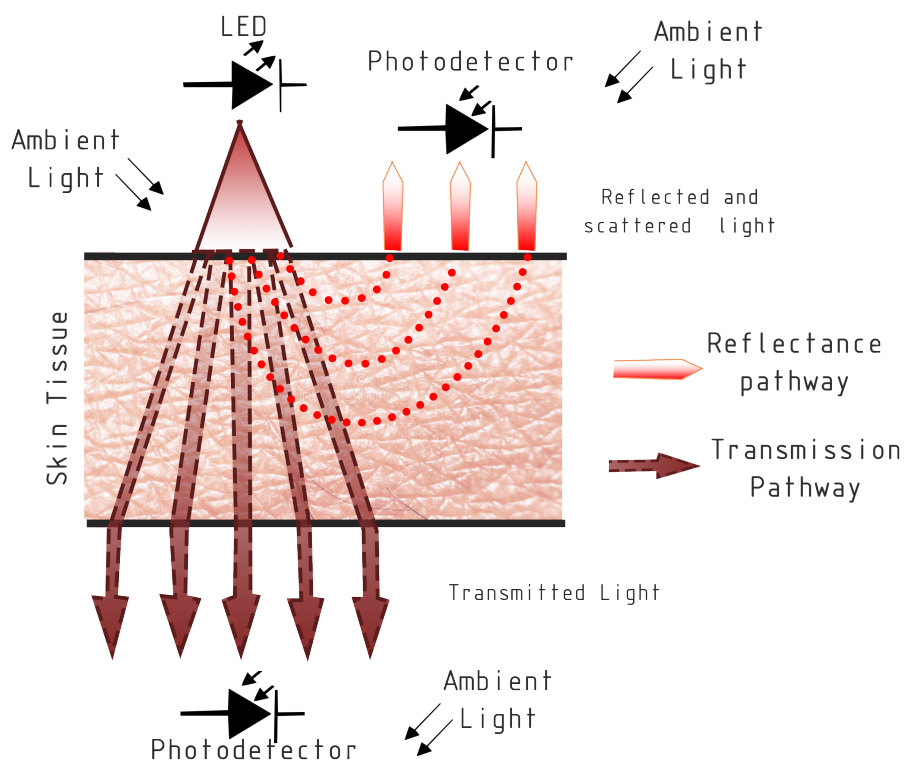


Figure 2.4: A schematic representation of both transmissive and reflective mode in PPG. The illustration shows pathways of light in both scenarios, as well as positioning of the devices and sources of ambient light noise.

Although transmission mode generally obtains an appropriate signal, it can may be limited to fewer measurement sites. This is due to the fact that the sensor must be located on the body at a site where transmitted light can be readily detected, such as the fingertip, nasal septum, cheek, tongue, or earlobe [23]. Another possible disadvantage of transmission mode sensors is the use of either a cuff or a clip to attach the sensor to the body. This causes a probe-dependent increase in transmural pressure, potentially causing vasoconstriction in the veins, degrading signal quality [21].

Reflection mode, on the other hand, may be attached to more sites on the body. Thus, placing

the PPG wearable devices in unobtrusive spots, such as at user's wrist, is possible. Nonetheless, reflection mode presents some disadvantages as well. Mainly, it is affected by motion artifacts – due to physical activity, for instance – and pressure disturbances – for example, variations in the contact force between the PPG sensor and measurement site [23].

Both PPG modes are relevant depending on the application, contemplating aspects such as measurement site, device build and user acceptance. All things considered, reflection mode is deemed more versatile for daily-use wearable, since it can be placed in unobtrusive sites and does not necessarily depend on cuffs or clips for probing.

2.3.3 Measuring PPG

By focusing on the simple and low-cost solution for building a PPG device, using at least one LED and a photodetector, this section determines further requirements for this system. Additional treatment of the measured signal, including filtering and amplification, is discussed in richer detail during this subsection. LEDs are highly efficient semiconductor devices with operation based on injection electroluminescence, a more extensive description is available at [24]. Photodiodes are commonly used to detect light signals in PPG systems. These devices are semiconductor junctions that rely on photogenerated charge carriers [24].

In order to correctly operate LEDs in PPG systems, it is important to add a proper driver circuit. Various LED drivers are available in the market as integrated circuits. Also, it is possible to build driver circuits using regular discrete components. Important characteristics of an LED driver are good power regulation and stable sourced current. As another key characteristic, the LED driver circuit must generate minimal noise, as any noise inside the signal bandwidth will degrade the overall performance [25]. Additionally, features such as adjustable current values and possibility of pulsed operation allow for further optimizations in power consumption, which is key in wearable devices due to battery size and power limitations.

As for photodiode interfacing, it is generally achieved by using a transimpedance amplifier (TIA). Such amplifiers take current input signals and output a correspondent amplified voltage signal. Thus, by connecting the photodiode as shown in Figure 2.5, in what is called photovoltaic mode, the small photocurrent generated into a short circuit is amplified by the TIA as a voltage signal with gain given by Equation (2.2) [26]. The 3 dB frequency of TIA is given in Equation (2.3) where GBP is the gain-bandwidth product of the operational amplifier. The key specifications for the TIA are extremely low input current, input current noise, and input voltage noise, as well as high-voltage operation [25].

$$V_{\text{out}} = R_f I_P, \quad (2.2)$$

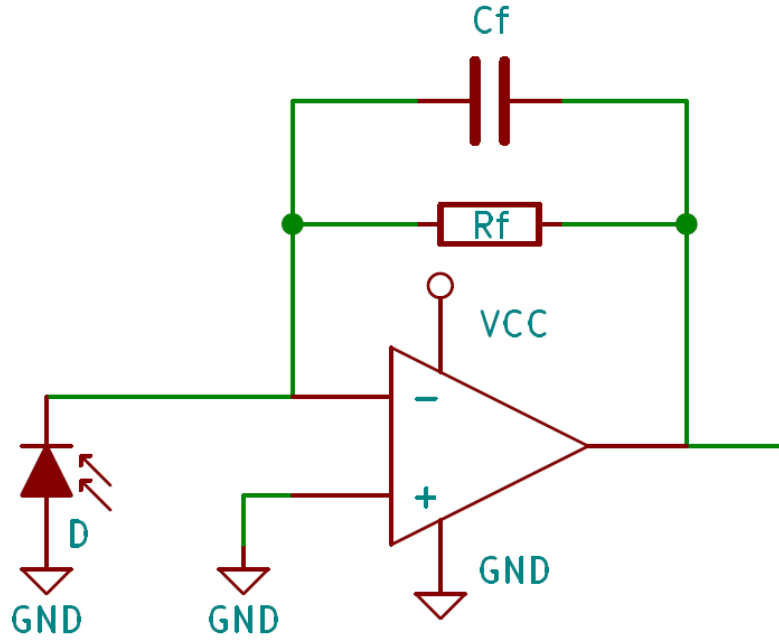


Figure 2.5: A transimpedance amplifier construction, commonly used to interface current input from photodiodes and amplifying it to a voltage signal output.

where I_P is the photodiode input current, R_f the feedback resistor and V_{out} the output voltage.

$$f_{3\text{dB}} = \sqrt{\frac{\text{GBP}}{2\pi R_f (C_f + C_D + C_{\text{TIA}})}}, \quad (2.3)$$

where, additionally, C_f is the feedback capacitor, C_D is the photodiode capacitance and C_{TIA} is the amplifier's capacitance.

After the PPG signal passes through the TIA, it is advisable to add further stages of filtering and amplification. Since the heart signal comprises of a short bandwidth in low frequencies (not more than 5 Hz), active low-pass filtering provides noise reduction and signal amplification. As an illustrative purpose, a voltage-controlled voltage-source (VCVS) filter is presented next. The circuit topology, shown in Figure 2.6, presents a non-inverting voltage gain greater than unity equals to K in Figure 2.6 [26]. The filter can be realized as Butterworth, Bessel or Chebyshev filters depending on the application, each pole of the filter is represented by one stage of a VCVS. Given as an example here, filters and amplifiers similar to the VCVS can be used for this application. As a relevant note, the DC component of the PPG signal is not necessary for heart rate detection, which only depends on the pulsating AC component. However, the DC signal cannot be filtered if pulse oximetry is carried out in parallel to heart rate measurements, due to its dependence on this part of the PPG signal.

Finally, the amplified signal must then be sampled by an analog-to-digital converter (ADC) to be digitally processed. The key specifications for the ADC are high resolution, signal-to-noise

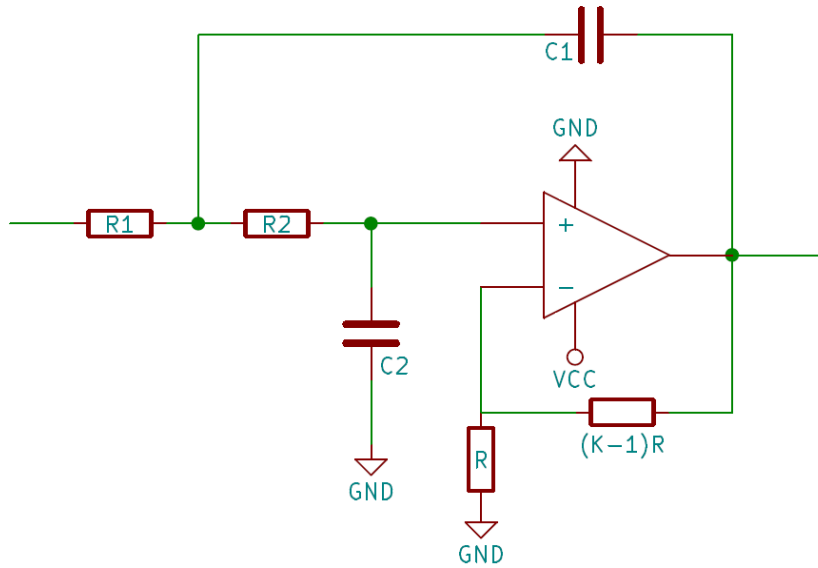


Figure 2.6: A proposed topology for an active low-pass filter, allowing the PPG signal to be amplified while high-frequency noise is filtered. Component values are chosen according to the desired gain and bandwidth.

ratio (SNR) and short acquisition time [25]. Such ADCs are present in microcontrollers or as separate components.

The complete set of the components described above is commonly referred as an analog front end (AFE) and is available as chipsets encompassing the necessary devices and additional digital bus interfaces for measurement reading, for instance.

2.3.4 Sources of noise in PPG

Noise is a prevalent aspect when dealing with signal processing. In wearable PPG sensors, many sources of noise are present, hence a brief analysis regarding this parameter is proposed in this subsection.

In PPG sensor systems, noise arise from either external or internal sources. The most significant external noise source is ambient light. Background ambient noise can be specially harmful to mid and far IR systems due to thermal radiation emitted by objects at room temperature [21]. Crosstalk in systems consisting of multiple LEDs is also present.

As for internal sources of noise, that is, present in the detection and amplification circuitry, the following list is presented as a summary of relevant factors [24]:

- Photon Noise: arises from the inherent uncertainty of the incident mean photon flux Φ , where this quantity fluctuates randomly;
- Photoelectron noise: occurs due to the probabilistic nature of the photogeneration of charge carriers, according to the probability of a photoevent to happen given the detector's quantum

efficiency η ;

- Gain noise and receiver circuit noise: random thermal noise added by amplifier and receiving circuitry;

By using appropriate signal processing techniques, ambient background light can be satisfactorily deducted from the PPG signal. Also, it is possible to isolate the PPG sensors from external light and from adjacent light emitters by building appropriate enclosures. On the other side, internal sources can only be reduced by low-noise circuit construction techniques, as photoelectric limitations are inherent to the system used. Thus, filtering processes have to take into account the various sources of noise present in PPG systems.

3 HEART-RATE DETECTION ALGORITHMS

3.1 PROBLEM OUTLINE

In digital signal processing (DSP), it is a common objective for frequency estimation of digitalized signals, with applications spanning from communication networks to medical imaging. Thus, many algorithms for frequency estimation have been well established up until the present time. Fundamentally, for application in this project, heart rate detection can be formulated as a frequency estimation problem. Heart rate is usually reported in beats-per-minute (BPM). Thus, by measuring the frequency parameter of a periodic heart signal, heart-rate is given by:

$$f_{\text{heart}} \cdot 60 = \text{BPM}. \quad (3.1)$$

As explained in section 2.3.1, the PPG signal is composed of both an pulsating AC component and a slowly varying DC component. Again, for HR detection, only the former (AC) part of the signal is necessary, with the latter (DC) part considered noise for all practical purposes. Therefore, the PPG signal may be considered periodic, with period equals to N and $60/N$ corresponds to the BPM value. By then sampling the PPG signal through an ADC for further digital processing, the result is a discrete-time signal $x(n)$ with an N amount of samples, and $x(n) = 0$ if $n < 0$ or $n \geq N$. The sampling rate f_s is equipment-dependent and must be taken into account. This concludes that the algorithms selected must be defined in terms of discrete signals.

HR, measured as beat-to-beat intervals, is not constant and varies in time. This property is known as heart rate variability (HRV) [27]. A noteworthy simplification is the disregard of said Heart Rate Variability (HRV) or Inter-beat intervals (IBI) in the measuring timeframe. Hence, the PPG signal is considered to have a single main frequency in a given timeframe. This simplification results no loss for HR detection in the application studied, only being relevant in other specialized utilizations.

This chapter explores five major algorithms for frequency estimation with different complexities implemented using MATLAB®(source code available in Appendix I), them being:

- ESPRIT (*Estimation of Signal Parameters by Rotational Invariance Techniques*);
- Frequency Analysis using FFT (*Fast Fourier Transform*);
- Autocorrelation;
- Zero crossing;
- Peak detection.

This comparative analysis is inserted in the wearable technology context. Therefore, a number of limitations are present, such as reduced processing capabilities and low memory storage. It is interesting then to assert the trade-offs between each algorithms, considering the demanded implementation costs and overall precision for the applications that each one offers. This chapter presents the theoretical basis of all the algorithms considered in this work. Selecting the best suited algorithm can be done through theoretical methodology – presented in this chapter – as well as experimental investigation – elucidated in the next chapter.

3.2 ESPRIT

3.2.1 Definition

Based on a technique of signal subspace decomposition, the Estimation of Signal Parameters by Rotational Invariance Techniques (ESPRIT) algorithm is an advanced method that can be used for frequency estimation. The algorithm exploits a property of rotational invariance of signal subspaces spanned by two temporally displaced data sets [28]. A simplified description of the least-squared version of ESPRIT algorithm is shown next based on [29].

Considering a single complex exponential $s_0(n) = \alpha e^{j2\pi f n}$ with complex amplitude α and frequency f , said signal has the following property:

$$s_0(n+1) = \alpha e^{j2\pi f(n+1)} = s_0(n) e^{j2\pi f n}, \quad (3.2)$$

thus the next sample is equal to a phase-shifted version of the current sample. This can be viewed as a rotation on the unit circle $e^{j2\pi f}$. Thus, by representing a signal composed of the complex exponentials $s(n)$ with added noise component $w(n)$ in a time-window vector model given in [29]:

$$\mathbf{x}(k) = \sum_{p=1}^P \alpha_p \mathbf{v}(f_p) e^{-j2\pi n f_p} + \mathbf{w}(n) = \mathbf{V} \Phi^n \alpha + \mathbf{w}(n) = \mathbf{s}(\mathbf{n}) + \mathbf{w}(n) \in \mathbb{C}^M, \quad (3.3)$$

where the P columns of matrix \mathbf{V} are length- M time-window frequency vectors of the complex exponentials

$$\mathbf{V} = [\mathbf{v}(f_1) \quad \mathbf{v}(f_2) \quad \dots \quad \mathbf{v}(f_P)] \in \mathbb{C}^{M \times P}. \quad (3.4)$$

Vector α contains the amplitudes of complex exponentials α_p and matrix Φ is a diagonal matrix, also known as the rotational matrix, containing the phase shifts between adjacent time

samples of $s(n)$:

$$\mathbf{\Phi} = \text{diag}\{\phi_1, \phi_2, \dots, \phi_P\} = \mathbf{W}_N = \begin{bmatrix} e^{j2\pi f_1} & 0 & \dots & 0 \\ 0 & e^{j2\pi f_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & e^{j2\pi f_P} \end{bmatrix}, \quad (3.5)$$

where $\phi_p = e^{j2\pi f_p}$ for $p = 1, 2, \dots, P$. Knowing that the frequencies of the complex exponentials f_p completely describe the rotation matrix $\mathbf{\Phi}$, estimates of frequency can be obtained by finding $\mathbf{\Phi}$. This is done by using the ESPRIT technique as explained in [29]. Once the rotation matrix $\mathbf{\Phi}$ is found, the frequency estimator is given by:

$$\hat{f}_p = \frac{\angle \phi_p}{2\pi}. \quad (3.6)$$

The technique for calculating the frequency estimations is summarized next, as of [30]. Initially, consider a data matrix $\mathbf{X} \in \mathbb{C}^{N \times M}$ containing N data records of the length- M time-window vector signal $\hat{\mathbf{x}}(n)$, an analytical representation of signal $s(n)$ defined previously, thus:

$$\mathbf{X} = [\mathbf{x}(0) \quad \mathbf{x}(1) \quad \dots \quad \mathbf{x}(N-1)]^T \in \mathbb{C}^{M \times N}, \quad (3.7)$$

where $\mathbf{x}(n) = [\hat{x}_a(n) \hat{x}_a(n+1) \dots \hat{x}_a(n+M-1)]^T$, and $()^T$ is the transposition operator of matrices.

By then using the Eigenvalue Decomposition (EVD) method on the sample covariance matrix $\mathbf{R}_x = \mathbf{X}\mathbf{X}^T/N$, corresponding to the correlation matrix of the recorded signal \mathbf{X} in (3.7), it is obtained that:

$$\mathbf{R}_x = \mathbf{U}\mathbf{\Sigma}\mathbf{U}^H, \quad (3.8)$$

where \mathbf{U} is an $M \times M$ matrix of right singular vectors and $()^H$ denotes the Hermitian operator. Both matrices are unitary, given that $\mathbf{U}^H\mathbf{U} = \mathbf{I}$, where \mathbf{I} is the identity matrix. Matrix $\mathbf{\Sigma}$ has dimensions $N \times M$ and is composed of singular values on the main diagonal ordered in descending magnitude.

Matrix \mathbf{U} can be decomposed as $\mathbf{U} = [\mathbf{u}_s | \mathbf{U}_{\text{noise}}]$, where \mathbf{u}_s , the first column of \mathbf{U} , is the vector that generates the signal subspace, of dimensions $M \times 1$, formed by the singular vector corresponding to the maximum singular value of the data matrix \mathbf{X} . The remaining singular vectors form a matrix in which its columns correspond to the basis that generate the noise subspace $\mathbf{U}_{\text{noise}}$, of dimensions $M \times (M-1)$, orthogonal to the signal subspace.

By writing vectors \mathbf{u}_u and \mathbf{u}_d formed by the first and last $M-1$ elements of \mathbf{u}_u , respectively, the rotational invariance presented previously, and exploited by ESPRIT, guarantees that:

$$\mathbf{u}_u \mathbf{\Phi} = \mathbf{u}_d, \quad (3.9)$$

where Φ corresponds to the rotation matrix. By solving 3.9, the phase value is given by:

$$\angle \phi_p = \angle \frac{\mathbf{u}_u^H \mathbf{u}_d}{\mathbf{u}_u^H \mathbf{u}_u}, \quad (3.10)$$

hence the frequency estimator is calculated using Equation (3.6).

From this summary, the ESPRIT-based algorithm for heart rate detection is formulated [30]:

Algorithm 1: ESPRIT-based heart rate estimator

Given signal $x(n)$, sampled with frequency f_s , during a time window T do:

- 1) Compute the correlation data matrix estimate \mathbf{R} between the signal $x(n)$ and $\lfloor \frac{T \cdot f_s}{2} \rfloor$ derived using nonwindowed data.
- 2) Calculate by eigendecomposition matrix \mathbf{E} , whose columns are the corresponding right eigenvectors of \mathbf{R} .
- 3) Determine the number of rows of \mathbf{R} .
- 4) Decompose \mathbf{R} in the corresponding EVD matrices.
- 5) Derive the signal subspace from matrix \mathbf{E} .
- 6) Estimate the rotation matrix Φ from the EDV matrices and the signal subspace.
- 7) Calculate the ϕ eigenvalues of Φ .
- 8) Determine the frequency estimator by computing the angle of eigenvalues ϕ :

$$\hat{f}_p = \frac{\angle \phi}{2\pi} \cdot f_s$$

- 9) Calculate the estimated BPM value form \hat{f} :

$$\text{estimated BPM} = \hat{f} \cdot 60$$

3.2.2 Complexity analysis

The upper-bound time and space complexity analysis are presented next, in accordance to the steps enlisted in Algorithm 1.

3.2.2.1 Time complexity

Step 1) calculates the data matrix estimate \mathbf{R} , which corresponds to the correlation of size- N signal $X(n)$ through a $N \times N$ matrix multiplication, which corresponds to complexity $\mathcal{O}(N^3)$.

Step 2) through 7) calculates the EVD of matrix \mathbf{R} , the signal subspace and eigenvalues of the rotation matrix Φ , which involve matrix multiplication, accounting for a $\mathcal{O}(N^3)$ total complexity. Steps 7) and 8) are direct calculations, and are thus $\mathcal{O}(1)$. Overall, the upper time bound for Algorithm 1 is $\mathcal{O}(N^3)$.

3.2.2.2 Space complexity

To simplify the analysis, the largest data matrix size is taken as the upper bound for space in Algorithm 1. The data matrix estimate \mathbf{R} consists of N data records of the length- M time-window vector signal, as postulated in the definition of ESPRIT. This brings the space complexity of Algorithm 1 to be $\mathcal{O}(NM)$, which depends on input size N and time window size M .

3.3 FREQUENCY ANALYSIS USING FFT

3.3.1 Definition

Frequency analysis is a powerful tool in digital signal processing. In order to use this technique, its necessary to represent a discrete-time signal – which is the case of a sampled PPG signal – in its frequency domain representation. Given a time-discrete signal $x(n)$, instead of calculating the continuous Fourier transform $X(\omega)$, it is more convenient to sample its spectrum. This representation results in a Discrete Fourier Transform (DFT), as demonstrated in [31]. Given a periodic finite-duration sequence $x(n)$, with length N , its DFT is given by:

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-j2\pi kn/N}. \quad (3.11)$$

The direct computation of the DFT for a real arbitrary signal is computationally demanding. Consider alternative representation of the DFT:

$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{kn}, \quad k = 0, 1, \dots, N-1, \quad (3.12)$$

in which W_N is as defined in Equation (3.11):

$$W_N = e^{-j2\pi/N}. \quad (3.13)$$

By representing Equation (3.12) as a linear transformation on $x(n)$, it is possible to visualize its computational complexity. For there, an N -point vector x_N corresponding to the sequence $x(n)$ is defined, as well as the corresponding N -point vector X_N for the frequency samples and an $N \times N$ matrix W_N as follows:

$$\mathbf{x}_N = \begin{bmatrix} x(0) \\ x(1) \\ \vdots \\ x(N-1) \end{bmatrix}, \quad \mathbf{X}_N = \begin{bmatrix} X(0) \\ X(1) \\ \vdots \\ X(N-1) \end{bmatrix}, \quad (3.14)$$

$$\mathbf{W}_N = \begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & W_N & W_N^2 & \dots & W_N^{N-1} \\ 1 & W_N^2 & W_N^4 & \dots & W_N^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & W_N^{N-1} & W_N^{2(N-1)} & \dots & W_N^{(N-1)(N-1)} \end{bmatrix}.$$

From this definitions, the N -point DFT is given by:

$$\mathbf{X}_N = \mathbf{W}_N \cdot \mathbf{x}_N. \quad (3.15)$$

By taking the dimensions of matrix \mathbf{W}_N , the reason for direct computation of the DFT being computationally inefficient can be verified. Mainly, the direct DFT calculation does not benefit from the periodicity and symmetry of the phase factors W_N , namely:

$$\text{Symmetry property:} \quad W_N^{k+N/2} = -W_N^k, \quad (3.16)$$

$$\text{Periodicity property:} \quad W_N^{k+N} = -W_N^k. \quad (3.17)$$

Thus, by taking a divide-and-conquer approach to calculating the DFT, a better family of algorithms is achieved, know as the Fast Fourier Transform (FFT) algorithms. The approach of the FFT is to decompose the N -point DFT into smaller DFTs, exploiting the symmetry (3.16) and periodicity (3.17) properties. Many different realizations of the FFT exist, such as the Radix-2 algorithms, presented on [31].

By using the frequency domain representation of a PPG signal, it is possible to determine the dominant frequency estimator of the signal by selecting the maximum argument of the absolute value of the FFT across half the frequency range [32]:

$$\hat{f} = \frac{1}{N'} \arg \max_{-N'/2 \leq m \leq N'/2} \left| \sum_{n=0}^{N-1} x(n) e^{-j2\pi mn/N'} \right|^2. \quad (3.18)$$

Specifically for the application of processing PPG signals, it is important to reinforce that the measurement contain a strong DC component, that is, low-frequency noise due to reasons explained in 2.3.1. Thus, correct filtering of low and high frequency noise is fundamental for the correct application of the FFT algorithm.

Another important aspect is the use of zero padding to increase the resolution of the signal's FFT. By padding the finite-duration signal of length L with $N - L$ zeros at the end and computing the result's FFT, the frequency resolution is increased [31]. Hence, this allows for higher precision when selecting the dominant frequency estimator through Equation (3.18), especially in the PPG signal's narrow frequency band. This, however, increases the computational effort of calculating the FFT for introducing additional points but no extra information, whatsoever.

From the above definitions and considerations, the FFT heart rate estimator algorithm is described as follows:

Algorithm 2: FFT-based heart rate estimator

Given signal $x(n)$, sampled with frequency f_s , during a time window T do:

1) Form the zero-padded signal by appending $N - T \cdot f_s$ zeros at its end, totalizing N points. The achieved frequency resolution is given by:

$$\Delta R = \frac{f_s}{N}$$

2) Calculate the N -point FFT of the signal and its corresponding frequency arguments.

- Calculate the absolute value of the frequency domain representation:

$$|X(f)| = \left| \sum_{n=0}^{N-1} x(n) e^{-j2\pi f n / N} \right|$$

- Calculate the vector f of length N corresponding to frequency values from 0 to f_s :

$$f = [0, f_s/N, 2f_s/N, \dots, f_s]$$

3) Compute the frequency argument corresponding to maximum amplitude of the FFT from half the spectrum $0 \leq k \leq N/2$:

$$\hat{f} = \arg \max_{0 \leq n \leq N/2} \left| \sum_{n=0}^{N-1} x(n) e^{-j2\pi m n / N} \right|$$

4) Calculate the estimated BPM value form \hat{f} :

$$\text{estimated BPM} = \hat{f} \cdot 60$$

3.3.2 Complexity analysis

The upper-bound time and space complexity analysis are presented next, in accordance to the steps enlisted in Algorithm 2, based on FFT.

3.3.2.1 Time complexity

Step 1) takes $N - L$ steps to zero pad the size- L input, hence it is $\mathcal{O}(N - L)$. Step 2) calculates the FFT for the zero padded input of size N , computing $(N/2)\log_2 N$ complex multiplications, according to [31]. Thus, step 2) is upper-bounded by $\mathcal{O}(N \log N)$. Then, step 3) computes the maximum value of the resulting N -point FFT for half of its frequency spectrum, thus having complexity $\mathcal{O}(N)$. Step 4) is a direct calculation of the BPM value, corresponding to $\mathcal{O}(1)$. Therefore, the algorithm is time bounded by step 3), making its time complexity of class $\mathcal{O}(n \log n)$.

3.3.2.2 Space complexity

Aside from the size L of the input, the Algorithm 1 has the following auxiliary space complexity: step 1) turns the input into a size N zero-padded array, thus it is $\mathcal{O}(N)$; step 2) can store in its worst case scenario a total of $(N/2)\log_2 N$ data points corresponding to its complex multiplications performed, making it $\mathcal{O}(N \log N)$; steps 3) and 4) store each one single variable – corresponding to the frequency of maximum FFT amplitude and estimated BPM value – making both a $\mathcal{O}(1)$ complexity. Hence, the space complexity of Algorithm 2 is upper bounded by $\mathcal{O}(n \log n)$.

3.4 AUTOCORRELATION

3.4.1 Definition

Correlation is used to measure how similar a signal is to another. By taking two real signal sequences $x(n)$ and $y(n)$, their crosscorrelation is defined as in [31]:

$$r_{xy}(l) = \sum_{n=-\infty}^{\infty} x(n) y(n-l), \quad l = 0, \pm 1, \pm 2, \dots \quad (3.19)$$

The index l corresponds to a time shift (or lag) parameter and the subscripts x, y on the cross correlation sequence $r_{xy}(n)$ indicates the sequences being correlated. In the special case in which $y(n) = x(n)$, an autocorrelation is performed, as defined by the sequence:

$$r_{xx}(l) = \sum_{n=-\infty}^{\infty} x(n) x(n-l), \quad l = 0, \pm 1, \pm 2, \dots \quad (3.20)$$

When calculating the autocorrelation of a finite duration signal, it is accepted to define the limits of the summation as those of the finite sequence.

$$r_{xx}(l) = \sum_{n=l}^{N-|k|-1} x(n) x(n-l), \quad l = 0, \pm 1, \pm 2, \dots \quad (3.21)$$

Some important properties of correlation are shown next. Initially, considering two finite-energy sequences $x(n)$ and $y(n)$, forming a linear combination:

$$ax(n) + by(n-l), \quad (3.22)$$

where a and b are arbitrary constants and l is a lag introduced. By calculating the combination's energy, it results in:

$$\begin{aligned} \sum_{n=-\infty}^{\infty} [ax(n) + by(n-l)]^2 &= a^2 \sum_{n=-\infty}^{\infty} x^2(n) + b^2 \sum_{n=-\infty}^{\infty} y^2(n-l) \\ &\quad + 2ab \sum_{n=-\infty}^{\infty} x(n) y(n-l) \\ &= a^2 r_{xx}(0) + b^2 r_{yy}(0) + 2abr_{xy}(0), \end{aligned} \quad (3.23)$$

It is noted that $r_{xx}(0) = E_x$ and $r_{yy}(0) = E_y$, correspondent to the energies of $x(n)$ and $y(n)$, respectively. Thus, it is concluded that:

$$a^2 r_{xx}(0) + b^2 r_{yy}(0) + 2abr_{xy}(0) \geq 0. \quad (3.24)$$

Equation (3.24) can be treated as quadratic of coefficients $r_{xx}(0)$, $2r_{xy}(l)$ and $r_{yy}(0)$. Since Equation (3.24) shows that it is nonnegative, then its discriminant must be nonpositive, thus:

$$4[r_{xy}^2(l) - r_{xx}(0)r_{yy}(0)] \leq 0. \quad (3.25)$$

From then, the following inequality is satisfied:

$$|r_{xy}(l)| \leq \sqrt{r_{xx}(0)r_{yy}(0)} = \sqrt{E_x E_y}. \quad (3.26)$$

For the particular case in which $y(n) = x(n)$, 3.26 is given by:

$$|r_{xx}(l)| \leq r_{xx}(0) = E_x. \quad (3.27)$$

From 3.27, it is affirmed that the autocorrelation sequence of a signal reaches its peak value at zero lag, given that the signal matches itself with zero shift. Therefore, $r_{xx}(0)$ is the upper bound of the autocorrelation sequence and this property must be taken into account while designing a

correlation algorithm.

A further property of interest is the relation of signal scaling and autocorrelation. Since the autocorrelation sequence does not varies with signal amplitude – only scaling its values accordingly –, it is often desirable to normalize the autocorrelation sequence by dividing it by its upper bound, namely $r_{xx}(0)$. Thus, the normalized autocorrelation sequence is defined:

$$\rho_{xx}(l) = \frac{r_{xx}(l)}{r_{xx}(0)}, \quad -1 \leq \rho_{xx}(l) \leq 1. \quad (3.28)$$

Additionally, from the definition of autocorrelation in 3.20, it is possible to demonstrate using symmetry of the lag shifts l that:

$$r_{xx}(l) = r_{xx}(-l). \quad (3.29)$$

Concluding that the autocorrelation sequence is an even function, consequently it is sufficient to compute $r_{xx}(l)$ for $l \geq 0$.

Applying the above definitions in the heart rate estimation case, it is assumed that the PPG signal is periodic. The definition of autocorrelation for periodic signals becomes:

$$r_{xy}(l) = \lim_{M \rightarrow \infty} \frac{1}{2M+1} \sum_{n=-M}^M x(n) y(n-l), \quad (3.30)$$

corresponding to the autocorrelation of power signals. For the particular case which the periodic sequences $y(n) = x(n)$, both with period N , the average over the infinite interval indicated by 3.31 is equal to the averages over a single period, reducing the definition to:

$$r_{xx}(l) = \frac{1}{N} \sum_{n=0}^{N-1} x(n) x(n-l), \quad (3.31)$$

in which the factor $\frac{1}{N}$ can be viewed as a normalization scale factor.

Applying the above definitions in the heart rate estimation case, the main interest in calculating the lags correspondent to the maximum values of the autocorrelation sequence. Because the PPG signal is periodic – and its period is related to the BPM values, as show in Section 3.1 –, the lag between two subsequent autocorrelation peaks is equal to the signal's period. From an arbitrary periodic signal $x(n)$ with period N :

$$\begin{aligned}
r_{xx}(l = N) &= \frac{1}{N} \sum_{n=0}^{N-1} x(n) x(n - N), \\
&\text{since } x(n) = x(n - N) \\
&\text{thus } r_{xx}(N) = \frac{r_{xx}(0)}{N}.
\end{aligned} \tag{3.32}$$

Equation (3.32) is valid for any harmonics $kN, k = 0, \pm 1, \pm 2, \dots$, corresponding to peaks of the autocorrelation sequence. Thus, it is possible to calculate the period of the PPG signal by computing the lag shift between autocorrelation peaks. From then, the calculation of BPM is direct.

Another relevant characteristic of the autocorrelation is its robust noise resistance. Given the received signal affected by an additive random interference factor $w(n)$ added to the arbitrary signal $x(n)$, of unknown period N , given by:

$$y(n) = x(n) + w(n). \tag{3.33}$$

Observing M samples of $y(n)$, given $0 \leq n \leq M-1$ where $M \gg N$. Thus, it can be assumed that $y(n) = 0$ for $n < 0$ and $n > M$. The autocorrelation of $y(n)$, with the normalization factor of $1/M$, is:

$$r_{yy}(l) = \frac{1}{M} \sum_{n=0}^{M-1} y(n) y(n - l). \tag{3.34}$$

Substituting 3.33 into 3.34, results in:

$$\begin{aligned}
r_{yy}(l) &= \frac{1}{M} \sum_{n=0}^{M-1} [x(n) + w(n)][x(n - l) + w(n - l)] \\
&= \frac{1}{M} \sum_{n=0}^{M-1} x(n) x(n - l) \\
&\quad + \frac{1}{M} \sum_{n=0}^{M-1} [x(n) w(n - l) + x(n - l) w(n)] \\
&\quad + \frac{1}{M} \sum_{n=0}^{M-1} w(n) w(n - l) \\
&= r_{xx}(l) + r_{xw}(l) + r_{wx}(l) + r_{ww}(l).
\end{aligned} \tag{3.35}$$

From Equation (3.4.1), the first factor represents the autocorrelation sequence of $x(n)$. Be-

cause $x(n)$ is periodic, the autocorrelation sequence presents the same periodicity, containing peaks at lags equal to 0 and to harmonics of N . Yet, the peaks' amplitudes decay as the lag shift l approaches M , since the products $x(n)x(n-l)$ become zero outside the bounds of the sampled signal. Thus, it is recommended to avoid calculating $r_{yy}(l)$ for large values of l .

The remaining factors of 3.4.1 are two crosscorrelations between $x(n)$ and the random interference $w(n)$ ($r_{xw}(l)$ and $r_{wx}(l)$) and an autocorrelation of the interference $r_{ww}(l)$. Since $x(n)$ and $w(n)$ are expected to be totally uncorrelated, the additive crosscorrelation factors are insignificant. The interference's autocorrelation is expected to have a peak at $l = 0$, as shown in 3.27, but decays fast for random interference is predictably not correlated to itself. Therefore, only the autocorrelation of $x(n)$ is expected to have significant peaks at $l \geq 0$, allowing the signal to be detected in the presence of random interference.

From this theoretical background, an algorithm to calculate BPM from a periodic PPG signal with N samples, sampled with frequency f_s is proposed:

Algorithm 3: Autocorrelation-based heart rate estimator

Given signal $x(n)$, sampled with frequency f_s , during a time window T do:

- 1) Compute the autocorrelation of the signal for $l \geq 0$:

$$r_{xx}(l) = \sum_{n=0}^{T \cdot f_s} x(n) x(n-l), \quad l = 0, 1, 2, \dots,$$

- 2) For $l \neq 0$, compute the lag corresponding to the maximum argument of $r_{xx}(l)$:

$$l_p = \arg \max_{l \geq 0} (r_{xx}(l)); \tag{3.36}$$

- 3) Calculate the frequency estimator \hat{f} from l_p :

$$\hat{f} = \frac{1}{l_p}; \tag{3.37}$$

- 4) Calculate the estimated BPM value from \hat{f} :

$$\text{estimated BPM} = \hat{f} \cdot 60. \tag{3.38}$$

3.4.2 Complexity analysis

The upper-bound time and space complexity analysis are presented next, in accordance to the steps enlisted in Algorithm 3, based on autocorrelation.

3.4.2.1 Time complexity

The step 1) calculates the autocorrelation of a size- N input by computing N multiplications, considering only lags greater than 0, making its upper bound equals to $\mathcal{O}(N)$. Step 2) computes the maximum value of the resulting N multiplications, thus having a time complexity of $\mathcal{O}(N)$ as well. Steps 3) and 4) are direct computations, resulting in a complexity of $\mathcal{O}(1)$ for both. Hence, the time complexity of Algorithm 3 is $\mathcal{O}(n)$.

3.4.2.2 Space complexity

Additionally to the space complexity of the N -size input, Algorithm 3 has its auxiliary space complexity given by: step 1) stores up to N variables for the lag values and the corresponding N multiplication results, characterizing a complexity of $\mathcal{O}(2N)$; step 2), 3) and 4) store a single variable each – corresponding to the lag of maximum argument, the estimated frequency and the resulting BPM value – making it all of them $\mathcal{O}(1)$. Thus, Algorithm 3 has a space complexity of class $\mathcal{O}(n)$.

3.5 ZERO-CROSSING

3.5.1 Definition

Zero-crossing detection is a common method for measuring the frequency of a periodic signal [33], which is applicable to HR estimation. For discrete time signals, it is possible to define zero-crossing as a function based on conditions such as those in [34], which is equal to one if the sequence $x(n)$ (indexed by an arbitrary sample $n \geq 2$) unequivocally crossed the zero-axis:

$$z(x) = \begin{cases} 1 & \text{if } x_n < 0 \text{ and } x_{n-1} < 0 \\ 1 & \text{if } x_n > 0 \text{ and } x_{n-1} < 0 \\ 1 & \text{if } x_n > 0, x_{n-1} = 0 \text{ and } x_{n-2} < 0 \\ 1 & \text{if } x_n < 0, x_{n-1} = 0 \text{ and } x_{n-2} > 0 \\ 0, & \text{otherwise.} \end{cases} \quad (3.39)$$

Two approaches can be taken, counting the amount of zero crossings in a given interval or calculating the time difference between sequences of two zero crossings. From the most basic

case, considering a sine wave with average equal to 0 in its period, it is known that during one period, the function is equal to 0 twice. Alternatively, the period of the signal is also given by the time delay between two zero crossings intercalated by another for a time frame longer than one single period.

By treating the PPG signal properly, it is possible to apply principles considered previously to estimate its frequency. Thus, in a given measurement time frame T , considering the time-discrete signal $x(n)$ sampled with frequency f_s is defined for $0 \leq n < T \cdot f_s$. Using the definition of 3.39, the frequency using the counting method is given by:

$$f_{\text{heart}} = \frac{\sum_{n=0}^{T \cdot f_s - 1} z(x(n))}{2}. \quad (3.40)$$

The definition for the time interval method, considering z_m as the position of samples in which a zero crossing occurred, is given by:

$$f_{\text{heart}} = [(z_{m+2} - z_m) \cdot f_s]^{-1}. \quad (3.41)$$

The BPM value can then be calculated directly using Equation (3.1).

It is then possible to construct an algorithm from the definitions above to calculate BPM from the PPG signal. Some conditions for its functioning, though, are presumed and must be taken into account for the design of signal treatment filters. Initially, it is assumed that the signal contains no DC bias. Presence of additional DC bias might render the algorithm conditions useless if there are no zero-crossings in a given time frame. Thus, proper high-pass filtering or other DC blocking treatment is necessary to be applied in the sampled PPG signal.

Other important aspect is that of multiple outputs for a single zero crossing due to noise, resulting in inaccurate results. In order to mitigate this, [33] proposes two approaches: rule based and digital filtering. Rule based design eliminates events that don't meet expected timing requirements. This approach inhibits zero crossing detection for a specified period after the first event, relying on the statistical probability that the next zero crossing will be close to the next half period. The threshold timing for inhibiting zero crossing detection must be designed in order to allow natural variations in the input signal. The other approach of Digital filtering has the ability to discriminate events based upon frequency of occurrence just as passive filtering but overcomes their disadvantages by having high accuracy and predictability [33].

The alternative considered will be the simpler zero crossing counting approach, defined next:

Algorithm 4: Frequency estimation by counting zero crossings

Given signal $x(n)$, sampled with frequency f_s , during a time window T do:

1) Compute the number of zero-crossings in the signal during T according to:

$$z(x) = \begin{cases} 1 & \text{if } x_n < 0 \text{ and } x_{n-1} < 0 \\ 1 & \text{if } x_n > 0 \text{ and } x_{n-1} < 0 \\ 1 & \text{if } x_n > 0, x_{n-1} = 0 \text{ and } x_{n-2} < 0 \\ 1 & \text{if } x_n < 0, x_{n-1} = 0 \text{ and } x_{n-2} > 0 \\ 0, & \text{otherwise.} \end{cases}$$

$$\text{zero-crossings} = \sum_{n=0}^{T \cdot f_s} z(x(n))$$

2) Divide the number of zero crossings by during T by 2;

3) Calculate the frequency estimator from half the number of zero crossings during T :

$$\hat{f} = \frac{\text{zero-crossings}/2}{T};$$

4) Calculate the estimated BPM value form \hat{f} :

$$\text{estimated BPM} = \hat{f} \cdot 60. \quad (3.42)$$

A alternative implementations can also be explored, such as using the derivative of the signal. Since the signal is a periodic function, therefore it can be written as a Fourier Series:

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos \frac{n\pi x}{L} + b_n \sin \frac{n\pi x}{L}. \quad (3.43)$$

By differentiating the signal in relation to time:

$$f(x) = \frac{n\pi x}{L} \left(\sum_{n=1}^{\infty} b_n \cos \frac{n\pi x}{L} - a_n \sin \frac{n\pi x}{L} \right). \quad (3.44)$$

Which results in a scaled periodic signal with the same frequency as the original one and a phase difference of 90° . Since phase and amplitude information are irrelevant for heart-rate measurement, the frequency parameter is kept and can be estimated by the zero crossing algorithm in the same manner as described above. Note that the DC bias – represented by a_0 – vanishes when applying differentiation. Therefore, this approach can be fruitful due to the requirement of DC blocking posed above. However, it is not considered in this work due to the additional filtering introduced by the derivative, which poses a possibly unfair advantage.

3.5.2 Complexity analysis

The upper-bound time and space complexity analysis are presented next, in accordance to the steps enlisted in Algorithm 4, based on zero crossings counting.

3.5.2.1 Time complexity

The comprised steps in Algorithm 4 are: step 1), that calculates the amount of zero crossings of the size- N signal by inspecting it in its complete length, thus is as $\mathcal{O}(N)$ process.; steps 2), 3) and 4) are direct calculations, thus making all of them $\mathcal{O}(1)$. So, the upper bound of time complexity of Algorithm 4 is $\mathcal{O}(n)$.

3.5.2.2 Space complexity

Auxiliary space complexity – which excludes mandatory N -size array allocation for the input – is given in accordance to Algorithm 4. Step 1) allocates a counter variable corresponding to the amount of zero crossings of the input, and is then $\mathcal{O}(1)$. Step 2) computes half the amount of a variable, and is also $\mathcal{O}(1)$. Finally, steps 3) and 4) are also single variable calculations, making them both $\mathcal{O}(1)$. Therefore, the space complexity considered for Algorithm 4 is $\mathcal{O}(1)$.

3.6 PEAK-DETECTION

3.6.1 Definition

Considering the waveform of a PPG signal exposed in Section 2.3.1, the correspondent definition of a heartbeat (complete heart cycle) is the interval between two consecutive systolic peaks. Thus, by detecting the number of peaks in a given interval of the sampled signal, it is possible to determine the heart-rate.

Two definitions of peak are relevant for an arbitrary function $x(t)$, the first being in terms of absolute amplitude in a given time interval T (3.45), and the second in terms of the signal's derivative (3.46):

$$\text{peak} = \max(x(t)), 0 \leq t \leq T, \quad (3.45)$$

$$x'(t) = 0, \quad \text{and} \quad x'(t - \Delta t) > 0, \quad x'(t + \Delta t) < 0, \quad \text{then } x(t) \text{ is a peak.} \quad (3.46)$$

Thus, by applying these definition as algorithms, the procedure is to checks either the maximum amplitude in a heart-beat impulse (between two signal's feet) or zero-crossings of its derivative. By taking into consideration the reasons given in Section 2.3.1, it is fundamental to flip the detected PPG signal along the x-axis if it is intended to detect systolic peaks. Disadvantages of amplitude detection are misreadings due to fluctuations caused by noise or slight body movement and the definition of the time interval T in which the maximum value of the function is verified, which must be accounted for and thus properly treated. As for derivative checking, disadvantages of the zero-crossing algorithm are introduced (e.g., sensibility to high-frequency noise).

Another problem of both algorithms is that additional local peaks, such as the weaker diastolic peak, may also be detected causing inaccurate readings. Therefore, it is viable to set a threshold that guarantees only systolic peaks are detected, or to account for two peaks (systolic and diastolic) in a given heartbeat interval. The algorithm proposed uses a direct computation of PPG signal feet, thus making it unnecessary to flip the signal and only one foot is expected in each heartbeat.

The peak detection counting algorithm is defined as follows:

Algorithm 5: Peak detection using maximum value in a given interval

Given signal $x(n)$, sampled with frequency f_s , during a time window T do:

- 1) Find the number of local peaks, in order to detect only the PPG signal's feet during interval T ;
- 2) Calculate the frequency estimator from the sample interval:

$$\hat{f} = \frac{\text{number of peaks}}{T};$$

- 3) Calculate the estimated BPM value form \hat{f} :

$$\text{estimated BPM} = \hat{f} \cdot 60.$$

Alternatively, since systolic peaks occur after the fastest variation in the signal, it is possible to

detect amplitude of the derivative instead of the signal itself. Thus, peaks in the derivative give a slightly delayed version of the systolic peaks, since the former is followed by the latter. This also results in a proper heart-rate measurement, as shown in the property of derivatives in Equation (3.44). This algorithm is achieved by inputting the derivative of the signal to Algorithm 5 instead of the signal itself.

3.6.2 Complexity analysis

The upper-bound time and space complexity analysis are presented next, in accordance to the steps enlisted in Algorithm 5, based on zero crossings counting.

3.6.2.1 Time complexity

Step 1) registers the peak locations of the N -size input, thus making this process $\mathcal{O}(N)$. Next steps, 3) and 4) are direct calculations of complexity $\mathcal{O}(1)$. Hence, the time complexity of Algorithm 5 is $\mathcal{O}(n)$.

3.6.2.2 Space complexity

The considered auxiliary space complexity, not taking into account the N -size array allocation for the input, is calculated for Algorithm 5. Step 1) logically marks the location of pulse peaks along the signal's length, which is of complexity $\mathcal{O}(N)$. Step 2) and step 3) and 4) allocate one variable for the frequency estimator and one to the BPM values, respectively, resulting in complexity $\mathcal{O}(1)$ for both. The derived space complexity for Algorithm 5 is $\mathcal{O}(n)$.

4 METHODOLOGY

In order to apply the processing techniques explored in the previous chapter, it is necessary to assemble a suitable circuit to save data from a photoplethysmography (PPG) sensor. Then, the appropriate filtering and calculations are applied to the captured heart signal. Each algorithm is treated as an estimator for the real measured heart-rate value, referred market-available devices. The resulting data is presented and analyzed in the next chapter.

The component selected for this project is the MAX30100 pulse oximeter and heart-rate sensor integrated circuit (IC). The device bought comes in a breakout board model RCWL-0530, containing all the necessary peripherals to operate the sensor. In order to serve as a communication bridge between the sensor I²C bus and the computer's serial port, an Arduino Pro Mini board is used. Raw data from the sensor is captured with a Python script and processed using the software MATLAB®.

4.1 PPG SENSOR

4.1.1 MAX30100

The MAX30100 pulse oximeter and heart-rate sensor IC used in this project is produced by Maxim Integrated. The component comes in a small form factor (5.6mm x 2.8mm x 1.2mm) and is integrated with the following components (which are specified in the correspondent datasheet [35]):

- infrared LED;
- red LED;
- photodiode;
- LED driver;
- ambient light cancellation filter;
- analog-digital converter (ADC);
- internal temperature sensor;
- proprietary digital filter;
- I²C communication interface.

The LEDs integrated allows for measuring both heart-rate (using infrared only) and pulse oximetry (both red and infrared). The latter parameter (oximetry) is interesting for a complete health-oriented wearable device, but it's analysis and implementation is outside the scope of this project. The IC also comes integrated with the so-called analog front-end – consisting of amplifiers, filters, ADC and digital communication interface – reducing the need for additional circuitry. Thus, this device serves the purpose of a PPG measuring system established in Chapter 2. Additional parts are necessary for power sourcing the sensor with appropriate voltage levels, I²C bus interfacing and prototyping with the surface-mounting device.

4.1.2 RCWL-0530

Since faster prototyping facilitates development of the project, the MAX30100 sensor was bought in the breakout board RCWL-0530, produced by the company RCWL. The board contains the MAX30100 biosensor soldered and all the necessary peripherals for operation, such as a 1.8 V and a 3.3 V voltage regulators, I²C pull-up resistors and pin connectors for protoboard use in its printed circuit board (PCB).

4.2 DATA ACQUISITION

Readings from the MAX30100 are stored in data registers of the device and can be read through the I²C bus. To save data from the sensor to a computer for further processing, it is necessary to communicate with the MAX30100 and the computer simultaneously. For this application, an Arduino Pro Mini is used to bridge the I²C bus to an USB interface used by desktop computers.

4.2.1 Arduino Pro Mini

The Arduino platform is an open-source project that started in 2009 with the objective of bringing the electronics hobby to a wider audience. The platform has become widely used prototyping with its accessible Integrated Development Environment (IDE), cheap breakout boards based on ATMEL AVR chips and user-contributed libraries.

The selected board for the project application is the Arduino Pro Mini, based on the ATmega328P chip and running at 8 MHz on 3.3 V. The board contains pins that can be configured as SDA and SCL ports for I²C communication, as well as separate ports RX and TX for serial communication. An USB interface is provided additionally for flashing and communicating with the Arduino board using a FTDI breakout board.

4.2.2 Recording data

The software used to interface the Arduino board with MAX30100 is freely available on GitHub as open-source [36]. The code enables the Arduino Pro Mini to access raw data from the sensor readings, transmitting them over serial bus to the computer. The code was modified to output data in a convenient form, separating readings of the IR and red LEDs, together with the milliseconds of each reading since the system started running, separated by commas. Thus, by connecting it the computer via USB, a Python script monitors the serial port Arduino uses and logs the data to a .csv (comma separated values) file.

4.3 EXPERIMENTAL SETUP

4.3.1 Circuit assembly

The test circuit was assembled on a 830 tie points breadboard. The Figure 4.1 illustrates the layout using the software Fritzing [37], which is described next. A breadboard-mountable power supply set at 3.3 V, produced by YwRobot, is used along a decoupling ceramic capacitor between the positive power supply and reference point. This power source was used to power both the Arduino and the MAX30100 breakout board at 3.3 V. The SDA and SCL lines of the I²C bus were connected to pins A4 and A5 of the Arduino Pro Mini (according to its datasheet [38]) and the corresponding terminals of the MAX30100. For interfacing with a computer (for firmware programming and data communication), the Arduino Pro Mini was connected to the FTDI breakout board according to the Serial communication specification, that is, TX terminal of one device connected to RX of the other, for both terminals. Communication with a computer is then established by using an USB cable. The circuit illustration produced using the open-source Fritzing is presented next, using additional open-source parts form [39]:

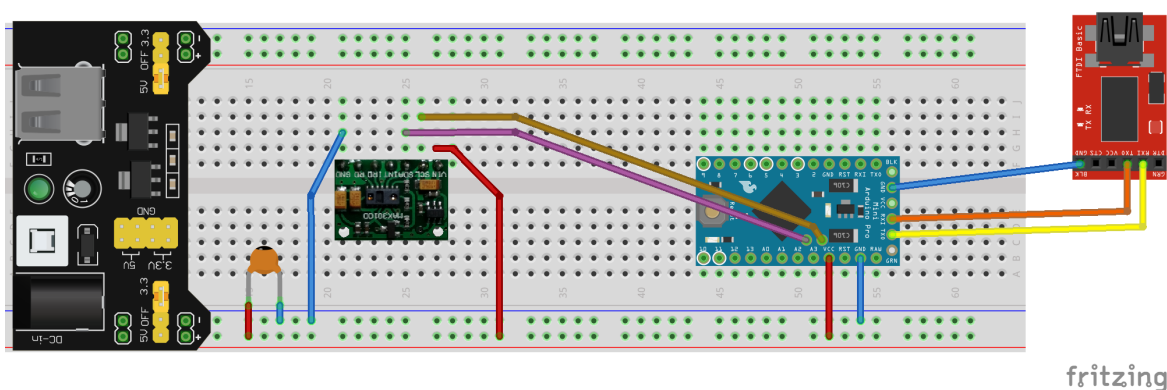


Figure 4.1: Circuit assembled for gathering PPG data from MAX30100, consisting of (from left to right) an YwRobot power source, a decoupling ceramic capacitor, an RCWL-0530 breakout board, an Arduino Pro and an FTDI breakout board. This illustration was produced using the software Fritzing.

4.3.2 Measurements references

In order to establish a referential target value for sensor readings, two market-available devices were used simultaneously as the experiment was carried out. The first was an medical oximeter, that measures heart-rate and oximetry using red and infrared light for transmissive PPG , used on user's finger. The second device is a fitness smartband, that uses two green light LEDs for reflective PPG, used on user's wrist. Immediately before and after each measurement, heart-rate values of both devices were logged. This gives a initial and final values reference for each device in the measurement time frame, allowing for heart-rate variation during the experiment to be taken into account.

4.3.3 Experimental procedure

The experiment consisted of resetting the Arduino board's software, starting the Python script in the computer, positioning the subject's finger firmly over the MAX30100 sensor and measuring his heart signals for a determined time of approximately 10 seconds. Thus, data acquired with this procedure provides several heart beat cycles, as well as a reasonable amount of samples to process using the algorithms specified in the project.

4.4 SIGNAL PREPROCESSING

In order to better analyze the proposed algorithms, additional signal processing is introduced to the sampled data, as exemplified in Fig. 4.2, in order to eliminate detrimental factors that could hinder the performance of each HR estimator. Such processes are described in this section, with intention to simulate signal processing present in real operation of PPG systems and maintaining the scope of this work.

4.4.1 Movement Artifact Removal

As explained in Section 2.3.2, reflective PPG sensors are prone to interference arising from motion artifacts due to user's movement. Artifact removal algorithms are a rich topic, but are not concerned in this work.

Nonetheless, motion artifacts are present in the test data originating from the positioning of the test subject's finger over the sensor at the beginning, and removing it after the measurement ended. This behavior can be observed in Fig. 4.2 during the edges of the measured PPG signal. They are identifiable by abrupt variations in the signal. Thus, in order to mitigate these artifacts, a simple filter was designed. Based on the successive differences of the signal (analogous to its derivative in discrete time), parts of the signal deviating from the mean above a certain threshold – corresponding to the motion artifacts – were removed.

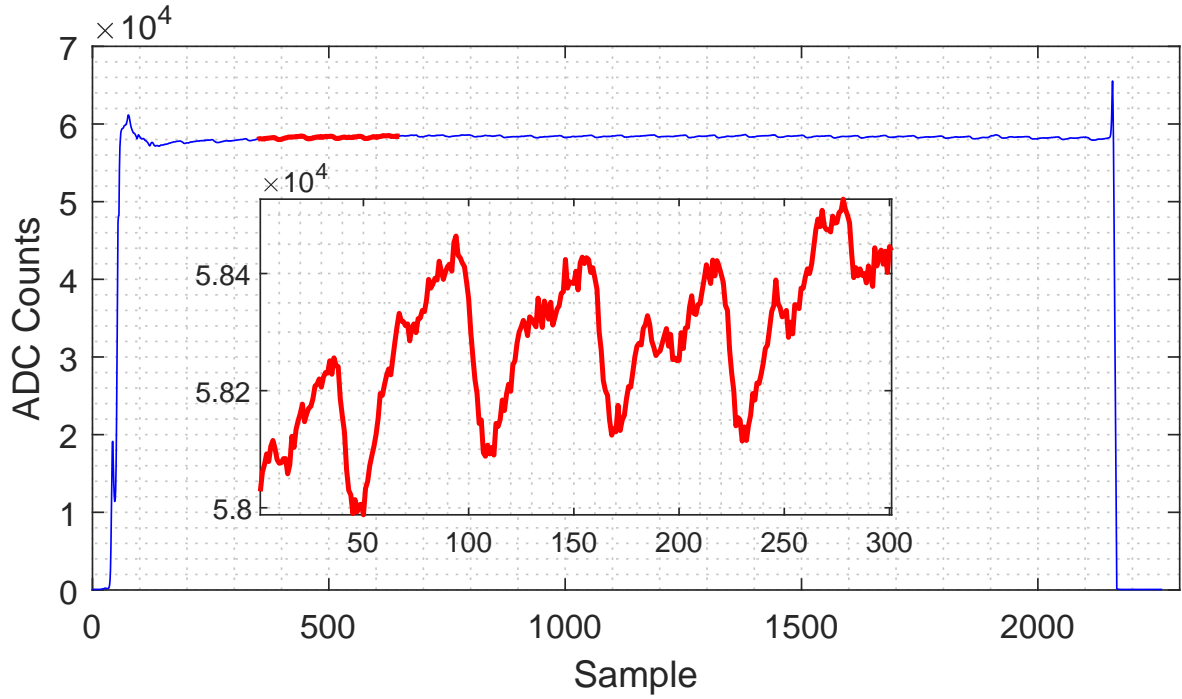


Figure 4.2: Example of an unprocessed PPG signal recorded using the experimental setup. In red, the highlighted part is plotted separately in order to show the PPG waveform. In the detected signal, the valleys represent systolic peaks, followed by diastolic peaks and signal feet, which are the highest values of each cycle, as described in Subsection 2.3.1. Motion artifacts are visible as abrupt variations at the beginning and end of the signal.

4.4.2 DC Filtering

The PPG signal measured by MAX30100 consists of a AC and DC components, as characterized in Section 2.3.1. For HR estimation, only the pulsating AC signal is necessary. Thus, the slowly varying DC components represent noise, and thus need to be blocked. As shown in Fig. 4.2, the highlighted part of the PPG signals contains a slowly varying behavior due to DC noise. A simple Infinite Impulse Response (IIR) recursive filter is implemented, as proposed by [40] and [41]. This DC blocker filter is specified by the following difference equation:

$$y(n) = x(n) - x(n-1) + Ry(n-1), \quad (4.1)$$

where $x(n)$ is the input signal, $y(n)$ is the filter output and R is a parameter chosen between 0.9 and 1, directly proportional to the frequency bands that is allowed to through the filter and inversely proportional to the convergence speed of the filter. The corresponding transfer function is given by:

$$H(z) = \frac{1 - z^{-1}}{1 - Rz^{-1}} \quad (4.2)$$

Thus, there is a zero at $z = 1$ and a pole at $z = R$, blocking mostly DC components. For higher frequencies, the pole and zero approximately cancel each other, not affecting higher frequencies perceptively.

4.4.3 Additional Filtering

Along with the previous processes, additional filtering is necessary to attenuate high frequency noise. Thus, supplementary filtering is used on the experimental PPG signals. The filters are designed as Butterworth filters in order to achieve a flat passband. Multiple filters are used and are properly defined with the correspondent results chapter.

The significance frequency interval of the PPG signal is estimated to be between 0.4 and 4 Hz, corresponding to 24 and 240 BPM. This interval guarantees that the signal band has significance most of the time, since the cutoff values of BPM represent extreme cases in which medical intervention is necessary.

5 MONITORING PLATFORM

Gathered health data must be securely stored, sorted and displayed in a friendly manner to the end-user, in this case, medical professionals. It is then necessary to develop a platform for data monitoring that can be easily accessible everywhere – personal computers, smartphones and tablets – to deliver health information collected by the wearable device. As part of this project, requirements for such platform are proposed and a minimum viable prototype is built directed at health-related data monitoring and acceptable end-user experience.

5.1 REQUIREMENTS

Based on validated medical needs [42] and projected features of a new wearable hardware, the development of a prototype visualization platform to display gathered sensor data started by outlining requirements for said platform. The first decision taken was to develop it as a web application, since web browsers are prevalent in all computers and mobile devices (such as smartphones). Thus, the project need to run well on computers and mobile devices. This is possible by using responsive design.

Requirements for development include selecting a programming environment that allows for versatile and maintainable code. This provides a development setting with more capabilities of adapting to user feedback and expanding functionalities.

Regarding the aspect of user experience, it is important to emphasize that the target public are physicians and healthcare professionals looking for a remote supervision platform for post-surgical patients in hospitals. Therefore, the features developed must allow said users to perform the necessary activities – monitor patients' health conditions using wearable sensors – and also provide element for improving the effectiveness of said activity. Some tools that may be useful for the final users are:

- text reminders for patients;
- use of visual graphics for data visualization;
- capabilities of sharing patients with teams.

The above features are going to be targeted in the scope of this project. They will be included in the prototype developed in order do demonstrate and further validate these tools for the healthcare platform concerned.

Finally, it is necessary to note that this project only covers the front end of the application prototype. That is, only the visual aspects and functionalities available to the end user are ex-

plored. The part concerning data storage, processing and network interfaces – known as backend – are treated in separate works, such as [42]. Scrutinizing only the frontend part of the prototype poses no loss of functionality, since the built website can be easily adapted for different backend architectures.

5.2 DEVELOPMENT OF THE APPLICATION

Web applications are currently based on well-established technologies. By studying such technologies and the capabilities of HTML5, CSS3 and the programming language JavaScript, the project settled with the Vue.js framework [43]. This toolchain allows the project requirements to be technically achievable, simultaneously providing a modular programming paradigm. Since Vue.js is centered around web components, it's more convenient to organize the project and reuse code, avoiding unnecessary work.

Additional open-source libraries are selected to provide extra features and make development more agile. Some noteworthy add-ons are:

- VUEX [44]: used for state management, that is, storing, updating and passing data between components;
- Vue Router [45]: allows for navigation between webpages by defining URL routes;
- Bulma CSS [46]: a CSS framework used for styling components;
- Chart.js [47]: library used for creating data charts showing health information;
- Axios.js [48]: web client used to make HTTP requests, allowing the website interface with backend architectures.

Using of the above tools, development of the visualization platform website was carried out in parallel with hardware development. The following section presents the achieved results and scrutinizes features objectives and usability.

5.3 FEATURES

The website built is centered around main features and additional tools. The main parts consist of an user login and registration systems, patients list, a news feed and user profile with the main contact and professional information of the medical user. Additional tools, discussed previously, are text reminders in patients profiles, data visualization through charts and a team sharing system. Each individual component is commented next.

5.3.1 Login and Registration

For users to retain their profiles and other data, as well as accessing it in different platforms, it is mandatory for the application to support user accounts. Hence, a login and registration systems were built, allowing users to create unique accounts with their e-mail and relevant information and access it with a secured password in any web browser. Both systems share the same screen and are separated by tabs, each one of them shown in Fig. 5.1 and Fig. 5.2. This allows new users to quickly create a new account and start using the platform.



The image shows a web application interface for login. At the top, there is a green smartphone icon with a heart and ECG symbol on its screen. To the right of the icon, the text "AEON | DATA" is displayed in a blue and black font. Below the icon, there are two tabs: "Login" (selected) and "Registre-se". The "Login" tab is active, showing two input fields: "email" and "senha". Below the "senha" field, there is a blue link "Esqueci minha senha" and a blue "Login" button.

Figure 5.1: Login component of the web application, where registered users can access their accounts.

In the login component, presented in Fig. 5.1, users can access their accounts by using the registered e-mail and corresponding password, in the designated fields, and clicking the “Login” button. Users can also recover their lost password in case of forgetfulness.

New users have to create a unique account by inputting their relevant personal data, with information fields shown in Fig. 5.2. Selected user information are:

- Name;
- Medical specialty;

- CRM (Regional Council of Medicine registration number);
- Telephone number;
- E-mail address;
- Password (with confirmation);
- CPF (personal registration number);
- User profile picture (optional).

Users are prompted the above information so as to check their personal identity – since the CPF number is unique to each Brazilian citizen –, medical credentials and contact information. Only the medical specialty, telephone number and e-mail may be shared with other users in the platform. Registration is only processed if every information field is provided – optionally for the user profile picture – and both password and confirmation are the same.

5.3.2 Patients list

The main objective of the platform is to provide a remote supervision system for multiple patients simultaneously. Therefore, users utilize this component to add new supervised patient and list all the current monitored patients.

To register a new patient in the component presented in Fig. 5.3, users must provide the following information:

- Patient name;
- Responsible physician;
- Birth date;
- Reason for hospitalization;
- Current hospital;
- Patient profile picture (optional).

Registered patients are listed in an additional screen in from the most recently added, as shown in Fig. 5.4.

From the screen in Fig. 5.4, users can check basic data from registered patients, such as birth date, reason for hospitalization and responsible physician. Through this list, user can access individual patients' profiles to check more detailed information.

From the patient's profile, users can check their full information as displayed in Fig. 5.5. Also, it is possible to verify user health situation through charts, add text reminders and share

them with teams with the menu present in the left part of Fig. 5.5. These functionalities are extended in ensuing sections.

5.3.3 News feed

The news feed provides an overview of user's activities. Actions including newly registered patients and added text reminders are displayed in chronological order, from the most recent. By clicking an item, such as a new text reminder or a new patient as exhibited in Fig. 5.6, users can check new information.

This feature allows for users to quickly control new actions, for instance reading a new text reminder left by another user for a common supervised patient or verify if a new inmate has stabilized his heartbeat in the last few hours.

5.3.4 User profile

Users' profiles display general information of their accounts, as discussed in the registration section, and enables them to edit or correct their data, as well as update their profile picture, all presented as in 5.7. Users can also change their password, for increased security, by inputting their current password and providing a new one and confirming it, in the appropriate fields illustrated in Fig. 5.8. Account's security is then updated if both the current password is correct and both new keys match.

5.3.5 Text reminders

Keeping attention on many patients can be a arduous task for a doctor. Busy schedules and working on different workplaces can lead to loss of information or forgetfulness. It is then important to allow users to record information regarding patients. Text-based reminders proved to be effective and versatile ways of conveying reports. Considering these scenario, the web application contains capabilities of writing and saving text notes on individual patients' profiles, allowing every doctor with whom the patient was shared. That enables users to conveniently share memos with one another, making the task of multiple supervision more favorable.

The reminders list, shown in Fig. 5.9, contains notes added to a patient for the most recent. Basic information contained in each memo is shown in this list, including its text content, added date and author. By clicking an item, the user can read and edit it, in the same interface displayed in Fig. 5.10.

Creation of text memos are available as shown in Fig. 5.10. Users input the desired information in the text area. Additionally, inputs for basic metabolic panel (BMP) and complete blood count (CBC) of the patient are available. Since both of these medical examinations are commonly used to give the general state of patient's health, adding it to text reminders are of interest in the

application. Details of both exams can be obtained on [49] and [50], respectively.

5.3.6 Data visualization

Data visualization gives detailed health information of each patient using a wearable device. Hence, it is important to present data clearly and allow for searching and classifying depending on doctor's necessities. The library Chart.js was utilized along with bindings for Vue.js to make integration between frameworks uncomplicated. User can select to view data including calories spent, steps taken, floors climbed and heart-rate from the menu presented on Fig. 5.11. This set of data options is present as available in the Fitbit environment, which was used to validate earlier versions of the web application prototype. Data can be filtered by date, displaying values from daily, weekly or monthly periods with the menu on the right of Fig. 5.11.

5.3.7 Team sharing

Sharing patients is made possible in the application so as to enable doctors to cooperate in supervision. The sharing feature is centered around medical teams. By creating private groups, users can add others through e-mail, as desired, with the interface present in Fig. 5.14. They can then add common patients to a team from the patient's profile, as shown in Fig. 5.12, allowing other members to visualize inmates' profiles and health data, as well as adding text memos to share information regarding the patient's situation. Shared patients are listed in similar manner to the regular list (Fig. 5.12), but is accessible through the teams interface presented as in Fig. 5.13. Component shown in Fig. 5.14 also enable users to edit a team's name or erase it from the application.

5.4 FUTURE WORK

As the time of this writing, the web application is under active development. It has been submitted to the local health ethics Council in order to approve it for further testing and validation in real hospitals. Based on future user's feedback, the project is committed to improve user experience and add new features. The main future objective is to integrate the web platform to the developed wearable hardware that this project handles. The current state of the web applications is ready for quick extension to accommodate new necessities of the wearable device being build. By integrating relevant data, for instance real-time heart-rate and pulse oximetry, doctor can have richer reports on patients' conditions.



[Login](#) [Registre-se](#)

Nome

Especialidade

CRM

Telefone

Email

Senha

Confirmar

CPF

Foto
 Procurar...

Figure 5.2: New users must register to use the application, entering their information using this component.

Feed Pacientes Equipes Perfil Sobre Logout

Pacientes
Lista de Pacientes
Adicionar Paciente

Nome

Médico Responsável

Data de Nascimento

Causa da Internação

Hospital

Foto Procurar...

Cadastrar Paciente

Figure 5.3: To add new supervised patients, users must register their corresponding information as indicated in this component.

Feed Pacientes Equipes Perfil Sobre Logout

Pacientes
Lista de Pacientes
Adicionar Paciente

Marcos Paulo
Data de nascimento: 11/11/1986
Causa da internação: Conjuntivite
Médico responsável: Gabriel Pinheiro

Miguel Carlos
Data de nascimento: 28/03/1990
Causa da internação: Tosse aguda
Médico responsável: Gabriel Pinheiro

Figure 5.4: Patients supervised personally by users are displayed in a list from the most recently registered, allowing for users to access each patient's profile.

Opções

Informações

Dados de Saúde

Pulseira

Lembretes

Equipes

Miguel Carlos



Data de nascimento: 28/03/1990

Causa da Internação: Tosse aguda

Médico responsável: Gabriel Pinheiro

Hospital: HUB

Figure 5.5: Patient profiles contain relevant information and a menu to access components to visualize health data, add a text reminder or share with a team.



Figure 5.6: The latest user activity is show in the news feed component from the most recent, informing of recently registered patients or a new text reminder.



Figure 5.7: Users' profiles allow editing account information and changing password.

Configurações

Perfil

Segurança

Atualizar senha

Senha Antiga

Nova Senha

Confirmar Nova senha

Atualizar

Figure 5.8: Password can be updated by inputting the current password, choosing a new one and confirming it.

Opções

Juliano Pretz

Informações

Dados de Saúde

Pulseira

Lembretes

Equipes

	Alterar dose de remédio	
01/06/2018		Gabriel
	Paciente apresenta melhora.	
06/05/2018		Gabriel



Figure 5.9: Reminders are listed from the most recent, showing the included text information, date added and author. Clicking a reminder allows users to check exam values and to edit it.

Novo lembrete

02/06/2018 Gabriel

Alterar dose de remédio

Dados químicos

Na

K

Cl

Co2

Bun

Creat

Gluc

wbc

HgB

Hct

Plt

Figure 5.10: Reminders are text messages that can also include exams results form BMP and CBC, allowing users to record information relevant to each patient.

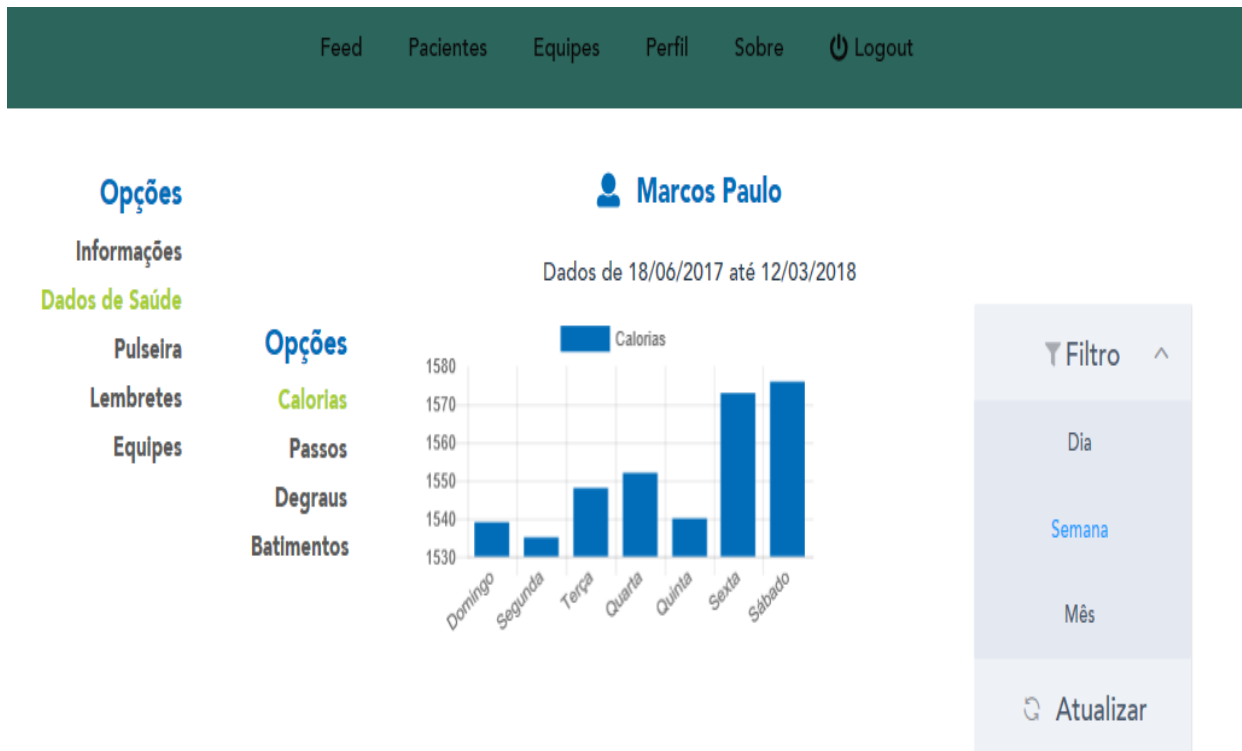


Figure 5.11: Health data measured with wearable devices can be visualized through charts in this component. The prototype includes information for calories, steps, floors and hear rate.



Figure 5.12: Patients can be shared with the users' teams by adding them with this component.



Figure 5.13: Users can access one of their teams profile by choosing it in this list, and also create a new team.

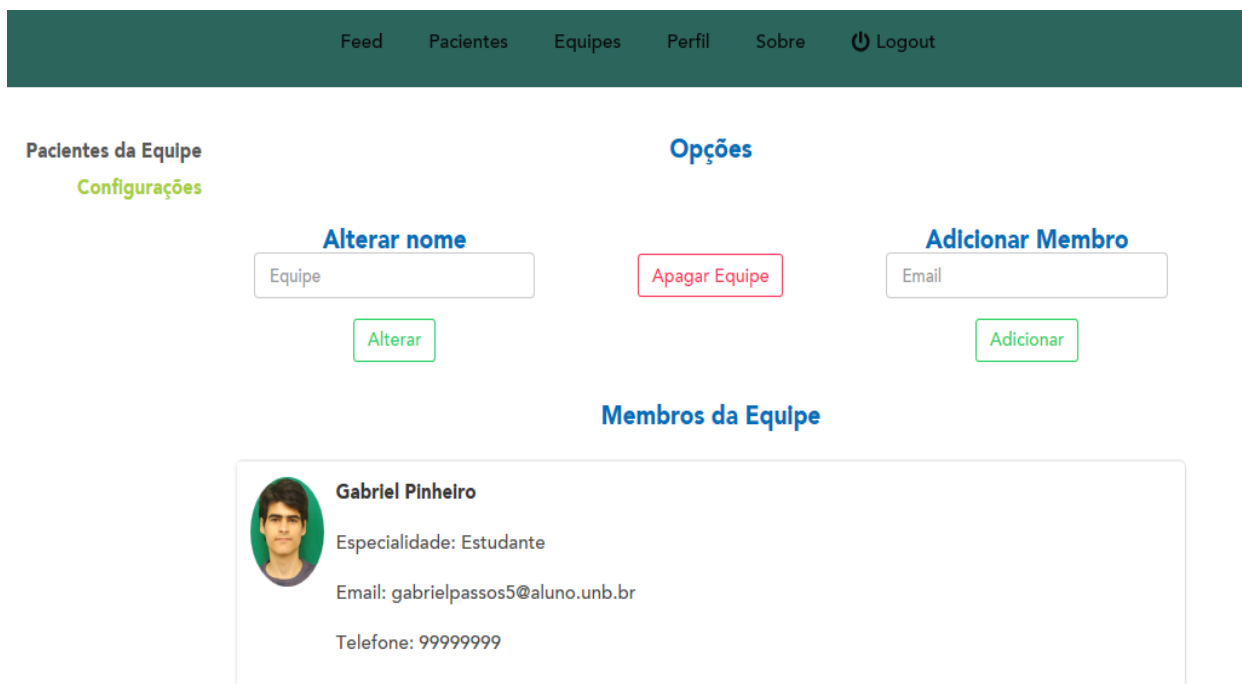


Figure 5.14: Team members can add new members through e-mail, change the team's name or delete it. Other members' contact information are also listed.

6 RESULTS AND DISCUSSION

The experimental procedure established in the previous chapter is carried out in order to find optimal operational parameters, aiming to achieve lowest power consumption and processing time without compromising precision. For this, the parameters of IR LED current, IR LED pulse width and ADC sample rate were configured to an assortment of values in the MAX30100 IC, according to its datasheet [35]. From the experimental data, each considered sample was preprocessed, as specified in Section 4.4, and the correspondent HR was estimated using the algorithms stated in Chapter 3.

Results BPM values are presented along each of the references from the oximeter and the smartband equipments at the start and at the end of each measurement. The samples only consider IR LED data, but the MAX30100 IC was operated at SPO₂ mode, that is, the red LED was functioning as well. This configuration is done to simulate HR detection in a real PPG system, which could also allow for pulse oximetry, when the red LED would be another aspect to consider. Any experimental or processing alterations are stated in the matching section. Discussion of the results are commented along each section, and a concluding remark is given at the end of the chapter.

Filtering of the data gathered experimentally is described in the methodology, more specifically at Section 4.4. The DC blocker filter is realized with $R = 0.95$ for Equation (4.1), which is sufficient for filtering most of the low frequency noise in the signal. The low-pass filter used is a 6-th order Butterworth filter with cutoff frequency of 4 Hz, attenuating high frequency noise satisfactorily. The next table presents a summary of the time and space complexities of each algorithm, regarding Chapter 3, for comparison purposes:

Algorithm	Time complexity T(n)	Space complexity S(n)
ESPRIT	$\mathcal{O}(N^3)$	$\mathcal{O}(NM)$
FFT	$\mathcal{O}(n \log n)$	$\mathcal{O}(n \log n)$
Autocorrelation	$\mathcal{O}(n)$	$\mathcal{O}(n)$
Zero crossing	$\mathcal{O}(n)$	$\mathcal{O}(1)$
Peak detection	$\mathcal{O}(n)$	$\mathcal{O}(n)$

Table 6.1: Time and space complexities considered for each algorithm tested.

6.1 LED CURRENT LEVEL

The LED Driver included in the MAX30100 IC can be digitally configured to typical current values between 0 mA and 50 mA. Although the datasheet states that these values can vary widely due to proprietary methodology used in the IC, they are significant as a mean current level of the

LED. Thus, it is possible to gather conclusions regarding the power consumption in each current level and its implications on overall precision of the studied algorithms. The other parameters are kept constant, at pulse width of $1600 \mu\text{s}$ and sampling rate of 100 sps.

During the experimental procedure, current level configurations of 50 mA, 46.8 mA, 43.6 mA, 40.2 mA, 37 mA, 33.8 mA, and 30.6 mA saturated the IC's ADC, meaning that no significant measurements could be achieved for these values of current in the experiment proposed. The remaining measurements were processed with a signal time window of 5 seconds, corresponding to 500 samples. In Fig. 6.1, results are presented in relation to the oximeter reference, while Fig. 6.2 refers to the smartband values from the start and the end of the experimental measurements.

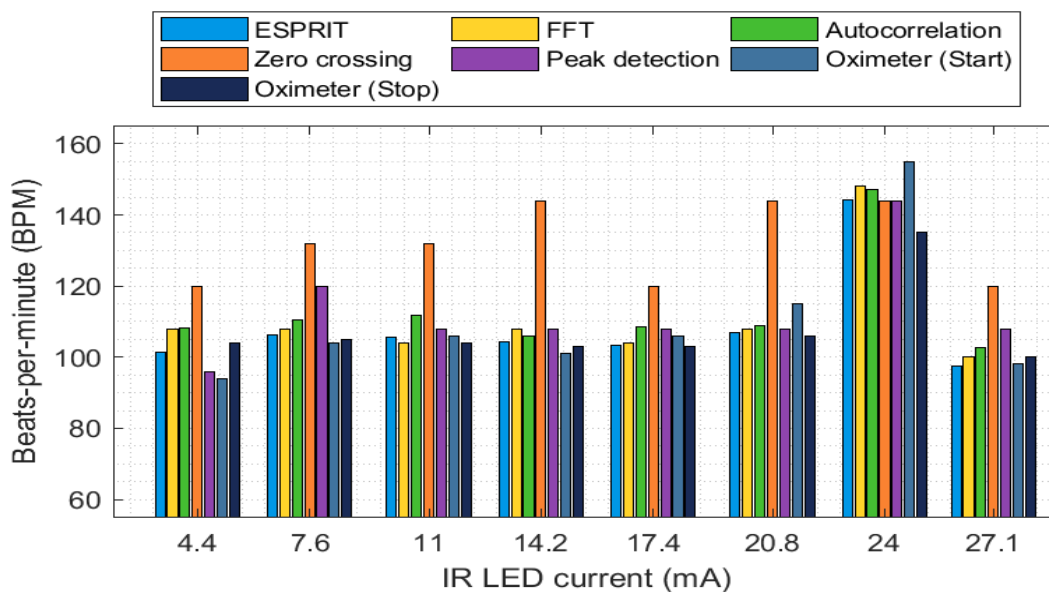


Figure 6.1: Estimated BPM values for each algorithm for different current level configurations. The reference BPM values are given by the oximeter during the start and at the end of the measurement.

6.1.1 Discussion

Some comments regarding the performance of algorithms in the various current levels considered are necessary. Initially, it is possible to conclude that the algorithms showed consistent results across all IR LED current levels, maintaining similar error margins in reference to the oximeter or the smartband. The expected results were that decreasing current values would hinder signal performance. Since there is lower signal power with smaller current levels, lower SNR occurs, with the sources of noise remaining constant. Although there is no evidence in the presented data that varying current affects precision of BPM estimation, it is sufficient information to assure that significant PPG measurements are possible on all current levels. Thus, operating at

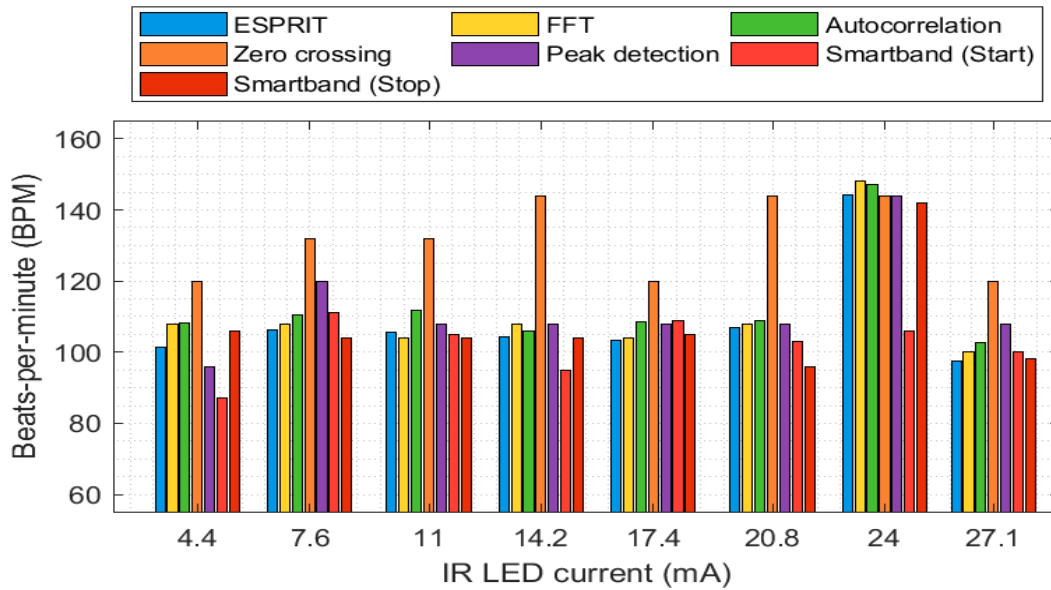


Figure 6.2: Estimated BPM values for each algorithm for different current level configurations. The reference BPM values are given by the smartband during the start and at the end of the measurement.

lower current values still result in satisfactory signal quality.

Regarding each algorithm, ESPRIT, FFT and autocorrelation presented sharper results overall. However, zero crossing and peak detection have shown mixed results.

The algorithm based on ESPRIT resulted in reasonable estimation to BPM. In relation to Fig. 6.1 (the oximeter reference), the ESPRIT estimation were close to either the starting or stopping values (e.g. for current levels 14.2 mA and 17.4 mA), or around the mean of the two oximeter reference values (e.g. for 24 mA and 4.4 mA). Comparing to the smartband reference, ESPRIT estimations differed widely from starting values of the smartband (in Fig. 6.2), but became closer at the end reference values of the measurements shown in Fig. 6.2. This observation may show that the smartband considers a longer measurement time window, taking more time to converge to actual values during experimental measures.

The performance of the FFT and autocorrelation were similar to the ESPRIT algorithms, resulting in close BPM estimation for most of the cases. Both of these algorithms may show a tendency to overestimate the BPM values, as they were usually above both the starting and stopping values for both references and the ESPRIT estimations. However, this overestimation is not too significant, differing less than 10 BPM to the values compared.

As for zero crossing and peak detection, both showed a clearer overestimation bias. The zero crossing algorithm especially presented disparities from 10 BPM up to 30 BPM from the references and other estimators. This poor performance may be noise-related – computing additional

incorrect zero crossings – or due to baseline fluctuations in the PPG signal – dislocating its curve downward or up, and thus registering the other regions of the PPG wave as zero crossings. Peak detection, on the other hand, displayed less flagrant disparities, differing from around 5 up to 10 BPM higher than the estimators and references.

A noteworthy outlier is the 24 mA current value, in which results mostly agreed with each other and stayed at the mean value of the oximeter reference. Yet, the smartband reference differed from other results by nearly 40 BPM, but at the end of the measurement became consistent with the rest, as evident in Fig. 6.2. This is another evidence of the longer time window that the smartband may consider for its estimation. Also, zero crossing and peak detection showed good performance in this outlier, but since they are suspected to have an overestimation bias, these might be false positives.

6.2 LED PULSE WIDTH

For power saving reasons, LEDs are operated in pulsed mode with a low duty cycle in the MAX30100. Thus, the LEDs are not always on, but proper measurements are still made. Power saving is achieved since the power consumption is approximately proportional to the percentage of the duty cycle of the LED [26]. This technique is commonly known as Pulse Width Modulation (PWM). The components datasheet provide average current values of the LED for different pulse width configurations [35], which are considered for discussion regarding power saving and algorithm precision. The IC also contains a “High Resolution Enable” bit flag, which, when set high, fixes the ADC resolution to 16 bits and pulse width to 1.6 ms. For any other pulse width configuration, this flag must be set low.

In the MAX30100, pulse width configuration implies an alteration in the component’s ADC resolution due to internal functioning of the IC. Thus, shorter pulse widths are compulsorily linked to smaller ADC resolutions. Possible values of LED pulse width and the corresponding ADC resolution in SPO₂ mode is present in the following table:

Pulse width (μ s)	ADC resolution (bits)
1600	16
800	15
400	14
200	13

Table 6.2: LED pulse width configurations tested and the implied ADC resolution.

Varying the LED pulse width, and consequently the ADC resolution, experimental data was gathered fixing the LED current level at 27.1 mA and sample rate at 100 sps. Data was processed with an amount of 500 samples, corresponding to a 5 seconds time window. The BPM estimations are show in accordance to the initial and final readings of each reference considered, in Fig. 6.3

to the oximeter, and in Fig. 6.4 to the smartband.

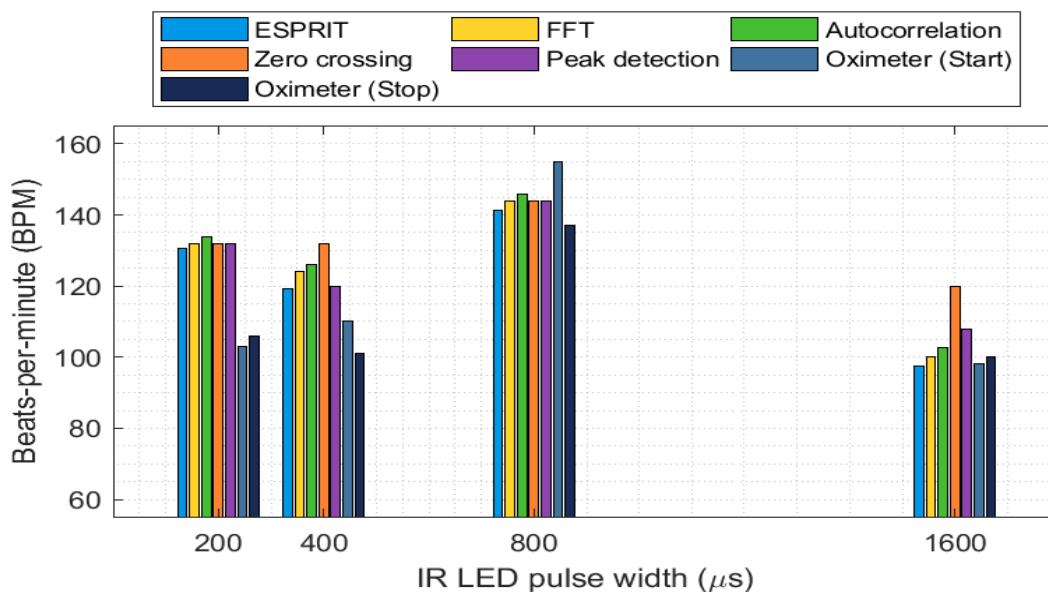


Figure 6.3: Estimated BPM values for each algorithm for different pulse width and corresponding ADC resolution configurations. The reference BPM values are given by the oximeter during the start and at the end of the measurement.

6.2.1 Discussion

Results of the estimator are consistent with the previous test case in Section 6.1 for the situations with 1.6 ms and 800 μs of IR LED pulse width. In these cases, estimations using ESPRIT, FFT and autocorrelation algorithms are observed to be around the average values measured by the oximeter reference in Fig. 6.3. Zero crossing and peak detection estimator show an previously noted overestimation bias for the 1.6 ms case and are consistent with other values for 800 μs.

The references – oximeter and smartband – agree between each other in Figs. 6.3 and 6.4, showing good convergence after the measurements for reasons noted before. From this observation, strong evidence of unsatisfactory performance is present in cases with 400 μs and 200 μs of IR LED pulse width for all estimators. For 400 μs the errors observed for the estimators range from 10 BPM (ESPRIT and peak detection) up to 20 BPM (zero crossing), with an error around 15 BPM for both FFT and autocorrelation algorithms. As for the 200 μs configuration, errors increased to around 25 BPM for all estimators in comparison to either reference.

These tests showed significant results regarding algorithm performance with reduced pulse width and ADC resolution. Pulse width shorter than 50% of the maximum value were believed to considerably degrade estimation precision. The shortest pulse width generated higher error

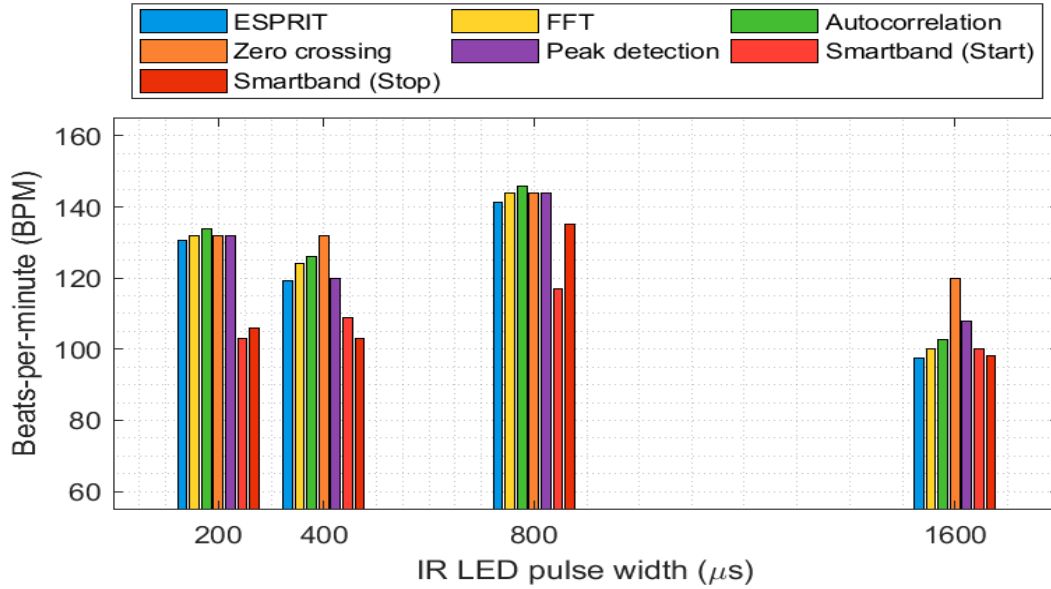


Figure 6.4: Estimated BPM values for each algorithm for different pulse width and corresponding ADC resolution configurations. The reference BPM values are given by the smartband during the start and at the end of the measurement.

values than other configurations. This might be due to signal ripple caused in PWM operation, which is more substantial the shorter the pulse is. Another factor to take into consideration is the lower ADC resolution corresponding to lower LED pulse width. Therefore, along with suspected ripple noise, the loss in sampling resolution could further impair the output PPG signal.

6.3 SAMPLE RATE

Sample rate corresponds to the amount of samples taken by the MAX30100 ADC each second, given in samples per second (sps). Thus, this is equivalent to the sampling frequency (f_s) of the system. The sample rate can be set from 50 sps up to 1 ksp. It is important to note that, for a given ADC resolution, not all sample rates are available to use. For example, with an ADC resolution of 13 bits, all sample rates can be used. However, for a resolution of 16 bits, only values of 50 or 100 sps are obtainable [35].

The experimental procedure was carried out with current level fixed at 27.1 mA and ADC resolution – and the implied pulse width, as described in the previous section – to the maximum value available. Sample rates of 600, 800 and 1000 sps were not supported by the recording equipment used, thus no tests were made with these values. The values used in this experiment are present in the following table:

Sample rate (sps)	ADC Resolution (bits)
50	16
100	16
167	15
200	15
400	14

Table 6.3: Maximum available ADC resolution for each sample rate configuration tested.

The resulting BPM values are shown with the corresponding sample rate. In Fig. 6.5, estimations are compared to the oximeter reference. In Fig. 6.2 the reference for estimation values are the smartband readings.

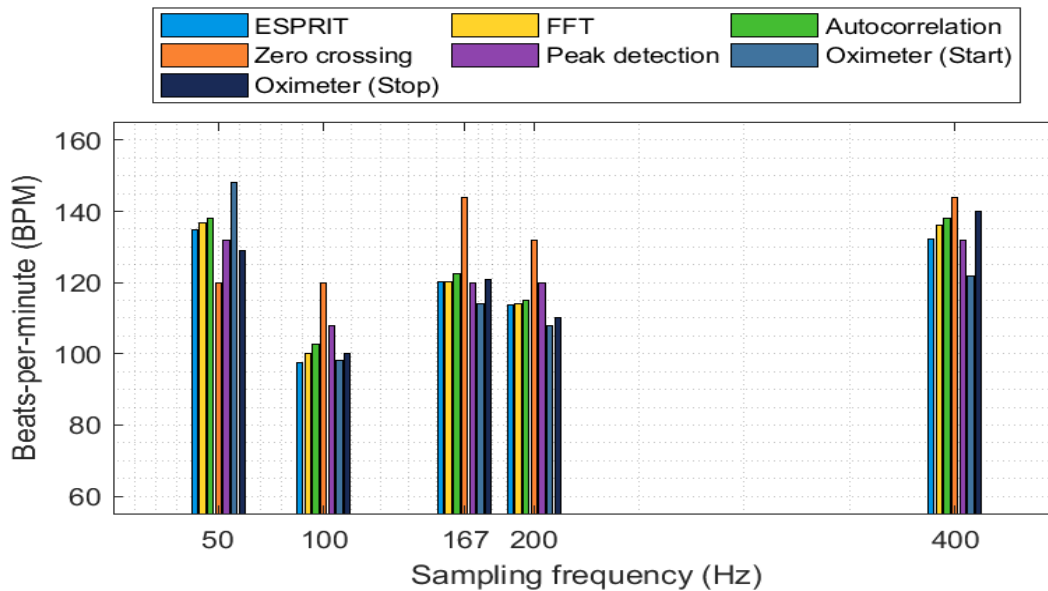


Figure 6.5: Estimated BPM values for each algorithm for different sample rate configurations, with the maximum ADC resolution available. The reference BPM values are given by the oximeter during the start and at the end of the measurement.

6.3.1 Discussion

This experimental stage presented results aligned to the preceding sections. The algorithms based on ESPRIT, FFT and autocorrelation displayed acceptable performance in relation to the references and they mutually agree. The peak detection algorithm has an overall increase in its performance, being in line with the references and the previous cited algorithms. Zero crossing-based algorithm once more shows a general overestimation bias, except for one outlier case (50 Hz), both in Fig. 6.5 and in Fig. 6.6, in which it underestimated the BPM value and thus sig-

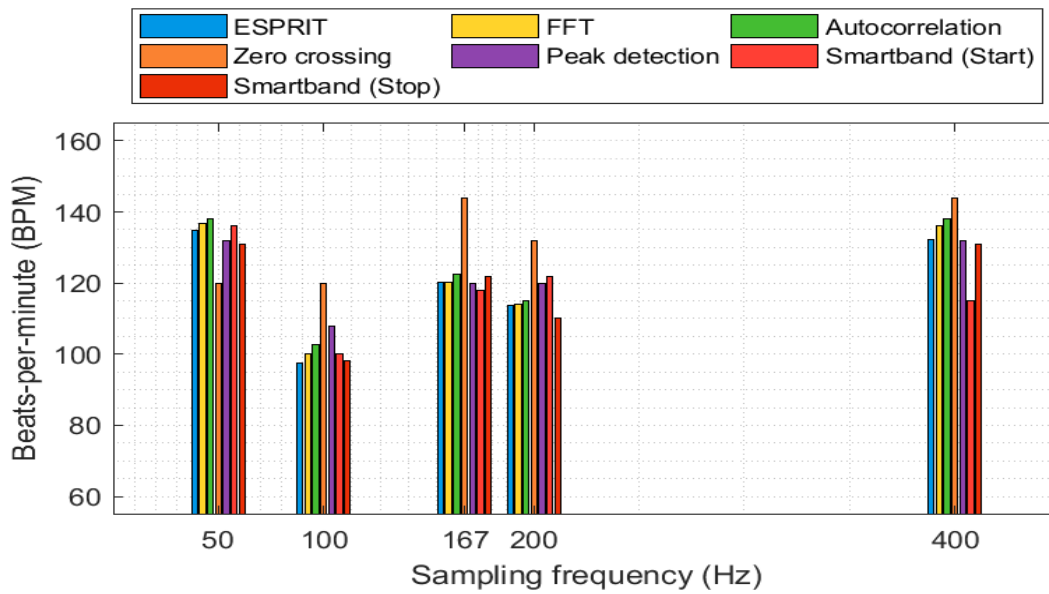


Figure 6.6: Estimated BPM values for each algorithm for different sample rate configurations, with the maximum ADC resolution available. The reference BPM values are given by the smartband during the start and at the end of the measurement.

nificantly differed from references and other measurements. This strengthens the unfitness of the implemented zero crossing algorithms for this application, with a global poor performance.

In terms of the effect of the sample rate configuration in the signal estimation, no strong indicators point to any performance alteration. Since significant measurements were achieved with all sample rates tested, as evident in adhering references of the oximeter values in Fig. 6.5 to the smartband values in Fig. 6.6, it is possible to draw conclusions based on previous experimental observations. For instance, higher sampling rates are only available for shorter pulse widths, which were suggested to decrease estimator precision and signal resolution. Therefore, operating at lower sampling rates (either 50 or 100 Hz) might be advantageous in terms of general performance.

7 CONCLUSION

Research presented in this work allowed for a wider understanding of characteristics of photoplethysmography systems, applied to the context of healthcare wearable devices. In order to implement a heart rate monitoring IoT device for post-surgical patients, a PPG system is outlined. Details for achieving a low-cost device using a near-infrared LED – deemed better alternative for being able to deliver reasonable measurements as well as being invisible, and thus unobtrusive, for users' vision – and a photodiode are posed. Consideration regarding addition aspects of the implementation, including photodiode interfacing using a TIA and filtering and amplification staged are also made.

Digital signal processing of PPG sensor signals are likewise studied, contemplating algorithms with distinct complexities are analyzed, namely ESPRIT, FFT, autocorrelation, zero crossing and peak detection. A discussion of each specific algorithm characteristics, possible advantages and disadvantages is carried out. From this, a code-implementable algorithm is proposed based on each technique considered.

To validate each signal processing method projected for heart rate estimation systems, experimental measurements are executed using a MAX30100 reflective PPG system to detect heartbeats through the test subject's finger. Afterward, the sample signals are filtered according to application's necessities and processed with each algorithm using MATLAB and compared to measurement references of an oximeter equipment and a commercial smartband.

Results show evidences that algorithms based on ESPRIT, FFT and autocorrelation show acceptable performance of HR estimation in relation to the references used. The implemented peak detection and zero crossing methods were deemed unfit for the application for showing inconsistent results. Although the peak detection had a suspected overestimation bias, its performance was tolerable for some test cases and could be optimized in the future. However, the zero crossing algorithm tested presented clear overestimation bias and was not considered reliable enough for heart rate estimation in this case.

Using different operational conditions, optimal conditions for power and processing saving are outlined. Algorithms based ESPRIT, FFT and autocorrelation were capable of computing reasonable BPM values for low IR LED current level as well as low sample rate. Moreover, the reduction of the LED's pulse width and ADC resolution resulted in sensible degradation of estimators, thus considered an inadequate mean of achieving power reduction.

In terms of processing power, although ESPRIT-based algorithms may be regarded as more precise, its higher processing complexity was considered and unreasonable trade off in comparison to other algorithms in the wearable context. Other high-precision algorithms such as FFT and autocorrelation present similar precision results with simpler implementations for targeted wearable devices. The most simple algorithms – namely zero crossing and peak detection – resulted in

rather unreliable HR estimation and thus need additional optimization to be used for PPG signal processing. Therefore, the simplicity of said algorithms may be counterbalanced for its lack of robustness and supplementary need for development.

Another noteworthy observation regards the measurement time window used for the commercial smartband used as reference. Experiments carried out showed that the smartbands refresh rate is slower than the oximeter and results from the algorithms, latter converging to more compatible measurements. It is then apparent that the smartband uses a longer window of time for its resulting BPM values, which can be estimated to around 10 seconds, which was the time each experimental measurement.

All in all, the results presented offer a solid basis for future work on building heart rate monitoring wearable devices. Additional studies may consider implementing and optimizing the algorithms proposed in deployable microcontrollers in order to garner real-time estimation results or integrating a blood oxygen saturation (SPO₂) capability. Other topic of interest is to adapt the proposed experiment for different measurement sites on the body, such as users' wrists. Related topics are the development of more robust filtering systems, especially concerning motion artifacts mitigation in PPG sensors. The monitoring platform prototype can be further validated and integrated the PPG sensor used during this work.

BIBLIOGRAPHY

- 1 Future Market Insights. *Global Wearable Fitness Trackers Market to Reach an Exponential Market Value of US\$ 93.69 Bn by 2027 End*. 2017. Last visited 10/6/2018. Disponível em: <<https://www.futuremarketinsights.com/press-release/wearable-fitness-trackers-market>>.
- 2 NELSON, E. C.; VERHAGEN, T.; NOORDZIJ, M. L. Health empowerment through activity trackers: An empirical smart wristband study. *Computers in Human Behavior*, v. 62, p. 364 – 374, 2016. ISSN 0747-5632. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S0747563216302369>>.
- 3 FEIJO, V. B. E. R.; JUNIOR, L. C.; SOUZA, R. K. T. d.; DIAS, A. O. Análise da demanda atendida em unidade de urgência com classificação de risco. *Saúde em Debate*, scielo, v. 39, p. 627 – 636, 09 2015. ISSN 0103-1104. Disponível em: <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S0103-11042015000300627&nrm=iso>.
- 4 REIS, P. M. G. I.; DA COSTA, J. P. C. L.; MIRANDA, R. K.; DEL GALDO, G. Esprit-hilbert-based audio tampering detection with svm classifier for forensic analysis via electrical network frequency. *IEEE Transactions on Information Forensics and Security*, v. 12, n. 4, p. 853–864, April 2017. ISSN 1556-6013.
- 5 REIS, P. M. G. I.; DA COSTA, J. P. C. L.; MIRANDA, R. K.; DEL GALDO, G. Audio authentication using the kurtosis of esprit based enf estimates. In: *2016 10th International Conference on Signal Processing and Communication Systems (ICSPCS)*. [S.l.: s.n.], 2016. p. 1–6.
- 6 CORDEIRO, T. F. K.; DA COSTA, J. P. L. C.; LIU, K.; BORGES, G. A. Kalman-based attitude estimation for an uav via an antenna array. In: *2014 8th International Conference on Signal Processing and Communication Systems (ICSPCS)*. [S.l.: s.n.], 2014. p. 1–10.
- 7 CORDEIRO, T. F. K.; DA COSTA, J. P. C.; SOUSA, R. T. de; SO, H. C.; BORGES, G. A. Improved kalman-based attitude estimation framework for uavs via an antenna array. *Digital Signal Processing*, v. 59, p. 49 – 65, 2016. ISSN 1051-2004. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1051200416300914>>.
- 8 PRETTZ, J. B.; DA COSTA, J. P. C. L.; ALVIM, J. R.; MIRANDA, R. K.; ZANATTA, M. R. Efficient and low cost mimo communication architecture for smartbands applied to postoperative patient care. In: *2017 Second Russia and Pacific Conference on Computer Technology and Applications (RPC)*. [S.l.: s.n.], 2017. p. 1–5.
- 9 DYER, A. R.; PERSKY, V.; STAMLER, J.; PAUL, O.; SHEKELLE, R. B.; BERKSON, D. M.; LEPPER, M.; SCHOENBERGER, J. A.; LINDBERG, H. A. Heart rate as a prognostic factor for coronary heart disease and mortality: findings in three chicago epidemiologic studies. *Am J Epidemiol.*, v. 112, n. 6, p. 736–49, 1980.
- 10 SIGMUND, A. E.; FANG, Y.; CHIN, M.; REYNOLDS, H. R.; HORWITZ, L. I.; DWECK, E.; ITURRATE, E. Postoperative tachycardia. *Mayo Clinic Proceedings*, Elsevier, v. 92, n. 1, p. 98–105, Jan 2017. ISSN 0025-6196. Disponível em: <<http://dx.doi.org/10.1016/j.mayocp.2016.08.005>>.
- 11 World Health Organization, WHO. *Density of physicians (total number per 1000 population, latest available year)*. 2017. <http://www.who.int/gho/health_workforce/physicians_density/en/>. Online; accessed 10/6/2018.
- 12 VORVICK, L. J. *Noninvasive*. [S.l.]: MedlinePlus, 2017. <<https://medlineplus.gov/ency/article/002269.htm>>. Last visited 10/6/2018.

- 13 PASTORE, C. A. et al. Diretriz de interpretação de eletrocardiograma de repouso. *Arq Bras Cardiol*, v. 80, 2003.
- 14 ELECTROCARDIOGRAM (ECG or EKG). American Heart Association, 2015. Last visited 5/6/2018. Disponível em: <http://www.heart.org/HEARTORG/Conditions/HeartAttack/DiagnosingaHeartAttack/Electrocardiogram-ECG-or-EKG_UCM_309050_Article.jsp>.
- 15 Michael A. Chen. *Electrocardiogram*. MedlinePlus, 2016. Last visited 5/6/2018. Disponível em: <<https://medlineplus.gov/ency/article/003868.htm>>.
- 16 AJAM, T. *Electrocardiography*. Medscape, 2017. Last visited 5/6/2018. Disponível em: <<https://emedicine.medscape.com/article/1894014-overview>>.
- 17 YANOWITZ, F. G. *The Standard 12 Lead ECG*. University of Utah School of Medicine, 2018. Last visited 5/6/2018. Disponível em: <<https://ecg.utah.edu/lesson/1#ecgwaves>>.
- 18 ALLEN, J. Photoplethysmography and its application in clinical physiological measurement. *Physiological Measurement*, v. 28, n. 3, p. R1, 2007. Disponível em: <<http://stacks.iop.org/0967-3334/28/i=3/a=R01>>.
- 19 JACQUES, S. L. Optical properties of biological tissues: a review. *Physics in Medicine & Biology*, v. 58, n. 11, p. R37, 2013. Disponível em: <<http://stacks.iop.org/0031-9155/58/i=11/a=R37>>.
- 20 FULL width half maximum. EDU.PHOTONICS.COM, 2017. Last visited 5/6/2018. Disponível em: <https://www.photonics.com/d4312/full_width_half_maximum>.
- 21 SAZONOV, E.; NEUMAN, M. R. *Wearable Sensors*. [S.l.]: Elsevier Inc., 2014.
- 22 CHEN, I. Using reflectometry for a ppg waveform. 2014.
- 23 TAMURA, T.; MAEDA, Y.; SEKINE, M.; YOSHIDA, M. Wearable photoplethysmographic sensors—past and present. *Electronics*, MDPI AG, v. 3, n. 2, p. 282–302, Apr 2014. ISSN 2079-9292. Disponível em: <<http://dx.doi.org/10.3390/electronics3020282>>.
- 24 SALEH, B. E. A.; TEICH, M. C. *Fundamentals of Photonics*. 2. ed. [S.l.]: John Wiley & Sons, Inc., 2007.
- 25 HALBRITTER, H.; WEBER, R.; STRUWING, S. *SFH 7050 - Photoplethysmography Sensor*. 2014.
- 26 HOROWITZ, P.; HILL, W. *The Art of Electronics*. 3. ed. [S.l.]: Cambridge University Press, 2015.
- 27 BUCCELLETTI, E.; GILARDI, E.; SCAINI, E.; GALIUTO, L.; PERSIANI, R.; BIONDI, A.; BASILE, F.; SILVERI, N. Heart rate variability and myocardial infarction: systematic literature review and metanalysis. PubMed, 2019. Disponível em: <<https://www.ncbi.nlm.nih.gov/pubmed/19694345>>.
- 28 ROY, R.; KAILATH, T. Esprit-estimation of signal parameters via rotational invariance techniques. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, v. 37, n. 7, p. 984–995, Jul 1989. ISSN 0096-3518.
- 29 MANOLAKIS, D. G.; INGLE, V. K.; KOGON, M. S. *Statistical and Adaptive Signal Processing*. [S.l.]: ARTECH HOUSE, INC., 2005.
- 30 REIS, P. M. G. I. *Detecção Robusta de Adulteração em Áudio Explorando a Forma Analítica e o Subespaço de Sinais Interferentes da Rede Elétrica*. Dissertação (Mestrado) — Universidade de Brasília, Brasília, 7 2016.

- 31 PROAKIS, J. G.; MANOLAKIS, D. K. *Digital Signal Processing*. 4. ed. [S.l.]: Pearson Education Limited, 2014.
- 32 BROWN, T.; WANG, M. M. An iterative algorithm for single-frequency estimation. *IEEE Transactions on Signal Processing*, v. 50, n. 11, p. 2671–2682, Nov 2002. ISSN 1053-587X.
- 33 WALL, R. W. Simple methods for detecting zero crossing. In: *Industrial Electronics Society, 2003. IECON '03. The 29th Annual Conference of the IEEE*. [S.l.: s.n.], 2003. v. 3, p. 2477–2481 Vol.3.
- 34 MathWorks. *dsp.ZeroCrossingDetector System object*. 2012. Last visited 10/6/2018. Disponível em: <<https://www.mathworks.com/help/dsp/ref/dsp.zerocrossingdetector-system-object.html>>.
- 35 MAXIM INTEGRATED. *MAX30100 Pulse Oximeter and Heart-Rate Sensor IC for Wearable Health*. [S.l.], 2014. Rev. 0.
- 36 OXullo Intersecans. *Arduino-MAX30100*. [S.l.]: GitHub, 2018. <<https://github.com/oxullo/Arduino-MAX30100>>.
- 37 FRITZING. Friends-of-Fritzing, 2016. Last visited 27/6/2018. Disponível em: <<http://fritzing.org>>.
- 38 SPARKFUN ELECTRONICS. *Arduino Pro Mini (DEV-11113)*. [S.l.], 2015.
- 39 PIETERS, A. *Fritzing-Custom-Parts / YwRobot Breadboard Power Supply.fzpz*. [S.l.]: GitHub, 2017. <<https://github.com/AchimPieters/Fritzing-Custom-Parts>>.
- 40 STROGONOV, R. Implementing pulse oximeter using max30100. 2017. Last visited 9/6/2018. Disponível em: <<https://morf.lv/implementing-pulse-oximeter-using-max30100>>.
- 41 SMITH, J. O. *INTRODUCTION TO DIGITAL FILTERS*. [S.l.]: Center for Computer Research in Music and Acoustics (CCRMA), Stanford University, 2007. <https://ccrma.stanford.edu/~jos/fp/DC_Blocker.html>. Last visited 10/6/2018.
- 42 PRETTZ, J. B. *Arquitetura MIMO para Pulseiras Inteligentes com Aplicação Móvel para Monitoramento de Baixo Custo de Pacientes no Pós-cirúrgico*. Dissertação (Mestrado) — Universidade de Brasília, Brasília, 10 2017.
- 43 YOU, E. *The Progressive JavaScript Framework*. [S.l.], 2018. Last visited 10/6/2018.
- 44 VUE JS. *VUEX*. [S.l.], 2018. Last visited 10/6/2018.
- 45 VUE JS. *Vue Router*. [S.l.], 2018. Last visited 10/6/2018.
- 46 THOMAS, J. *Bulma - Modern CSS framework based on Flexbox*. [S.l.]: GitHub, 2018. <<https://github.com/jgthms/bulma>>.
- 47 Chart.js. *Simple HTML5 Charts using the <canvas> tag*. [S.l.]: GitHub, 2018. <<https://github.com/chartjs/Chart.js>>.
- 48 Axios. *Promise based HTTP client for the browser and node.js*. [S.l.]: GitHub, 2018. <<https://github.com/axios/axios>>.
- 49 BASIC Metabolic Panel (BMP). Lab Tests Online, 2015. Last visited 10/6/2018. Disponível em: <<https://labtestsonline.org/tests/basic-metabolic-panel-bmp>>.
- 50 COMPLETE blood count (CBC). Mayo Clinic, 2018. Last visited 10/6/2018. Disponível em: <<https://www.mayoclinic.org/tests-procedures/complete-blood-count/about/pac-20384919>>.

APPENDICES

I. SOURCE CODE

This appendix presents the source code used in this work implemented in MATLAB®. The filters explained in Chapter 4 are presented, followed by the algorithms used in Chapter 6.

- Removing motion artifacts:

```
1 function signal_noart = remove_artifacts(signal)
2
3     % Calculates the differences of the signal
4     d = [0; diff(signal)];
5
6     % Compute valid sections based on a difference criteria
7     isValid = ~logical(abs( d ) > 100);
8
9     % Cuts invalid regions of the signal (motion artifacts)
10    signalCut = signal;
11    signalCut(~isValid) = NaN;
12    signalCut(isnan(signalCut(:))) = [];
13    indices = abs(signalCut)<100;
14    signalCut(indices) = [];
15
16    % Returns the valid regions of the signal
17    signal_noart = signalCut;
18
19 end
```

- DC Blocker:

```
1 function signal_out = dc_blocker(signal, alpha)
2
3     % Defines de numerator coefficients of the difference equation
4     b=[1 -1];
5
6     % Defines de denominator coefficients of the difference equation
7     a = [1 -alpha];    % alpha is the DC blocker ration
8
9     % Defines the initial condition as the first sample of the signal
10    zi = signal(1);
11
12    % Returns the signal filtered by difference equation defined by [b, a]
13    signal_out = filter(b,a,signal,zi);
14
15 end
```


- ESPRIT:

```

1 function result = estimator_esprit(raw_data, fs, timeWindow)
2
3     % Removes motion artifacts
4     signal_noart = remove_artifacts(raw_data);
5
6     % DC Blocker
7     signal_nodc = dc_blocker(signal_noart, 0.95);
8
9     % Low pass filter
10    fc = 4; % cutoff frequency of 4 Hz
11    [b,a] = butter(6,fc/(fs/2)); % 6th order Butterworth filter
12    signal_filt = filter(b,a,signal_nodc);
13
14    % Convergence time of the filter
15    filt_converg = floor(150*log10(fs));
16
17    % Option to plot the filtered signal
18    %plot(signal_filt(filt_converg:end))
19
20    % Calculates the correlation data matrix estimate
21    % in the time interval given
22    [~, R] = corrmtx(signal_filt(filt_converg:end), ...
23    floor((timeWindow*fs)/2), 'cov ');
24
25    % Returns 0 if frequency estimation is empty or
26    % the BPM estimation otherwise
27    if isempty(estFreqEsprit(R, fs))
28        result = 0;
29    else
30        result = estFreqEsprit(R, fs)*60;
31    end
32
33 end
34
35 function [freq] = estFreqEsprit(R, fs)
36
37     % Eigendecomposition
38     [E, ~] = eig(R);
39
40     % Number of rows on R
41     nRows = size(R, 1);
42
43     % Number of harmonics desired (2 * the actual number if real signal)
44     nHarmonics = 2;
45
46     % Selection matrices
47     J1 = [eye(nRows-1,nRows-1) zeros(nRows-1,1)];
48     J2 = [zeros(nRows-1,1) eye(nRows-1,nRows-1)];
49

```

```

50 % Signal subspace
51 Es = E(:, (nRows - nHarmonics + 1):nRows);
52
53 % Compute Phi
54 Phi = pinv(J1 * Es) * J2 * Es;
55
56 % Compute eigenvalues of Phi
57 eigenvaluesPhi = eig(Phi);
58
59 % Compute the frequencies from the eigenvalues of Phi
60 freqs = angle(eigenvaluesPhi);
61
62 % Extract the harmonic of interest
63 freq = freqs(freqs > 0)/(2 * pi) * fs;
64
65 end

```

- FFT:

```

1 function result = estimator_fft(raw_data, fs, timeWindow)
2
3 % Removes motion artifacts
4 signal_noart = remove_artifacts(raw_data);
5
6 % DC Blocker
7 signal_nodc = dc_blocker(signal_noart, 0.95);
8
9 % Low pass filter
10 fc = 4; % cutoff frequency of 4 Hz
11 [b,a] = butter(6,fc/(fs/2)); % 6th order Butterworth filter
12 signal_filt = filter(b,a,signal_nodc);
13
14 % Convergence time of the filter
15 filt_converg = floor(150*log10(fs));
16
17 % Option to plot the filtered signal
18 %plot(signal_filt(filt_converg:end))
19
20 % Zero pads the signal with 1000 zeros,
21 % cutting it in the time window defined
22 signal_zeros = [signal_filt(filt_converg:(timeWindow*fs), ...
23 + filt_converg);zeros(1000,1)];
24
25 % Calculates the FFT and the corresponding frequencies of the signal
26 [fft_data, freqs_data] = fft_signal(signal_zeros, fs);
27
28 % Option to plot the FFT
29 %plot(freqs_data, fft_data)
30
31 % Returns the BPM estimation

```

```

32     result = bpm_calc_fft(fft_data, freqs_data);
33
34 end
35
36 function bpm = bpm_calc_fft(signal, freqs)
37
38     % Defines interval equal to half of the spectrum
39     interval = signal(1:fix(length(signal)/2));
40
41     % Calculates the index of maximum amplitude in the FFT
42     [M, I] = max(interval);
43
44     % Calculates BPM from the dominant frequency of the signal
45     bpm = freqs(I)*60.0;
46
47 end

```

- Autocorrelation:

```

1  function result = estimator_corr(raw_data, fs, timeWindow)
2
3     % Removes motion artifacts
4     signal_noart = remove_artifacts(raw_data);
5
6     % DC Blocker
7     signal_nodc = dc_blocker(signal_noart, 0.95);
8
9     % Low pass filter
10    fc = 4; % cutoff frequency of 4 Hz
11    [b,a] = butter(6,fc/(fs/2)); % 6th order Butterworth filter
12    signal_filt = filter(b,a,signal_nodc);
13
14    % Convergence time of the filter
15    filt_converg = floor(150*log10(fs));
16
17    % Option to plot the filtered signal
18    %plot(signal_filt(filt_converg:end))
19
20    % Calculate the autocorrelation of the signal in the given timewindow
21    corr = calculate_correlation(signal_filt(filt_converg:, ...
22    (timeWindow*fs)+filt_converg));
23
24    % Returns the estimated BPM value
25    result = bpm_calc_corr(corr, fs);
26
27 end
28
29 function correlation = calculate_correlation(signal)
30
31     % Returns the autocorrelation of the input signal

```

```

32     correlation = xcorr(signal);
33
34     % Option to plot the autocorrelation
35     % plot(correlation)
36
37 end
38
39 function bpm = bpm_calc_corr(corr, fs)
40
41     % Computes the peak location of the autocorrelation
42     % with a minimum distance of 1/3 second in between
43     % to avoid detection of secondary peaks
44     [~,locs]=findpeaks(corr, MinPeakDistance ,fs/3);
45
46     % Option to plot the peaks locations
47     % findpeaks(corr, MinPeakDistance ,fs/3)
48
49     % Estimates BPM from the average lag between peaks
50     % corresponding to the signal s period
51     bpm = mean(fs./diff(locs))*60.0;
52
53 end

```

- Zero crossing:

```

1 function result = estimator_zc(raw_data, fs, timeWindow)
2
3     % Removes motion artifacts
4     signal_noart = remove_artifacts(raw_data);
5
6     % DC Blocker
7     signal_nodc = dc_blocker(signal_noart, 0.95);
8
9     % Low pass filter
10    fc = 4; % cutoff frequency of 4 Hz
11    [b,a] = butter(6,fc/(fs/2)); % 6th order Butterworth filter
12    signal_filt = filter(b,a,signal_nodc);
13
14    % Convergence time of the filter
15    filt_converg = floor(150*log10(fs));
16
17    % Option to plot the filtered signal
18    %plot(signal_filt(filt_converg:end))
19
20    % Calculates the zero crossings of the signal
21    zc = calculate_zero_crossing(signal_filt(filt_converg:end), fs*timeWindow);
22
23    % Returns the estimated BPM value
24    result = bpm_calc_zc(zc, timeWindow);
25

```

```

26 end
27
28 function zero_crossings = calculate_zero_crossing(signal, samples)
29
30     % Returns the number of zero crossings of the signal
31     zcd = dsp.ZeroCrossingDetector;
32     zero_crossings = zcd(signal(1:samples));
33
34 end
35
36 function bpm = bpm_calc_zc(zero_crossings, timeWindow)
37
38     % Estimates BPM by the counting the amount of zero crossings
39     % in the time interval considered divided by 2
40     bpm = (zero_crossings/2)*(60.0/timeWindow);
41
42 end

```

- Peak detection:

```

1 function result = estimator_pk(raw_data, fs, timeWindow)
2
3     % Removes motion artifacts
4     signal_noart = remove_artifacts(raw_data);
5
6     % DC Blocker
7     signal_nodc = dc_blocker(signal_noart, 0.95);
8
9     % Low pass filter
10    fc = 4; % cutoff frequency of 4 Hz
11    [b,a] = butter(6,fc/(fs/2)); % 6th order Butterworth filter
12    signal_filt = filter(b,a,signal_nodc);
13
14    % Convergence time of the filter
15    filt_converg = floor(150*log10(fs));
16
17    % Option to plot the filtered signal
18    % plot(signal_filt(filt_converg:end))
19
20    % Calculate the number of peaks in the given time interval
21    pk = calculate_peaks(signal_filt(filt_converg:end), fs*timeWindow, fs);
22
23    % Returns the estimated BPM value
24    result = bpm_calc_pk(pk, timeWindow);
25
26 end
27
28 function num_peaks = calculate_peaks(signal, samples, fs)
29
30     % Computes the number of peaks in the time interval

```

```

31     % with a minimum distance of 1/3 second in between
32     % to avoid detection of secondary peaks
33     pks = findpeaks(signal(1:samples), MinPeakDistance , fs/3);
34
35     % Option to plot the peaks locations
36     %findpeaks(signal(1:samples) , MinPeakDistance , fs/3)
37
38     % Returns the amount of peaks detected in the interval
39     num_peaks = length(pks);
40
41 end
42
43 function bpm = bpm_calc_pk(num_peaks, timeWindow)
44
45     % Estimates BPM from the number of peaks corresponding
46     % to periods in the time intervall given
47     bpm = num_peaks*(60.0/timeWindow);
48
49 end

```