

DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

**SMART SPORTSWEAR DESIGN THAT CAN
DETECT VITAL PARAMETERS**

by
Ömür CERAN

May, 2024
İZMİR

SMART SPORTSWEAR DESIGN THAT CAN DETECT VITAL PARAMETERS

**A Thesis Submitted to the
Graduate School of Natural And Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Master of Science in
Department of Electric and Electronic Engineering**

**by
Ömür CERAN**

**May, 2024
İZMİR**

M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled "**SMART SPORTSWEAR DESIGN THAT CAN DETECT VITAL PARAMETERS**" completed by **Ömür CERAN** under supervision of **Ass. Prof. ÖZGÜR TAMER** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

.....
Ass. Prof. Özgür TAMER

Supervisor

.....
Assoc. Prof. Ömer Akgüller

Jury Member

.....
Assoc. Prof. Yavuz Şenol

Jury Member

Prof. Dr. Okan FISTIKOĞLU
Director
Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENTS

I am grateful for the chance to express my profound appreciation to my supervisor, Özgür Tamer, for his invaluable guidance, unwavering encouragement, and tender mentoring during this research project. This thesis would not have taken this form without his vast knowledge and unmatched vision.

Ömür CERAN

SMART SPORTSWEAR DESIGN THAT CAN DETECT VITAL PARAMETERS

ABSTRACT

The field of wearable electronics is rapidly growing, and it has only recently started to offer consumer electronics consumers valuable commercial products. Biopotential signals are predicted to revolutionize wearable systems when used as biofeedback or command commands. There are many technologies, including point-of-care health monitoring systems, brain-computer interfaces, and rehabilitation aids. Textile electrodes have been developed as a result of approximately ten years of research on electrodes, since they are considered to be an essential part of such things. This research examines wearable sports technologies that are capable of detecting key metrics. The ECG and the acceleration-based Inertial Measurement Unit (IMU) sensors from this project. Smart textile products are used for data collection and testing. In this thesis, Decision Tree, Gradient Boosted Tree, Logistic Regression, Hidden Markov Model, Naive Bayes, Random Forest and Support Vector Machine methods are used for motion classification and the performances of these methods are shown in tables and graphs. Considering the results obtained, Hidden Markov Model gave the best result in motion classification. Running and jumping motion classification can be easily performed with the data obtained using this method. However, since very good results could not be obtained for Stairs up and Down, the classification of these movements could not be done completely.

Keywords: ECG, Smart Textile, IMU based acceleration , Bluetooth low energy, Machine Learning

HAYATI PARAMETRELERİ ALGILAYABİLEN AKILLI SPOR KIYAFETİ

TASARIMI

ÖZ

Giyilebilir elektronik konusu hızla genişliyor ve son zamanlarda tüketici elektroniği pazarına karlı ticari ürünler sağlamaya başladı. Biyo-geribildirim veya komut komutları olarak giyilebilir sistemlerde biyopotansiyel sinyallerin kullanılmasının devrim yaratacağı tahmin edilmektedir. Beyin-bilgisayar arayüzleri, bakım noktası sağlık izleme sistemleri, rehabilitasyon araçları gibi çok sayıda teknoloji vardır. Elektrotlar bu tür öğelerin çok önemli bir bileşeni olarak görüldüğünden, yaklaşık on yıldır araştırılmakta ve bu da tekstil elektrotlarının geliştirilmesine yol açmaktadır. Elektrokardiyografi(EKG) ve Atalet ölçüm birimi(IMU) tabanlı ivmelenme gibi birkaç sensör vardır. Akıllı tekstil ürünleri test etmek ve veri almak amacıyla kullanılmaktadır. Bu tezde hareket sınıflandırması için Karar Ağacı, Gradyan Destekli Ağaç, Lojistik Regresyon, Gizli Markov Modeli, Naive Bayes, Rastgele Orman ve Destek Vektör Makinesi yöntemleri kullanılmış ve bu yöntemlerin performansları tablo ve grafiklerle gösterilmiştir. Elde edilen sonuçlara bakıldığından hareket sınıflandırmada en iyi sonucu Saklı Markov Modeli vermiştir. Bu yöntem kullanılarak elde edilen veriler ile koşma ve zıplama hareket sınıflandırması rahatlıkla yapılmaktadır. Ancak Merdiven Çıkma ve İnme için çok iyi sonuçlar elde edilemediğinden bu hareketlerin sınıflandırılması tam olarak yapılamamıştır.

Anahtar kelimeler: EKG, Akıllı Tekstil, IMU tabanlı ivme, Bluetooth düşük enerji, Makine Öğrenmesi

CONTENTS

	Page
M.Sc THESIS EXAMINATION RESULT FORM.....	ii
ACKNOWLEDGEMENTS	iii
ABSTRACT	iv
ÖZ	v
LIST OF FIGURES	xi
CHAPTER ONE – INTRODUCTION	1
1.1 Aim of the study	1
1.2 Thesis Outline	2
CHAPTER TWO – RELATED WORK	3
CHAPTER THREE – METHODOLOGY.....	10
3.1 Measurement.....	12
3.1.1 Electrocardiogram measurement	12
3.1.2 Inertial Measurement Unit Sensor	15
3.2 Microcontroller and Communication	18
3.2.1 RSL10 microcontroller.....	18
3.2.2 Bluetooth low energy	19
3.3 Schematic and PCB Design.....	19
3.3.1 Schematic Design.....	19
3.3.2 PCB Design	21
3.4 Tshirt Design.....	23
3.5 Algorithm Design.....	24
3.5.1 General Algorithm and Bluetooth Low Energy Services.....	24
3.6 Machine Learning Methodology	27
3.6.1 Decision Tree	27

3.6.2 Gradient Boosted Trees	29
3.6.3 Logistic Regression	30
3.6.4 Hidden Markov Models.....	32
3.6.5 Naive Bayes	34
3.6.6 Random Forest	35
3.6.7 Support Vector Machine	37
3.6.8 Performance Metrics.....	38
CHAPTER FOUR – APPLICATIONS AND RESULTS	42
4.1 Pattern measurement	44
4.1.1 Running Pattern measurement	45
4.1.2 Jumping Pattern measurement	47
4.1.3 Stairs Pattern measurement	49
4.1.3.1 Stairs Down Pattern	49
4.1.3.2 Stairs Up Pattern	51
CHAPTER FIVE – ANALYSIS.....	54
5.1 Statistical Analysis	54
5.1.1 Running-Jumping Pattern Analysis.....	55
5.1.2 Running-Stairs Up and Down Pattern Analysis.....	56
5.1.3 Jumping-Stair Up and Down Pattern Analysis	57
5.1.4 Stairs Up and Down Analysis	57
5.1.5 Running BPM analysis.....	58
5.1.6 Jumping BPM analysis.....	59
5.1.7 Stairs BPM analysis.....	59
5.1.7.1 Stairs Up BPM analysis	59
5.1.7.2 Stairs Down BPM analysis	60
5.2 Machine Learning Analysis and Discussion	61
5.2.1 Discussion	69
CHAPTER SIX – CONCLUSION	77

REFERENCES.....	80
------------------------	-----------

LIST OF FIGURES

	Page
Figure 3.1 Block diagram of the system	10
Figure 3.2 Heart rate measurement setup	12
Figure 3.3 ECG Waveform (Manullang, Simanjuntak & Ramdani , 2019)	13
Figure 3.4 ECG Electrodes Positions.....	13
Figure 3.5 IMU sensor circuit	15
Figure 3.6 RSL10 Block Diagram (Anonim (RSL10), 2020)	18
Figure 3.7 Block Design	19
Figure 3.8 Schematic Design	20
Figure 3.9 Circuit Design	21
Figure 3.10 Bottom and Top view of PCB.....	22
Figure 3.11 3D Bottom and Top view of PCB.....	22
Figure 3.12 Tshirt view inside	23
Figure 3.13 Tshirt view outside.....	24
Figure 3.14 Bluetooth Low Energy Services.....	25
Figure 3.15 BLE and HW connection	26
Figure 3.16 General Algorithm	26
Figure 3.17 Real Bottom and Top view of PCB	27
Figure 4.1 ECG signal	42
Figure 4.2 ECG signal(zoom)	43
Figure 4.3 IMU sensor position	43
Figure 4.4 Running Pattern and Activity Diagram	45
Figure 4.5 Running Pattern ECG and Activity Diagram.....	46

Figure 4.6 ECG Running Pattern and Y axis.....	46
Figure 4.7 Jumping Pattern and Activity Diagram	47
Figure 4.8 Jumping Pattern ECG	48
Figure 4.9 Jumping Pattern ECG (Zoom).....	48
Figure 4.10 Jumping Pattern ECG and X,Y and Z axis.....	48
Figure 4.11 Stairs Down Pattern and Activity Diagram	49
Figure 4.12 Stairs Down ECG Pattern and X,Y and Z axis.....	50
Figure 4.13 Stairs Down ECG Pattern	50
Figure 4.14 Stairs Down ECG Pattern	51
Figure 4.15 Stairs Up Pattern and Activity Diagram	51
Figure 4.16 Stairs Up ECG Pattern and Activity Diagram.....	52
Figure 4.17 Stairs Up ECG Pattern	52
Figure 4.18 Stairs Up ECG Pattern	53
Figure 5.1 Running Pattern vs BPM comparision	59
Figure 5.2 Jumping Pattern vs BPM comparision	59
Figure 5.3 Stairs Up Pattern vs BPM comparision	60
Figure 5.4 Stairs Down Pattern vs BPM comparision	60
Figure 5.5 Confusion matrices for activity time series classification (on the left) and ECG time series (on the right) for decision tree.	62
Figure 5.6 Confusion matrices for activity time series classification (on the left) and ECG time series (on the right) for XGBoost.	63
Figure 5.7 Confusion matrices for activity time series classification (on the left) and ECG time series (on the right) for logistic regression.....	64
Figure 5.8 Confusion matrices for activity time series classification (on the left) and ECG time series (on the right) for hidden Markov model.	65

Figure 5.9 Confusion matrices for activity time series classification (on the left) and ECG time series (on the right) for Naive Bayes approach.	66
Figure 5.10 Confusion matrices for activity time series classification (on the left) and ECG time series (on the right) for random forests.	67
Figure 5.11 Confusion matrices for activity time series classification (on the left) and ECG time series (on the right) for support vector machines.	68

CHAPTER ONE

INTRODUCTION

Significant progress has been made in the field of sports science recently, particularly in wearable technology. The development of wearable technology has completely changed the way we track and examine physiological data from athletes. When worn as part of clothing or integrated into portable equipment, wearable sensors can continuously monitor a variety of physiological indicators, allowing for the real-time and comprehensive collection of data.

Wearable electronic devices have greatly decreased in size due to advances in materials science and electronic engineering, and their shape has changed from being thick and inflexible to being thin and flexible. The wireless system is heavily utilized in the transmission of vital signs monitoring, movements, etc., and the user-machine interface becomes more friendly (Zysset, Kinkeldei, Munzenrieder, Cherenack & Troster, 2012).

1.1 Aim of the study

Tao, (2005) presents that a recent advancement in electronics, wearable electronics, also known as wearable computing involves the integration of devices into apparel, eyewear, and smart watches while also shrinking their physical dimensions. Wearable smart technology is quite common these days. In the past, it was common practice to start with smart textile objects.

Hu, Kaynak, & Li, (2005) research that the realm of intelligent fabrics, alternatively recognized as electronic textiles (e-textiles), explores the integration of functional materials with conventional clothing. This innovative domain seeks to seamlessly incorporate diverse functionalities into garments, including sensors, energy harvesters, antennas, and advanced textiles designed for self-heating and cooling.

Khan, Hussain, Nur & Willander, (2014) presents that smart textiles also make a mark in the realm of fashion applications, ushering in a new era where technology converges with style.

Athletes can now provide a variety of real-time data during training sessions and competitions thanks to the development of sophisticated sensors and equipment. Electrocardiogram (ECG) signals and activity patterns are among the factors included in these data. We can learn more about the variability in heart rate in athletes and keep an eye on their cardiac health while exercising by collecting ECG data. We may also evaluate the body movements of athletes by examining their acceleration, which gives us important details about their technique, balance, and coordination. Sensors provide information about cardiac activity, accelerometers measure patterns of movement, and ECG sensors monitor cardiac activity.

1.2 Thesis Outline

This thesis consists of five chapters. Chapter One presents the Introduction section in order to provide information, including the explanation of smart sport technology, the purpose of the vital parameter measurement. Related works on techniques and example usage are presented in Chapter Two. Chapter three informs about proposing measurement system on body. Chapter Four includes the applications, their results. Chapter Five shows analysis and Chapter Six is conclusions.

CHAPTER TWO

RELATED WORK

In this section, literature research and similar studies are examined. Although there are many different studies, similar and close studies have been discussed. The methods that are suitable for the basic design and can be used are collected in this section. In addition, similar studies were categorized sequentially by roughly grouping them.

Mehta , Nazir, Trohman, & Volgman, (2015) proposes that portable ECG devices to quickly and easily evaluate heart problems and findings. Prior to the addition of Rhythm Strips. Within the category of difficult-to-understand rhythm strips, less than half of the participants agreed between the hospital cardiac monitor and the rhythm strips of portable ECG equipment.

Sun, Yi, Li, & Li, (2017) is researched that a redesigned sports H-shirt, equipped with a computerized heart rate threshold lactate threshold, has been suggested and implemented. The high degree of flexibility guarantees stable skin-electrode contact, inhibits the measurement of discomfort, and uses Bluetooth low power technology (BLE) to transfer data between the T-shirt and the smartphone. An infrastructure based on mobile phones has been developed for ECG analysis and exercise evaluation.

Baig, Gholamhosseini, & Connolly, (2013) presents a diverse array of more than 120 wearable ECG monitoring systems designed for adults, including solutions that incorporate wireless, mobile, and remote technologies. As a result, devices were classified into wireless, mobile, smart wearable, and associated signal analysis methods.

Spanò, Di Pascoli, & Iannaccone, (2016) suggests a wearable wireless system for monitoring ECGs that is connected to an Internet of Things (IOT) platform. It has a

long battery life, integrates several nodes and applications, and generates high-quality ECG signals. In this research, they made use of mobile and wireless communication technologies, such as BLE, which allow devices to transmit data to remote terminals like smartphones.

Guzik, & Malik, (2016) conducts a research to explore mobile ECG technology as part of their study, a smartphone, a mobile device, an ECG. Mobile devices are compact computers with numerous alternatives for data delivery, a suitable display, enough computing power, and enough data storage. The use of mobile ECG devices does not yet adhere to recognized clinical standards or approved criteria. The new mobile ECG technology should soon be the subject of extensive clinical investigation to show its true benefit.

Jung, Lee, Lee, & Kim, (2015) presents on ubiquitous healthcare and proposes an AMI diagnosis approach and software solution that could help with face-to-face patient-physician contact. Based on ISO/IEEE 11073, a protocol was used to establish the monitoring and diagnostic system. When information is transferred from the patient's smartphone to a hospital server, the medical team reviews the patient's biosensor data to assess the patient's condition and provide the necessary medical care.

Dias, & Paulo Silva Cunha, (2018) developed that a system also includes a heartbeat (HR) tracking device. In addition, a server, a Web page, a BLE device, a smartphone, and other remote monitoring components have been added to the system. Following a discussion of the most important vital signs for health evaluation using wearable health devices, the origins, effects on health, monitoring, and most recent scientific advancements in the field (ECG, blood pressure, etc.) of each vital sign are discussed: heart rate, blood pressure, respiration rate, blood oxygen saturation, blood glucose, skin perspiration, and capnography are discussed.

Patel, Park, Bonato, Chan, & Rodgers , (2012) presents physiological elements,

such as muscular-skeletal and cardiovascular problems, that can be tracked to help ensure a worker's physiological health, as well as their activities and degree of weariness.

Van Loon, van Zaane, Bosch, Kalkman, & Peelen, (2015) suggests that to follow ergonomic guidelines, researchers were pushed to improve their well-being, safety and health. The presented study includes technologies that prioritize early disease diagnosis, house recovery, therapy success assessment, and safety.

Ozkan, Ozhan, Karadana, Gulcu, Macit, & Husain, (2019) proposes an inventive design for an elastic singlet that has been altered to include fabric electrodes (TE). This work presents a unique architecture for a wearable Tele-ECG and heart rate (HR) monitoring system. It consists of a flexible singlet that has been rebuilt with textile electrodes (TEs), textile threads, Velcro, sponges, and an ECG circuit. To further enhance the remote monitoring system, a server, a web page, a smartphone, and Bluetooth low energy (BLE) have been included.

Guo, Peterson, Qureshi, Kalantar Mehrjerdi, Skrifvars, & Berglin, (2011) conducted research on the breathing analysis rate capabilities of an IMU belt. The monitor belt provides an approach for the surveillance of vital indicators in hospital settings or home care by tracking the rate of breathing and heart rate in addition to the ECG data. Currently, cloth knitting sensors are believed to be used to detect respiration of patients without interfered with their daily activities.

Di Tocco, Raiano, Sabbadini, Massaroni, Formica, & Schena, (2021) present a suggested system that incorporates four conductive sensors into the user's chest, allowing it to retrieve breathing activity through the deformations the sensors cause when the rib cage expands and contracts cyclically. The left side of the chest wall is implanted with an IMU to track heart rate.

Beck, Laufer, Krueger-Ziolek, & Moeller, (2020) conducted this systematic study

to find out whether continuous noninvasive respiration tracking reduces crucial occurrences in healthcare facilities and enhances early identification of patient deterioration. Cardiovascular rate (HR) and respiratory rate (RR) have attracted a great deal of interest in these settings because they are closely related to a range of physiological and disease-related characteristics of patients (e.g., early identification of critical events) and a variety of external pressures. The most advanced wearable RR monitors use methods that account for the periodic growth and shrinkage of the rib cage during respiratory activities.

Casale, Pujol & Radeva, (2011) proposes as a prominent subset of the broader "smart textile industry," "wearable technology" can be interpreted as a recently emerged technological domain that emerged from the combination of multiple well-established industries, such as barrier insulation, technical fabrics, flexible and compressed electronics, etc., are given access to a novel set of features that are generated from wearable data, which are computationally competitive and able to provide classification performance that is comparable to that of wearable systems at the forefront. The proposed features may be computed in real time and provide the quantities involved in classification with a physical meaning. The novel set of characteristics has been validated by a reliable study that contrasts it with most of the characteristics commonly used in accelerometer-based physical activity identification.

Nweke et al. , (2019) study that multisensor fusion analysis of the detection and tracking of human activity using a gyroscope and an accelerometer. They used significant voting systems and random forest (RF) to get good results after attaching several sensors to various body sites, including the wrist, chest, ankle, and hip.

Yang, et al., (2009) proposed that a evaluation technique known as leave-one-person-out cross-validation was implemented. Data from six individuals were used for training purposes, while data from one person were reserved for testing. A single accelerometer, attached to the wrist, was utilized along with a Neuro-Fuzzy classifier. In addition to performing frequency domain analysis, standard statistical

features were derived during the attribute computation phase.

Long, Yin, & Aarts, (2009) recommended a study aimed at analyzing the daily activity patterns of users and, in conjunction with an assessment of energy expenditure, facilitate straightforward and inspiring lifestyle adjustments. It focused on five prevalent daily physical activities: walking, jogging, cycling, driving, and sports. Activity classification was performed using the CAR method in conjunction with the Decision Tree (DT) classifier.

ÇALIŞAN, & Talu, (2020) proposed the study assessed techniques for recognizing basic body movements like walking, running, stopping, sitting, and lying down from IMU data. Traditional methods (ANN, SVM, k-NN) and contemporary ones (Convolutional Neural Networks-ESA) were compared. Data collection involved placing accelerometers on ten individuals to capture movements, followed by creating short-term windows from these data. Traditional methods extracted features from these windows, while deep learning transformed the data into 2D matrices. Architectures employing ANN, SVM, k-NN, and CNN were developed for classification. The results showed high precision, with ANN achieving 99% and CNN 95% accuracy in mapping the IMU data to activities.

Yen, Liao & Huang, (2020) present a wearable gadget that uses a deep learning algorithm to identify six common human activities: walking, walking upstairs, walking downstairs, sitting, standing, and lying. This device, in contrast to the traditional wrist-worn one, is designed to be worn around the waist. A three-axis accelerometer, a gyroscope, and an inertial sensor with a microcontroller make up its hardware. For activity recognition, the software uses motion signal capture, signal normalization, and a feature learning technique built around a 1D convolutional neural network. The usefulness of the proposed algorithm was confirmed by experimental data collected from 21 people performing common movements, who achieved recognition rates of 97.19% and 93.77% with recorded data and 98.93% and 95.99% in training and testing samples, respectively, with UCI dataset activities.

Arif & Kattan, (2015) research a method of using wireless sensors to track physical activity by examining acceleration signals that are obtained from sensors placed on people's ankles, wrists, and chests. It suggests a feature set for successfully identifying twelve unique activities based on time domain features of these signals. Nine participants in the study, whose body mass index (BMI) was 25.11 kg/m^2 and whose average age was 27.2 years, participated in a variety of activities, including vacuuming, ironing clothes, jumping rope, sitting, standing, walking, running, cycling, Nordic walking and climbing and descending stairs. The results show a strong performance, with the recall and precision for each task exceeding 95% and an amazing overall classification accuracy of 98% when using all available sensor data. This method shows great promise for helping medical professionals accurately gauge levels of physical activity.

Bayat, Pomplun & Tran, (2014) conduct a study that delves into utilizing acceleration data from a user's cell phone to discern specific human physical activities. They propose an identification system that separates the body acceleration component from the gravity-induced acceleration in raw data using a newly developed digital low-pass filter. This system was trained and tested through real-life experiments with numerous individuals. Diverse statistical features were used to evaluate multiple classifiers, encompassing both low-frequency and high-frequency data components. By amalgamating five classifiers, each proficient in detecting particular activities, the study aimed to determine an optimal combination. Their findings revealed that employing the average probability as the fusion approach could yield a total precision rate of 91.15%.

Wearable smart textile technology has been developed, as seen in this article. ECG, temperature and accelerometer sensors, among others, have been incorporated into the textile product. Using BLE wireless technology transmits to a smart phone the temperature of the skin, as well as vital signals including heart rate, breathing, ECG, and active motion. In this thesis, Decision Tree, Gradient Boosted Tree,

Logistic Regression, Hidden Markov Model, Naive Bayes, Random Forest and Support Vector Machine methods are used for motion classification

Comparing the suggested approach for movement and heart rate detection with comparable works reveals both similarities and differences. In most cases, heart rate and movement are integrated into T-shirts, even if several projects do. In this study, jumping, running, and stair up and down movements are observed using an IMU sensor. The determination of movement is simply explored as it exists or not in related works. Measurements taken in sync with an IMU and an ECG sensor yield information on heart rate during specific movements. Some daily activities can also be recorded and extracted from the ECG and IMU sensors. Good prospects for physical activity are satisfied by the integration in t-shirts.

CHAPTER THREE

METHODOLOGY

In this study, a tiny circuit board containing a microcontroller that can communicate over Bluetooth is paired with an IMU and an ECG sensor. To track an athlete's activity and ECG, a board with electrodes and a battery is worn on a T-shirt. The ECG sensor we utilized is the Analog Devices AD8232, which needs three electrodes placed in a certain way(Devices,2012) as shown in figure 3.2. The microcontroller's analog input is linked to the ECG sensor. The recommended IMU sensor is the InvenSense MPU6050, a motion sensor with nine axes, three accelerometers, and three gyroscopes that can also be connected using an interface for I2C(Fasate, 2023). The suggested microcontroller is the Onsemi RSL10, which transfers data obtained from the sensors using the Bluetooth Low Energy protocol. Figure 3.1 displays the system's block diagram.

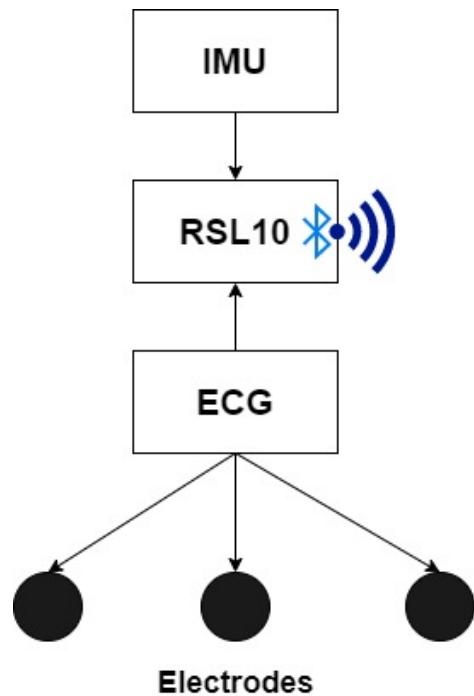


Figure 3.1 Block diagram of the system

Measurement devices include an AD8232 ECG sensor, MPU6050 acceleration sensor, a low power microcontroller which is RSL10 and BLE communication

together. The Bluetooth Low Energy protocol is used by a microcontroller to transfer data from an ECG sensor and IMU to a mobile device. The AFE series of highly integrated circuits includes the AD8232 microchip, which is used to process and collect biomedical information in addition to producing an analog ECG. An analog readout can be obtained from this electrical activity by charting it as an ECG. The AD8232 Single Lead Heart Rate Monitor functions as an op-amp to make it easier to quickly extract a clear signal from the PR and QT Intervals, which can be particularly noisy on ECG. The central signal processing unit for the ECG and several other biopotential measurement systems is the heart ECG monitoring sensor module, AD8232 (Devices, 2012).

The MPU-6050 is equipped with a 3-axis MEMS accelerometer, which is capable of measuring acceleration in the Z, Y, and X directions. The range of its complete scale is $\pm 2g$, $\pm 4g$, $\pm 8g$, or $\pm 16g$. A capacitive sensor element is used by the accelerometer to measure acceleration. The movable mass that is suspended between the two fixed plates makes up the sensing element. When the accelerometer is accelerated, the moving mass is displaced, which affects the capacitance between the plates. The MPU-6050's ADC digitizes the electrical signal that is created from this change in capacitance (Fasate, 2023). The I2C communication protocol is used to communicate with the IMU sensor (MPU6050).

One of the critical metrics of the battery of the device in this project is power consumption. Through analog-to-digital conversion (ADC), the microcontroller and AD8232 ECG sensor can communicate. As a result, they are one of the most crucial, if not the most important, aspects of e-textile technology. This section gives a comprehensive review of the many processes and materials that are used to make electroconductive fabrics. The creation of e-textiles is based on the stable fusing of conductive materials with fibers and fabrics.

3.1 Measurement

3.1.1 *Electrocardiogram measurement*

Dry electrodes are used to assess the electrocardiogram (ECG; AD8232). Electrocardiogram readings are generated in analog form, and in this project, the RSL10 was linked to the AD8232 module in order to obtain the ECG waveform. The AD8232 single-lead heart rate monitor works as an op amp to help with the easy acquisition of a clear signal from the PR and QT intervals. This is a three-electrode ECG. Electrodes should be placed on the chest wall, with equal distance between it and the heart, for optimal effect (instead of the targeted limbs). As shown in figure 3.2, data is obtained from the AD8232 ECG sensor using the Onsemi RSL10 microcontroller.

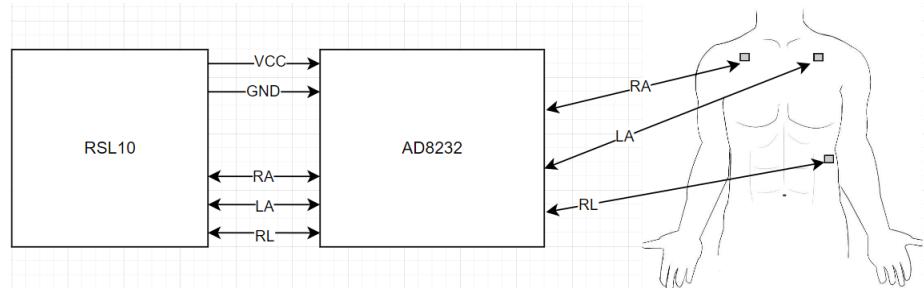


Figure 3.2 Heart rate measurement setup

ECG sensors are instruments that are commonly utilized in the health sector to detect the muscular contractions that occur in the heart during beats per minute by putting electrodes applied to the outer layer of skin. A signal or pulse is produced as a result of this capture. Various waves may be seen depending on where the heart activity monitor is placed. Every variation in a lead can be seen from different perspectives as a depiction of the same phenomena. Professional ECG sensors require 10–12 electrodes, although an ECG signal can be obtained with as little as three electrodes (Potter, 2023). The heart contracts as a result of depolarization, which occurs with each heartbeat. It is possible to detect this electrical activity on the skin. Figure 3.3 displays this signal. P waves are produced during the depolarization process. The S section comes after the initial negative deflection, or Q wave, and the highest wave, or R wave, is commonly

used to measure the average beat per minutes. Ventricular repolarization is represented by the T wave. Biphasic, positive, or negative T waves are all possible (Ashley & Niebauer, 2004).

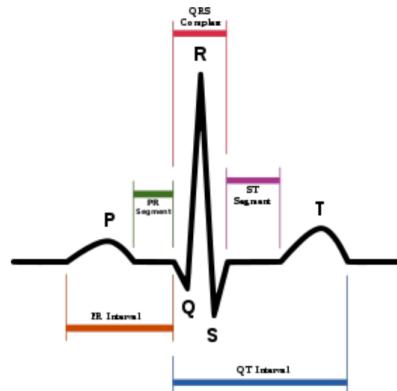


Figure 3.3 ECG Waveform (Manullang, Simanjuntak & Ramdani , 2019)

A heart signal from the first lead, or the signal from the three electrodes put on the chest as shown in figure 3.4, is what this research aims to be able to collect. This monitoring culminates in a wave that has a plethora of information that can be utilized to generate a rough evaluation of how the heart is functioning. This component measures heart rate using the ECG sensor AD8232.

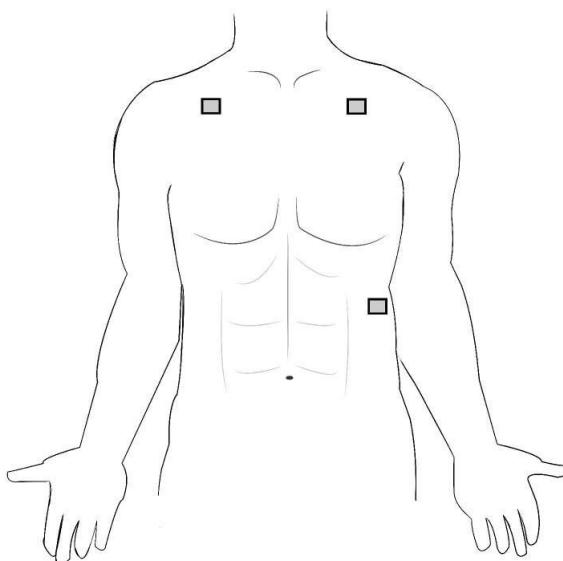


Figure 3.4 ECG Electrodes Positions

A moderately cheap sensor board is utilized for the purpose of measuring cardiac electrical activity. An electrocardiogram, or ECG, can be utilized to monitor this sensor activity and generate a reading in analog form. As an op-amp, the AD8232 One-Source Heart Rate Monitoring System helps to quickly and easily extract clean data obtained from the QT and PR intervals. The signals of the three electrodes will produce a graph showing the electrical activity of the heart as a series of hills and valleys. The signal graph is composed of five segments, PQRST. Each of these parts models the cardiac process, making it possible to spot abnormalities and disorders on the PQRST graph (Manullang, Simanjuntak & Ramdani , 2019). The graphic representation of PQRST is shown in figure 3.3. P waves are produced during the depolarization process. Through a process known as depolarization, the heart beats and creates atrial polarization contractions. Part Q, the initial negative deflection, activates interventricular septum depolarization from left to right and lasts a normal amount of time. The R wave, which is the highest wave, is usually used to calculate the typical heart rate. The S section follows the second R and is the first a negative deflection. Known as the S wave, deflection. The T wave, which signifies ventricular repolarization, is frequently less than 5 mm wide and 10 mm high in the extremity leads or precordial leads. Biphasic, positive, or negative T waves are all possible. Skin perspiration is the sole electrolyte layer found in dry electrodes. It might be difficult to comprehend the precise nuances of how contact or noncontact dry electrodes interact with one another. Dry electrodes can provide a galvanic or capacitive electrical current path when in close proximity to the skin. Even while wet electrodes are the standard in clinical settings, gel-free, "dry" electrodes can be great candidates for wearable, long-term, point-of-care personal health monitoring applications and many other related systems. Wearable conductive fabrics provide a useful alternative in this case.

We obtained ECG data from the AD8232 sensor's output and performed ADC (Analog-to-Digital Converter) readings using the RSL10 development board. The ADC functionality required to transform the analog signals from the AD8232 into digital data was provided by the RSL10 development board. We set up the RSL10

board's ADC settings using the ON Semiconductor IDE to guarantee precise and dependable data collection. These settings included variables that were specific to the needs for ECG data acquisition, such as sampling rate, resolution, and input range. After that, the obtained ECG data was recorded and stored in a log file. We were able to record and store the raw ECG data for later processing or analysis thanks to this logging procedure. We created a graphical representation of the recorded log file in order to see the ECG data. We were able to extract the ECG waveform by graphing the data points, which gave us important information about the patient's cardiac activity.

3.1.2 Inertial Measurement Unit Sensor

The IMU sensor collects data on the necessary portion of the user and uses it for computation activities. to collect the three-axis activity value, also known as the gyro value, and convert it to digital output.

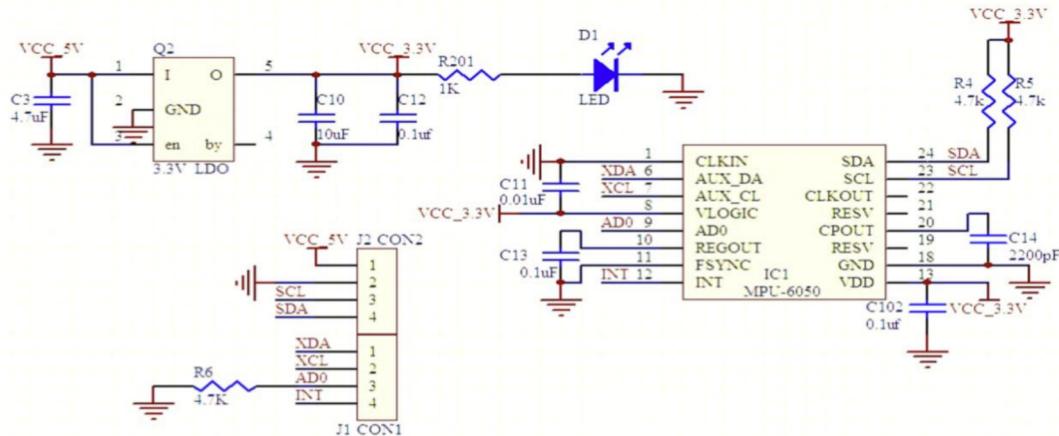


Figure 3.5 IMU sensor circuit

Gyroscope, magnetometer, and accelerometer are the three primary sensors of an IMU system. All three are worthwhile on their own, but when combined, they offer a great way to gauge athletic activity in general. Since the degree of accuracy and precision depends on the computations of algorithms when they are established, the development of a system of checks and balances is necessary. While you move around, an accelerometer frequently detects movement activities. Additional activity

values may have an impact on the orientation's accuracy. Another problem with accelerometers is noise, which is an unpleasant interruption in an electrical transmission. Even with a low pass filter, the accelerometer can measure any angle, but its readings are noisy and have a small degree of inaccuracy.

To maintain a stable environment, the initialization step begins by masking all interruptions. Then, all active interrupts are turned off, and all active interrupt sources are cleared. The application also makes it simple to re-flash by waiting for the DIO12 pin to be activated. The 48 MHz crystal is then set up, and the VDDRF (RF power supply) is powered on and setup after that. Before continue, the application waits for the VDDRF source to open. In order to gain access to RF equipment, RF power switches are activated and RF isolation is lifted. The desired prescaler value for the oscillator divider is set, and the 48 MHz oscillator is initialized. The 48 MHz oscillator is not yet ready to run, therefore the application waits. To use the 48 MHz oscillator clock as the clock source, the system clock configuration is modified. After initializing the GPIO structure, the GPIO driver is also set up with a specific callback function for button events. Then interrupt masking is terminated. The I2C module is initialized with a registered callback function, and the I2C driver structure is initialized. I2C connection is now possible, and the driver is now ready for more interactions. In short, the code that is provided initializes and sets up the settings and components required for I2C connection with an MPU6050 sensor. In addition to handling interrupts, it configures the RF power supply, turns on the 48 MHz oscillator, and starts the GPIO and I2C drivers. These actions serve as the foundation for all subsequent I2C protocol communications with the MPU6050 sensor. The device address for the MPU6050 is 0x68. There are a few crucial registers that needs to be considered so as to get reliable data from the MPU6050 sensor. Important configuration options for the gyroscope are stored in the GYROCONFIG register, which is located at address 0x1B. Similar to that, key accelerometer settings may be found in the ACCELCONFIG register at address 0x1C. We put the value 0x00 to the register at location 0x6B to initialize the sensor and begin data collection. It's crucial to use certain register addresses when reading data from the MPU6050. Each of the

three axes (x, y, and z) contains 2 bytes of data, and the activity data is accessed starting at address 0x3B. The address 0x41address can be used to read two bytes of temperature information. Starting at location 0x43, gyroscope data can be accessed. Each axis (x, y, and z) comprises 2 bytes of data. The MPU6050 sensor's default device address is 0x68. To connect with the sensor, we set the device address to 0x68 in accordance with the I2C addressing rule. We use the GYROCONFIG register at address 0x1B to specify the gyroscope's entire scale range. The whole scale range is set to 500 degrees per second by putting the value 0x08 to this register. With this set up, the gyroscope can gauge angular velocity within the given range. For the accelerometer, we accessed the ACCELCONFIG register at address 0x1C. The whole scale range of this register is set to 8g by writing the value 0x10 to it. The accelerometer can now measure activity in the desired range thanks to this arrangement. We entered the value 0x00 into register MPU6050 at address 0x6B to initialize the sensor. This action starts the sensor, enabling it to supply measurements to the gyroscope and accelerometer.

We customized the MPU6050 sensor's settings to meet needs by executing these unique register configurations. We enable the gyroscope and accelerometer to precisely measure angular velocity and activity at specified ranges by setting appropriate full-scale ranges. The sensor is made operational and capable of providing useful data by the write initialization step 0x00 to register 0x6B. In conclusion, the instructions comprised configuring the MPU6050 sensor by choosing the appropriate registers and writing particular values to them. This enabled us to configure the sensor for precise readings and select the desired full-scale ranges for the accelerometer and gyroscope. After the data was obtained as 16-bit values, the higher value byte and the lower value byte were combined to obtain the full accelerometer, gyroscope and temperature data. This was achieved by shifting the higher value byte by 8 bits and performing an OR operation with the lower value byte. By doing this, the 16-bit data was reconstructed. In the case of gyroscope data, the offset obtained during the calibration process has been subtracted from the gyroscope data to obtain the final calibrated values. These steps were performed successfully

and resulted in accurate accelerometer, gyroscope and temperature data.

3.2 Microcontroller and Communication

3.2.1 RSL10 microcontroller

The RSL10 is a multi-protocol 2.4 GHz radio with extremely low power consumption and great flexibility that is intended for use in high-performance wearable and medical applications. There are various benefits to selecting the RSL10 CPU among available microprocessor options.

The low power level operation of RSL10 is made possible via BLE technology. This reduces the frequency of device recharges and extends the battery life of the wearable. Figure 3.6 shows block diagram of RSL10. BLE enables wearables to establish wireless connections with other smart devices. RSL10, which provides robust communication and data transfer capabilities, supports the BLE protocol. It is possible that the RSL10 CPU has the ports and interfaces needed to easily link it with other components. This simplifies the process of designing and developing wearables.

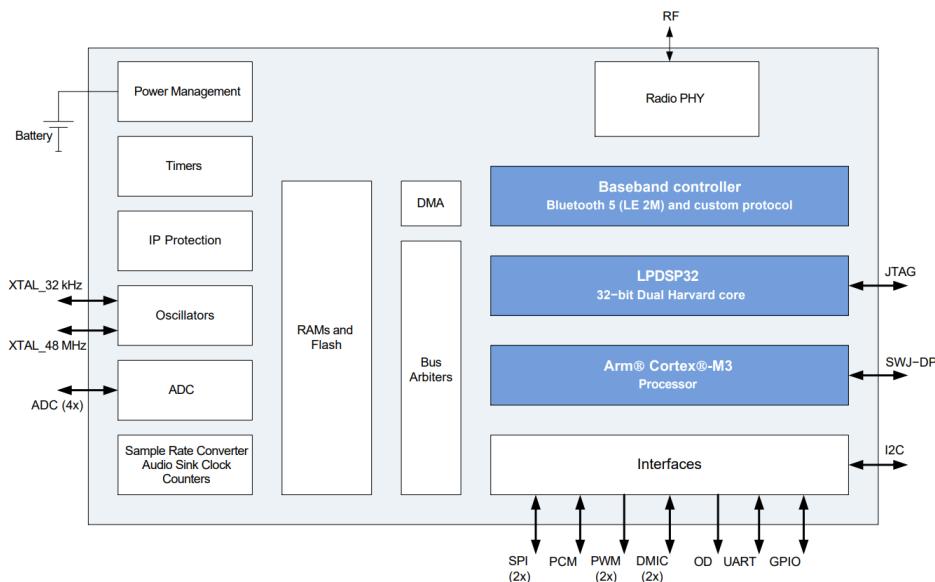


Figure 3.6 RSL10 Block Diagram (Anonim (RSL10), 2020)

3.2.2 Bluetooth low energy

In addition to using distinct channels from conventional Bluetooth, similar to Bluetooth, BLE functions within the industrial, technological, and clinical (ISM) spectrum at 2.4 GHz. BLE uses Gaussian frequency shift modulation to send data over 40 2-MHz channels, resulting in smoother transitions between data pulses. This reduces interference problems associated with frequency hopping compared to normal Bluetooth communications. BLE technology uses direct sequence spread spectrum to reduce interference from BLE signal broadcasts.

3.3 Schematic and PCB Design

3.3.1 Schematic Design

The design starts with the definition of requirements and a block diagram is prepared in figure 3.7 as a result of the search for the necessary electronic elements.

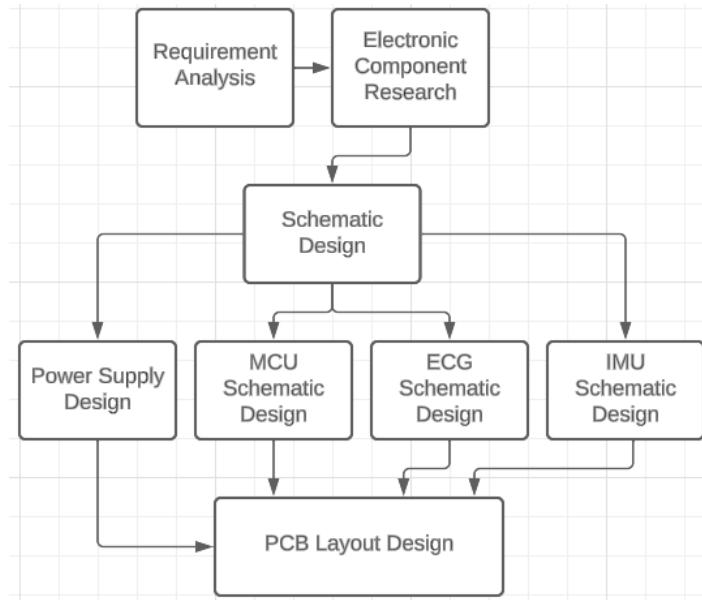


Figure 3.7 Block Design

Using the KiCad application's hierarchical circuit design capability, the electronic circuit schematic for the project circuit was created as it can be seen at figure 3.8. The

connections between the microprocessor and sensor, as well as the circuit's power line architecture, are shown in the generic electronic circuit schematic.

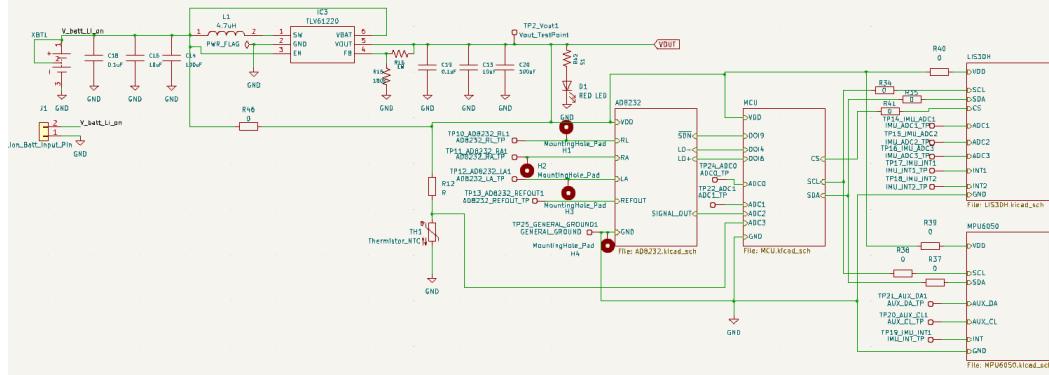


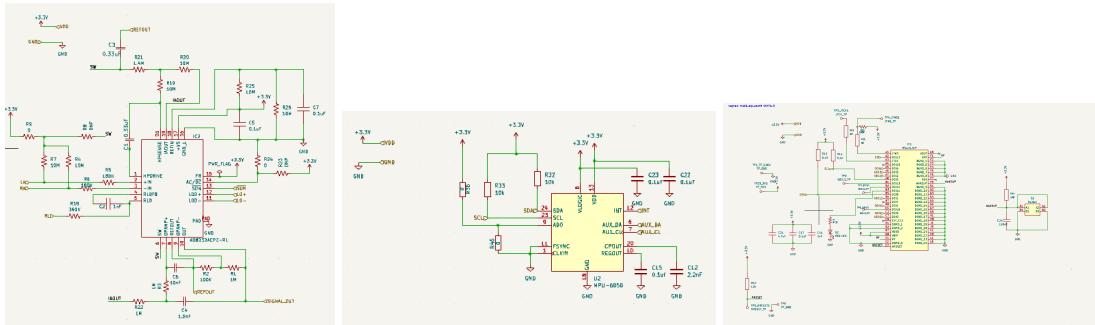
Figure 3.8 Schematic Design

The sample diagram used from the Texas Instruments TLV61220 IC datasheet, which we use as a voltage converter, is used as a guide when creating the power line of the electronic circuit. The TLV61220's input is equipped with capacitors to enhance the regulator's transient behavior. The TLV61220's output is equipped with capacitors to reduce the possibility of output voltage ripple. The TLV61220's output is equipped with capacitors to reduce the possibility of output voltage ripple.

The 3V CR2450 lithium battery was deemed suitable for use as a power source when building the power line. This battery is recommended because of its relatively modest size. The following equation illustrates how the voltage regulator resistors and the output voltage are related:

$$R_1 = R_2 * \left(\frac{V_{out}}{V_{FB}} - 1 \right) \quad (3.1)$$

The connections between the RSL10SIP microprocessor, the MPU6050 IMU sensor and the AD8232 ECG sensor are other features displayed in the primary circuit schematic in figure 3.9. The AD8232's input signals will come from ECG electrodes positioned at three distinct spots on the athlete's body, which will be connected to the PCB via conductive threads. The AD8232's structure includes an amplifier and high-pass filters, which enable the faint ECG signals from the athlete to be produced as a meaningful and analyzable signal. The RSL10 SIP's analog input ports are coupled to the relevant ECG signals that are produced.



(a) AD8232 Schematic Design

(b) MPU6050 Schematic Design

(c) RSL10 Schematic Design

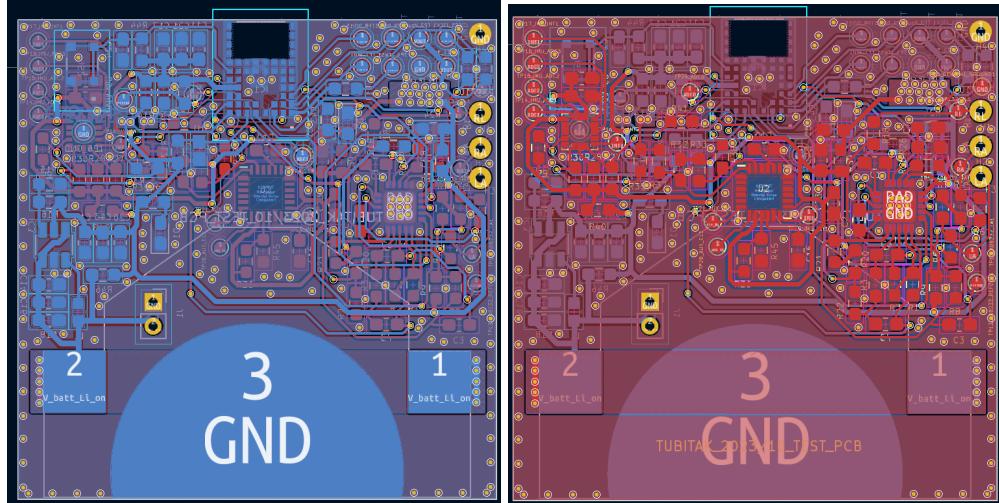
Figure 3.9 Circuit Design

The RSL10 SIP evaluation and development board user's manual was consulted when designing the microprocessor circuit. The RSL10 SIP evaluation and development board users handbook was consulted when designing the microprocessor circuit. Values for capacitors and resistors that match those in the comparison diagram are recommended. The processor can be put to sleep by connecting a button to the wake-up pin. Programming RSL10 SIP is best done by using the JTAG pin. This is recommended as the PCB we created is the initial sample and will be applied to experiments.

3.3.2 PCB Design

This section of the report will provide details on the PCB design that will be included in the smart t-shirt. KiCad 7.0 was used in its design. Every effort has been made to maintain the compact size of the PCB. The datasheets for the components utilized in the PCB were extensively consulted when designing the PCB layout for the thesis. Since PCB production is accomplished and testing is done on the generated PCB, there are many test locations left in the PCB layout.

When designing the arrangement, thin copper roads and vias have been used as much as possible because the overall voltage and current values of the electronic circuit are at low levels that will not damage the athlete. Since the AD8232 circuit has an excessive number of passive components, the layout sketching began with it. Subsequently, the connections between AD8232 and RSL10 SIP were established, followed by the connections between MPU6050 and the microcontroller. Lastly, the

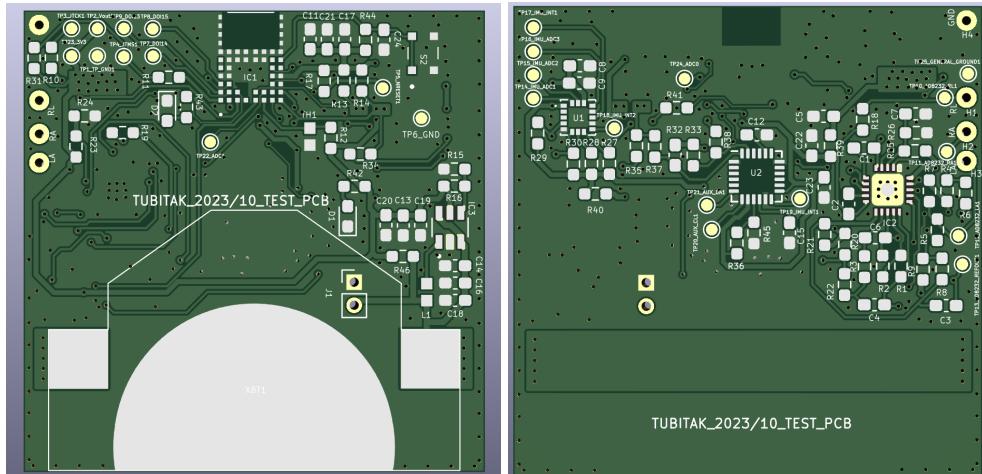


(a) Bottom view of PCB

(b) Top view of PCB

Figure 3.10 Bottom and Top view of PCB

layout was completed by adding the circuit's power line and JTAG programming pins.



(a) 3D Bottom view of PCB

(b) 3D Top view of PCB

Figure 3.11 3D Bottom and Top view of PCB

The two-layer PCB design method was used in the initial prototype design in figure 3.10. The two-layer PCB design is an important tool in reducing the size of the PCB to be produced. In a two-layer PCB design, some of the components can be located on one side of the PCB and others on the other in figure 3.11. In this way, the PCB size is reduced by roughly half, saving space, and because the size is small, cost reduction is also achieved during production. In a two-layer PCB design, the communication, power, and data paths drawn as well as the components can be

moved to the other side of the PCB. This is achieved by using small holes called vias in PCB design. Via's are useful in transferring the paths on both sides of the PCB, since they are conductive. When creating the PCB layout, the thickness of the data, power, and communication pathways must also be taken into account. The amount of current that will flow through the roads will determine how thick they should be. In other words, the path needs to be thicker, and the bigger the magnitude of the current that needs to pass. Since the thesis would operate at extremely low current levels, 0.3 mm was selected as the road thickness. The separation between the signal paths on a PCB is another important consideration. Because they carry electricity, paths that are close to one another can produce a magnetic influence on one another. It was also preferred that there be 0.3 mm of space between the highways to avoid this scenario.

3.4 Tshirt Design

The method used for T-shirt design is the placement of conductive threads on the T-shirt at specified locations. Fig. In order to connect the electrode probes to the areas shown in figure 3.4, the conductive thread connection was performed as figure 3.12. 3 electrode points were chosen because a minimum of 3 electrodes are needed to ensure measurement stabilisation.



Figure 3.12 Tshirt view inside

The designed PCB board is also shown in figure 3.12. The red and black parts under the PCB board are made to reinforce the connection between the PCB and the conductive threads. Since it is a prototype product, it is not a final product. When it is the final product, figure 3.13 will be integrated on the t-shirt.



Figure 3.13 Tshirt view outside

In order to avoid any problems during the measurement, this strengthening operation ensured that the measurements taken were more accurate and therefore prevented the conductive yarns from wearing out.

The actual photograph shown in figure 3.13, the wearer holds the PCB after putting it on, as specifically shown. The reason for this is to stabilise the X, Y and Z axis of the IMU and to analyse the possible movement according to the measurements taken. Because, although the axes of the sensor itself are determined, the motion values may change depending on the positioning.

3.5 Algorithm Design

3.5.1 General Algorithm and Bluetooth Low Energy Services

In this thesis, it is aimed to provide low energy bluetooth communication. The embedded system code was developed on the PCB designed for communication. With this algorithm developed, ECG and IMU data can be sent low-energy via Bluetooth. Basically, custom services are used for low-energy Bluetooth communication. Each custom service has its own Universally Unique Identified (UUIDs), and the read data can be distinguished according to these UUIDs as shown figure 3.14. Each bluetooth device must be a client or host. The PCB peripheral developed in this thesis and the cell phone that reads this peripheral are central.

The services used must be set as Generic Attribute (GATT) services. Because each UUID corresponds to a service, and these services have sub-communication service

IDs called characteristics. These two services are the characteristics used to receive

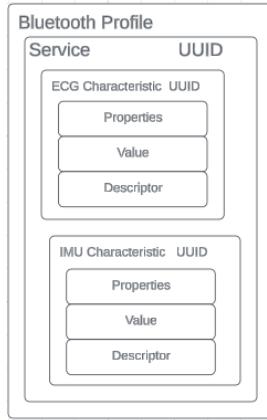


Figure 3.14 Bluetooth Low Energy Services

both ECG and IMU data. First, using the DIO2 pin with the RSL10SIP analog readout feature, voltage values between 0 and 3V are received via AD8232. In order to do this, a voltage divider was added on the PCB. Because RSL10SIP can read between 0 and 2V.

Instead of using the measured values directly, to obtain more accurate data, 10 data are collected and the average of them is sent over the ECG characteristic, and since the values received come in a specific format, it requires an extra calculation. The IEEE-11073 format is widely utilized in medical equipment. Basically, this format organizes double-type data as uint32, sending double during transmission consumes more power.

MPU6050 was used for IMU measurements and the data was sent to the IMU characteristic. Measurements were taken on the X, Y, and Z axes and this measurement was sent via bluetooth. I2C protocol was used to receive the data, and the I2C hardware connection figure 3.9b is shown. First of all, the designed device is powered, and with this power, the bluetooth device starts to broadcast. ECG measurements are taken while the Bluetooth device is broadcasting. Each of these measurements is taken every 100 ms, and 10 measurements are taken. An average of the 10 measured values was taken, and then the IMU measurement was started. In the IMU measurement, one measurement was taken every second. There is then a data

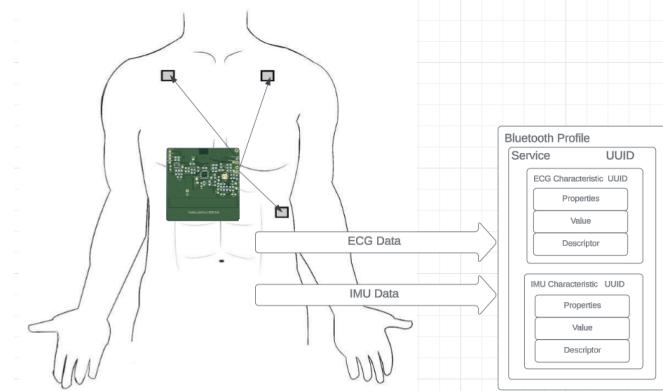


Figure 3.15 BLE and HW connection

processing part where the incoming values are checked so that these two measurements can be sent at once as a bluetooth packet. Then the bluetooth connection is made and all the measurement values are sent at once with the characteristics shown in figure 3.14.

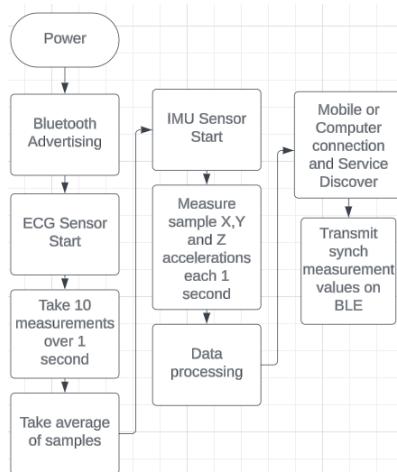


Figure 3.16 General Algorithm

Currently, a personal computer is used for offline data processing. In order for at least some of the algorithm to be coded directly onto the microprocessor in the future, it was designed to be as simple as feasible. The total amount of data produced may be reduced if the sensor system calculated the data directly and only the results could be kept. This method can greatly shorten the postprocessing times for long-term measurements. Real photos of PCB is shown at 3.17.



(a) Real Bottom view of PCB (b) Real Top view of PCB

Figure 3.17 Real Bottom and Top view of PCB

3.6 Machine Learning Methodology

In this study, we aim to classify human activity based on time series data captured from various physical activities, including jumping, running, descending stairs, and ascending stairs. Human activity classification plays a crucial role in fields such as sports biomechanics, healthcare monitoring, and activity recognition systems. To achieve accurate classification, we employ ten distinct classification methods: Decision Tree, Gradient Boosted Trees, Logistic Regression, Markov Models, Naive Bayes, Random Forest, and Support Vector Machine (SVM). Each technique possesses distinct strengths and qualities, which span from simplicity and interpretability to intricate ensemble learning and the ability to do non-linear mapping. Through the process of comparing and evaluating different strategies, our goal is to determine the most efficient approach for precisely categorizing human activity patterns based on time series data.

3.6.1 Decision Tree

Decision trees are a popular supervised learning method used for classification and regression tasks. A decision tree partitions the feature space into regions, assigning a class label or a regression value to each region. The tree structure consists of nodes representing feature tests and edges representing the outcome of those tests. Decision trees are built recursively by selecting the best feature and split point at each node based on criteria such as Gini impurity or information gain.

A decision tree consists of nodes and edges. Each node represents a feature test, and each edge represents the outcome of that test, leading to subsequent nodes or leaf nodes. The structure of the decision tree can be represented mathematically using a tree data structure, where each node contains information about the feature to split on and the threshold (for continuous features) or categories (for categorical features). Decision trees are built by recursively selecting the best feature and split point at each node to maximize the information gain or minimize impurity. Common splitting criteria include Gini impurity and information gain (or entropy). Let $I(D)$ denote the impurity measure of dataset D , and let

$$I_{\text{split}}(D, \text{feature}, \text{split_point})$$

denote the impurity after splitting D based on the selected feature and split point. The information gain ΔI can be calculated as:

$$\Delta I = I(D) - I_{\text{split}}(D, \text{feature}, \text{split_point}) \quad (3.2)$$

The feature and split point that maximize the information gain are chosen for the split at each node.

Gini impurity measures the probability of incorrectly classifying a randomly chosen instance in the dataset. For a dataset D with K classes, the Gini impurity $I_G(D)$ can be calculated as:

$$I_G(D) = 1 - \sum_{i=1}^K p_i^2 \quad (3.3)$$

where p_i is the probability of randomly choosing an instance of class i in dataset D . Information gain measures the reduction in entropy (or uncertainty) achieved by splitting the dataset based on a particular feature. For a dataset D , the entropy $H(D)$ is calculated as:

$$H(D) = - \sum_{i=1}^K p_i \log_2(p_i) \quad (3.4)$$

where p_i is the proportion of instances of class i in dataset D . The information gain ΔH is then calculated as the difference in entropy before and after the split.

By recursively selecting the best feature and split point based on the chosen criteria, decision trees partition the feature space into regions that are increasingly homogeneous with respect to the target variable. This process continues until certain stopping criteria are met, such as reaching a maximum tree depth or minimum number of instances per node. Decision trees provide interpretable models for classification and regression tasks and can handle both numerical and categorical features.

3.6.2 Gradient Boosted Trees

Gradient Boosted Trees (GBT) is an ensemble learning method that builds a predictive model by combining the outputs of multiple decision trees. GBT works by iteratively training weak learners (typically decision trees) to correct the errors of the previous models. It minimizes a loss function by adding new trees that predict the residuals of the current model.

GBT aims to minimize a predefined loss function $L(y, F(x))$, where y represents the true labels and $F(x)$ represents the ensemble's predictions. The objective function to be minimized can be expressed as:

$$\text{Obj} = \sum_{i=1}^n L(y_i, F(x_i)) \quad (3.5)$$

GBT employs gradient descent optimization to iteratively improve the ensemble's predictions. At each iteration k , a new weak learner $h_k(x)$ is trained to predict the negative gradient of the loss function with respect to the current ensemble's predictions. The negative gradient for each instance i can be calculated as:

$$r_i^{(k)} = -\frac{\partial L(y_i, F(x_i))}{\partial F(x_i)} \quad (3.6)$$

The new weak learner $h_k(x)$ is trained to fit the residuals $r_i^{(k)}$ of the current ensemble's predictions. The ensemble's predictions are updated by adding the predictions of the new weak learner, scaled by a learning rate η :

$$F^{(k+1)}(x) = F^{(k)}(x) + \eta \cdot h_k(x) \quad (3.7)$$

To prevent overfitting, GBT typically employs regularization techniques such as shrinkage (learning rate) and tree structure constraints (maximum depth, minimum samples per leaf). The learning rate η controls the step size of each iteration, while tree structure constraints limit the complexity of individual trees.

The final prediction of the GBT ensemble is the sum of the predictions of all weak learners:

$$F(x) = \sum_{k=1}^K \eta \cdot h_k(x) \quad (3.8)$$

GBT algorithms such as XGBoost (Extreme Gradient Boosting) and LightGBM (Light Gradient Boosting Machine) implement variations of the GBT algorithm with optimizations for computational efficiency and model performance. These algorithms are widely used in various classification tasks due to their ability to handle complex nonlinear relationships and large-scale datasets while achieving high accuracy and flexibility.

3.6.3 Logistic Regression

Logistic regression is a linear classification model used for binary classification tasks. It models the probability of the occurrence of a binary outcome based on one or more predictor variables. Logistic regression applies the logistic function (sigmoid function) to the linear combination of predictor variables to transform the output into a probability between 0 and 1. The model parameters are estimated using maximum

likelihood estimation, and regularization techniques like L1 and L2 regularization can be applied to prevent overfitting.

In logistic regression, the relationship between the predictor variables x and the binary outcome y is modeled using a logistic function. Given n predictor variables, the logistic regression model can be represented as:

$$P(y = 1|x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n)}} \quad (3.9)$$

where $P(y = 1|x)$ represents the probability of the positive class given the input x , and $\beta_0, \beta_1, \dots, \beta_n$ are the model parameters (coefficients) to be estimated.

The logistic function, also known as the sigmoid function, maps the linear combination of predictor variables to a probability between 0 and 1. The logistic function is defined as:

$$g(z) = \frac{1}{1 + e^{-z}} \quad (3.10)$$

where z is the linear combination of predictor variables and model parameters:

$$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_n x_n \quad (3.11)$$

The parameters $\beta_0, \beta_1, \dots, \beta_n$ of the logistic regression model are estimated using maximum likelihood estimation (MLE). The likelihood function $L(\beta)$ is defined as the probability of observing the given outcomes (positive and negative classes) given the input data and the model parameters. Maximizing the likelihood function is equivalent to minimizing the negative log-likelihood function:

$$\text{minimize} \left\{ - \sum_{i=1}^N (y_i \log(P(y = 1|x_i)) + (1 - y_i) \log(1 - P(y = 1|x_i))) \right\} \quad (3.12)$$

To prevent overfitting, regularization techniques such as L1 and L2 regularization can

be applied to the logistic regression model. L1 regularization adds a penalty term proportional to the absolute values of the coefficients, while L2 regularization adds a penalty term proportional to the square of the coefficients.

Logistic regression is widely used for binary classification tasks due to its simplicity, interpretability, and ability to model the probability of class membership. By estimating the parameters using maximum likelihood estimation and applying regularization techniques, logistic regression can effectively handle overfitting and improve generalization performance.

3.6.4 *Hidden Markov Models*

Hidden Markov Models (HMMs), are used for sequential data analysis, including classification tasks such as speech recognition and part-of-speech tagging. In a Markov model, the probability of transitioning between states depends only on the current state and not on the sequence of previous states. The model parameters, including transition probabilities and emission probabilities, are typically estimated using algorithms like the Baum-Welch algorithm (an expectation-maximization algorithm).

In a Markov model, the probability of transitioning from one state to another depends only on the current state and not on the sequence of previous states. This property is known as the Markov property. Let π_i denote the probability of being in state i at time t , and let A be the state transition probability matrix, where A_{ij} represents the probability of transitioning from state i to state j . The state transition probability π_{t+1} at time $t + 1$ can be calculated as:

$$\pi_{t+1} = \pi_t \cdot A \quad (3.13)$$

In Hidden Markov Models (HMMs), each state emits an observation according to a probability distribution. Let B be the emission probability matrix, where B_{ij} represents the probability of emitting observation j given that the model is in state i . The emission

probability b_{ij} of emitting observation j from state i can be calculated as:

$$b_{ij} = P(\text{observation } j|\text{state } i) \quad (3.14)$$

The initial state probability distribution π_0 represents the probability of starting in each state. Let π_0 be a vector where π_{0i} represents the probability of starting in state i . The Baum-Welch algorithm, also known as the forward-backward algorithm, is an expectation-maximization (EM) algorithm used to estimate the parameters of Hidden Markov Models (HMMs) from observed data when the underlying states are not directly observable. The algorithm iteratively updates the parameters (transition probabilities, emission probabilities, and initial state probabilities) of the HMM to maximize the likelihood of the observed data.

Given a sequence of observations $O = (o_1, o_2, \dots, o_T)$, the probability of observing this sequence in an HMM can be calculated using the forward algorithm. Let $\alpha_t(i)$ denote the probability of being in state i at time t and observing the first t observations. The probability of observing the entire sequence O can be calculated as:

$$P(O|\lambda) = \sum_{i=1}^N \alpha_T(i) \quad (3.15)$$

where N is the number of states in the HMM and λ represents the model parameters (transition probabilities, emission probabilities, and initial state probabilities).

Markov models, particularly Hidden Markov Models (HMMs), are powerful tools for sequential data analysis due to their ability to model temporal dependencies and uncertainty in observed data. By estimating parameters using algorithms such as the Baum-Welch algorithm, HMMs can effectively capture the underlying structure of sequential data and be applied to various classification tasks such as speech recognition, part-of-speech tagging, and bioinformatics.

3.6.5 Naive Bayes

Naive Bayes is a probabilistic classification method based on Bayes' theorem with the "naive" assumption of independence between features. Despite its simplicity, Naive Bayes often performs well in text classification and other domains with high-dimensional data. It calculates the posterior probability of each class given the input features and selects the class with the highest probability. Different variants of Naive Bayes, such as Gaussian Naive Bayes and Multinomial Naive Bayes, are used depending on the distribution of features.

Naive Bayes classification is based on Bayes' theorem, which provides a way to calculate posterior probabilities given observed data. Bayes' theorem is expressed as:

$$P(y|x_1, x_2, \dots, x_n) = \frac{P(y) \cdot P(x_1, x_2, \dots, x_n|y)}{P(x_1, x_2, \dots, x_n)} \quad (3.16)$$

where y represents the class label, x_1, x_2, \dots, x_n represent the input features, $P(y)$ is the prior probability of class y , $P(x_1, x_2, \dots, x_n|y)$ is the likelihood of observing the features given class y , and $P(x_1, x_2, \dots, x_n)$ is the marginal probability of observing the features.

Naive Bayes makes the "naive" assumption that the features are conditionally independent given the class label y . Mathematically, this assumption can be expressed as:

$$P(x_1, x_2, \dots, x_n|y) = P(x_1|y) \cdot P(x_2|y) \cdot \dots \cdot P(x_n|y) \quad (3.17)$$

This assumption simplifies the calculation of the likelihood term.

Different variants of Naive Bayes, such as Gaussian Naive Bayes and Multinomial Naive Bayes, are used depending on the distribution of features: Gaussian Naive Bayes assumes that the continuous features follow a Gaussian (normal) distribution. The likelihood $P(x_i|y)$ is modeled using the probability density function of the Gaussian distribution. Multinomial Naive Bayes is Suitable for features that represent

counts or frequencies, such as word counts in text classification. It assumes that the features follow a multinomial distribution. The likelihood $P(x_i|y)$ is modeled using the multinomial distribution.

Naive Bayes classification selects the class with the highest posterior probability given the input features. This decision rule is known as the Maximum A Posteriori (MAP) decision rule:

$$\hat{y} = \operatorname{argmax}_{y \in \{1, 2, \dots, C\}} P(y|x_1, x_2, \dots, x_n) \quad (3.18)$$

Naive Bayes classification, despite its "naive" assumption of feature independence, often performs well in practice, especially in text classification and other domains with high-dimensional data. By estimating the prior probabilities and likelihoods from the training data and applying the MAP decision rule, Naive Bayes provides a simple yet effective probabilistic framework for classification tasks.

3.6.6 Random Forest

Random Forest is an ensemble learning method that constructs a multitude of decision trees during training and outputs the mode of the classes (classification) or the mean prediction (regression) of the individual trees. Each tree in the forest is built using a random subset of the training data and a random subset of features at each split point. Random Forest reduces overfitting and improves generalization by averaging predictions from multiple trees, thereby achieving higher accuracy compared to individual decision trees.

Random Forest is an ensemble of decision trees. A decision tree partitions the feature space into regions and assigns a class label (for classification) or a prediction value (for regression) to each region. The decision tree is constructed recursively by selecting the best feature and split point at each node based on certain criteria such as Gini impurity or information gain.

Random Forest builds each decision tree using a bootstrapped sample of the training data. Given a dataset of size N , a random sample of size N' (where $N' < N$) is drawn with replacement. This sampling technique, known as bootstrapping, introduces randomness and diversity into the training process.

At each split point in a decision tree, only a random subset of features is considered for splitting. This further introduces randomness and decorrelates the individual trees in the forest. Let m denote the total number of features, and m' denote the size of the random feature subset (typically \sqrt{m} or $\frac{m}{3}$).

In classification tasks, the mode (most frequent class) of the class predictions from individual trees is taken as the final prediction of the Random Forest. In regression tasks, the mean prediction from all trees is computed as the final prediction. This aggregation of predictions helps reduce variance and improve generalization performance.

Random Forest reduces overfitting by averaging predictions from multiple trees trained on different subsets of the data and features. This ensemble approach mitigates the risk of individual trees memorizing noise in the training data, leading to better generalization to unseen data.

Random Forest estimates the generalization error using out-of-bag (OOB) samples, which are the data points not included in the bootstrapped sample used to train each tree. The OOB error is computed as the error rate of each data point based on the predictions from trees that did not use that data point in training. This provides an unbiased estimate of the model's performance without the need for a separate validation set.

Random Forest is a powerful ensemble learning method that combines the strength of decision trees with randomness and diversity to achieve high accuracy and robustness in classification and regression tasks. By building multiple trees using

bootstrapped samples and random feature subsets, Random Forest effectively reduces overfitting and improves generalization performance.

3.6.7 Support Vector Machine

Support Vector Machine (SVM) is a powerful supervised learning method used for classification and regression tasks. SVM finds the optimal hyperplane that separates the classes in the feature space with the maximum margin, where the margin is the distance between the hyperplane and the closest data points (support vectors). SVM can handle linearly separable as well as non-linearly separable data by using kernel functions to map the input features into higher-dimensional space. The model parameters are determined by solving a convex optimization problem.

In the case of linearly separable data, SVM aims to find the optimal hyperplane that separates the classes with the maximum margin. Mathematically, the decision boundary (hyperplane) is represented as:

$$\mathbf{w} \cdot \mathbf{x} + b = 0 \quad (3.19)$$

where \mathbf{w} is the weight vector perpendicular to the hyperplane, \mathbf{x} is the input feature vector, and b is the bias term.

The margin is the distance between the hyperplane and the closest data points (support vectors). SVM maximizes the margin to improve generalization performance and robustness. Mathematically, the margin M is calculated as the distance between two parallel hyperplanes:

$$M = \frac{2}{\|\mathbf{w}\|} \quad (3.20)$$

SVM aims to minimize the norm of the weight vector $\|\mathbf{w}\|$ subject to the constraint that all data points are correctly classified and lie outside the margin. This leads to the following optimization problem:

$$\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 \quad (3.21)$$

subject to $y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1$ for all i , where y_i is the class label of data point \mathbf{x}_i .

SVM can handle non-linearly separable data by mapping the input features into a higher-dimensional space using kernel functions. Kernel functions compute the inner product of transformed feature vectors efficiently without explicitly computing the transformation. Common kernel functions include:

- Linear kernel: $K(\mathbf{x}, \mathbf{x}') = \mathbf{x} \cdot \mathbf{x}'$
- Polynomial kernel: $K(\mathbf{x}, \mathbf{x}') = (\gamma \mathbf{x} \cdot \mathbf{x}' + r)^d$
- Gaussian (RBF) kernel: $K(\mathbf{x}, \mathbf{x}') = \exp(-\gamma \|\mathbf{x} - \mathbf{x}'\|^2)$

In cases where the data is not perfectly separable, SVM allows for soft margins by introducing slack variables ξ_i to tolerate misclassifications. The optimization problem becomes:

$$\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i \quad (3.22)$$

subject to $y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1 - \xi_i$ and $\xi_i \geq 0$ for all i , where C is the regularization parameter.

Support Vector Machines (SVMs) are powerful models for both linear and non-linear classification tasks. By finding the optimal hyperplane with the maximum margin, SVMs achieve good generalization performance and can handle high-dimensional data efficiently. The use of kernel functions further extends SVM's applicability to non-linearly separable data, making it a versatile tool in machine learning.

3.6.8 Performance Metrics

This study utilizes many classification measures to evaluate the classification performance of different machine learning methods on each time series. For each of the classes, classification results are measured respect to following metrics:

The area under the ROC curve (AUC) measures the ability of a classifier to

distinguish between classes. For each class c , the AUC can be calculated by integrating the ROC curve ROC_c with respect to the false positive rate (FPR) FPR_c and true positive rate (TPR) TPR_c as:

$$AUC_c = \int_0^1 TPR_c(FPR_c^{-1}(t)) dt \quad (3.23)$$

The mean cross entropy for each class c measures the average information gain or surprise associated with the classification predictions. Given the true class probabilities y_{ij} and the predicted class probabilities \hat{y}_{ij} for instance i and class j , the cross entropy CE_{ij} is calculated as:

$$CE_{ij} = -y_{ij} \log(\hat{y}_{ij}) \quad (3.24)$$

The mean cross entropy for class c is then calculated as the average of cross entropies over all instances i belonging to class c .

The rejection rate for each class c represents the proportion of instances misclassified as other classes. It is calculated as:

$$\text{Rejection Rate}_c = \frac{\sum_{i=1}^N \mathbb{1}(y_i \neq c \wedge \hat{y}_i = c)}{\sum_{i=1}^N \mathbb{1}(y_i = c)} \quad (3.25)$$

where N is the total number of instances, y_i is the true class label of instance i , \hat{y}_i is the predicted class label of instance i , and $\mathbb{1}(\cdot)$ is the indicator function.

The F1 score for each class c is the harmonic mean of precision P_c and recall R_c , and it measures the balance between precision and recall:

$$F1_c = \frac{2 \cdot P_c \cdot R_c}{P_c + R_c} \quad (3.26)$$

The false discovery rate for each class c represents the proportion of instances misclassified as class c among all instances predicted as class c . It is calculated as:

$$FDR_c = \frac{\sum_{i=1}^N \mathbb{1}(y_i \neq c \wedge \hat{y}_i = c)}{\sum_{i=1}^N \mathbb{1}(\hat{y}_i = c)} \quad (3.27)$$

The false negative rate for each class c represents the proportion of instances belonging to class c that were incorrectly classified as other classes. It is calculated as:

$$\text{FNR}_c = \frac{\sum_{i=1}^N \mathbb{1}(y_i = c \wedge \hat{y}_i \neq c)}{\sum_{i=1}^N \mathbb{1}(y_i = c)} \quad (3.28)$$

The false positive rate for each class c represents the proportion of instances not belonging to class c that were incorrectly classified as class c . It is calculated as:

$$\text{FPR}_c = \frac{\sum_{i=1}^N \mathbb{1}(y_i \neq c \wedge \hat{y}_i = c)}{\sum_{i=1}^N \mathbb{1}(y_i \neq c)} \quad (3.29)$$

The Matthews correlation coefficient for each class c measures the correlation between the true and predicted class labels, accounting for imbalanced class distributions. It is calculated as:

$$\text{MCC}_c = \frac{TP_c \times TN_c - FP_c \times FN_c}{\sqrt{(TP_c + FP_c)(TP_c + FN_c)(TN_c + FP_c)(TN_c + FN_c)}} \quad (3.30)$$

where TP_c , TN_c , FP_c , and FN_c represent the true positive, true negative, false positive, and false negative counts for class c , respectively.

The negative predictive value for each class c represents the proportion of instances not belonging to class c among instances not predicted as class c . It is calculated as:

$$\text{NPV}_c = \frac{TN_c}{TN_c + FN_c} \quad (3.31)$$

Precision for each class c represents the proportion of instances correctly classified as class c among instances predicted as class c . It is calculated as:

$$P_c = \frac{TP_c}{TP_c + FP_c} \quad (3.32)$$

Recall rate for each class c represents the proportion of instances correctly classified as class c among instances belonging to class c . It is calculated as:

$$R_c = \frac{TP_c}{TP_c + FN_c} \quad (3.33)$$

Specificity for each class c represents the proportion of instances not belonging to class c among instances not belonging to class c . It is calculated as:

$$\text{Specificity}_c = \frac{TN_c}{TN_c + FP_c} \quad (3.34)$$

These extended mathematical formulations offer a more in-depth comprehension of each classification measure, as well as the manner in which they quantify various aspects of classifier performance for each class.

CHAPTER FOUR

APPLICATIONS AND RESULTS

A review of the capability of this RSL10 platform for the creation of digital filters revealed that it has sufficient runtime and memory. The AD8232 may serve as a normal ECG monitor for basic ECG monitoring and analysis, and is statistically significant. Since the voltage value changes by mV, noise signals can affect the accuracy of the measurement. Given that the voltage obtained from the ECG is within the millivolt range, the RSL10 uses a Kalman filter to de-noise the signal. The figure of the measurement results for the ECG signal is figure 4.1. You can use the analog command on an RSL10 board to convert the input voltage of a specific channel to a 10-bit integer to calculate the voltage range between ground and Vcc (5.5V). The AD8232's maximum working voltage is less than 5.5V, and hence we are using 3.3V.

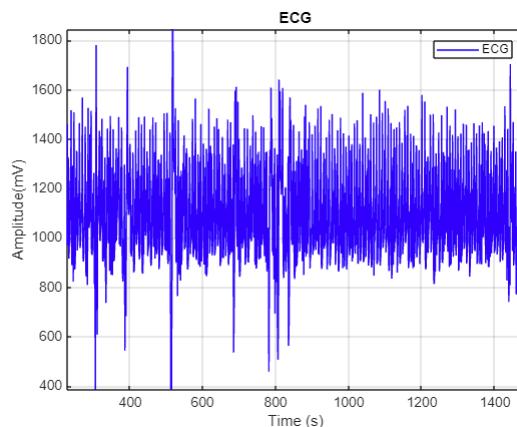


Figure 4.1 ECG signal

The majority of the acquired graphs did not even at first display ECG-like curves. The results were calculated using the maximum or zero value of the sensor. After experimentation with different body positions, it was discovered that this only occurred when feet were on the ground. When high-input impedance amplifiers are employed, it is straightforward to manufacture electrodes of various sizes using flexible boards or fabrics. The ground approach significantly reduces the intensity of the noise. However, the sitting and standing positions with the feet on the ground were discarded. For the next tests, the participant was lying down or having their feet raised. Understanding more about sensor operation, best electrode locations, and

optimal values for the variables mentioned above led to improvements in the clarity and quietness of the images, as seen in figure 3.15.

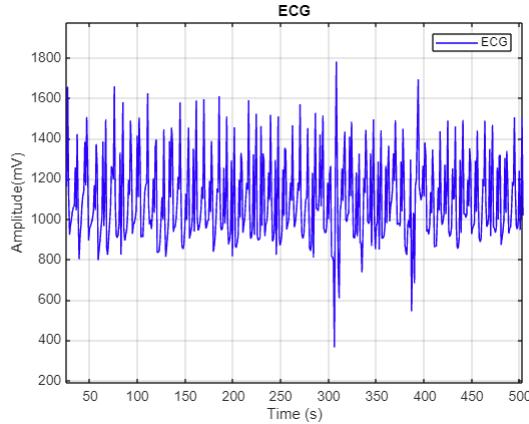


Figure 4.2 ECG signal(zoom)

Figures shows data on activity readings acquired from the right arm side body location. This figure can be used to determine the player's activity. There are modifications on the Z axis that can be used to comprehend activity. The default settings for the accelerometer are +/- 2g and 250% every second, respectively. The settings in question are the most delicate. Before being used for a project, the IMU must be calibrated.



Figure 4.3 IMU sensor position

IMU sensor at point 1 of figure 4.3 gathers measurements as the player moves and changes positions. These values, as indicated in the figures, are used to determine the required activity values. ECG and activity measurements can be used to show how movement and heart rate are related. There is a relationship between activity and

ECG. A person traveling down the Z-axis has an increase in heart rate as they advance. His pulse will not increase, and his heart rate will remain steady if he remains motionless.

The ECG and activity measurements can be used to show how movement and heart rate are related. The figures show the relationship between activity and the ECG. When a person moves along the z axis, their heart rate increases with them. His pulse will not increase and his heart rate will stay regular if he remains motionless. The QRS complex is often the place where the repeat pattern of the ECG is highest. Consequently, the cardiac rate, measured in BPM, is determined by the time gap between two consecutive QRS complexes.

Movement can also be determined from activity data by employing the signal's frequency. The activity data along the z-axis that came from the QRS complex can be used to compare this condition to a steady and quick walking situation. The frequency of the peak value of the z-activity can be used to estimate movement if it counts.

4.1 Pattern measurement

In the next section, the measurements of the four activities are illustrated figures and bpm measurements.

Running: The user is running;

Jumping: The user is jumping,

Stairs-Up: The user is going up a staircase and

Stairs-Down: The user is going down a staircase.

Data on acceleration gathered between the activity start and stop times were labeled with the activity's name. The ECG sensor measures several body location values during data collection. This query emerges from the fact that the acceleration measurements for the same physical activity at various points on the user's body differ significantly. The sampling rate of each data is almost 80ms depending on the

Bluetooth data timestamp. The data collection of the ECG and IMU sensors is noisy due to wireless data sharing and activities.

4.1.1 Running Pattern measurement

The running pattern X, Y and the Z axis are showed at figure 4.4a , 4.4b , 4.4c. Depending on the location of the sensor, the movement is on the Y axis, and the frequency of the Y axis can be used to determine the frequency of activity.

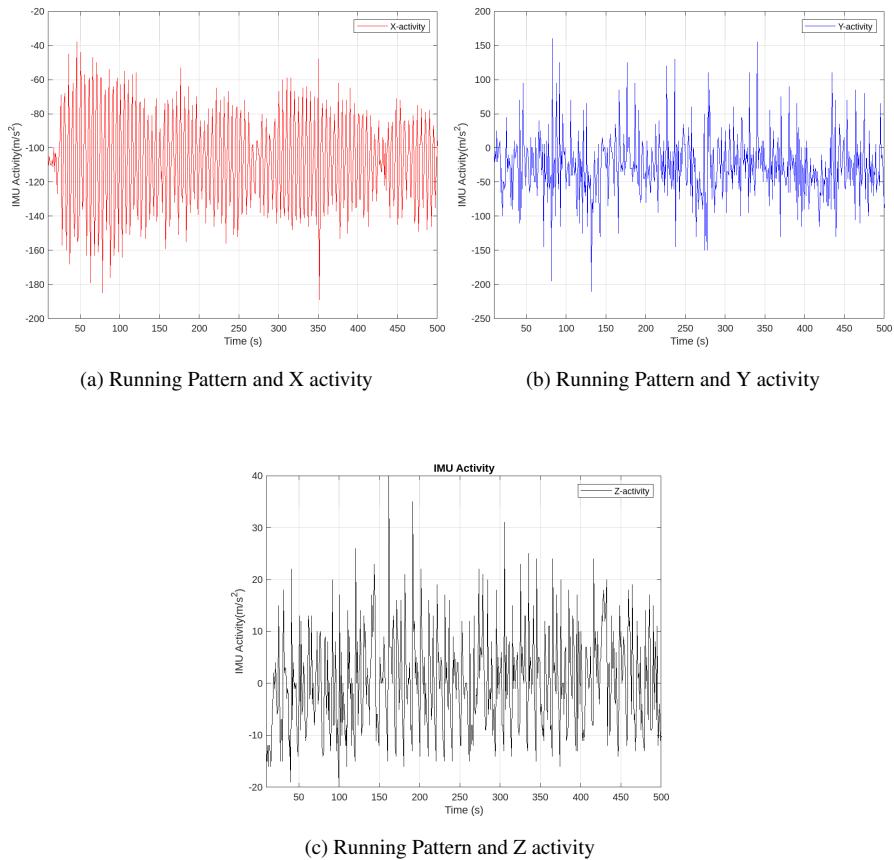


Figure 4.4 Running Pattern and Activity Diagram

The frequency of Y activity varies depending on the movement variations and the conditions of the running pattern. This modification shows the athlete's activity graph while sprinting or jumping.

According to sensor positioning, X, Y and Z activities and ECG values during running activity can be seen in figure 4.5. As can be seen in the figures, the data

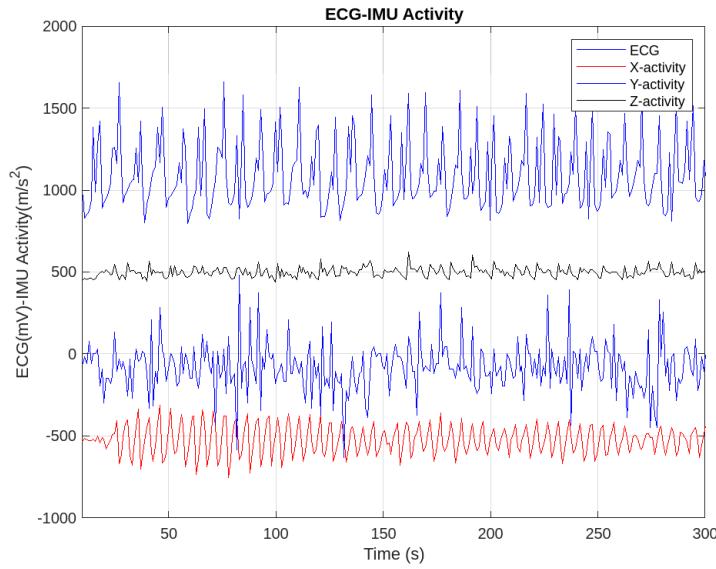


Figure 4.5 Running Pattern ECG and Activity Diagram

obtained with the conductive thread almost resembles the normal ECG signal. ECG data obtained during running are shown in figure 4.6 and figure 4.2. The heart rate

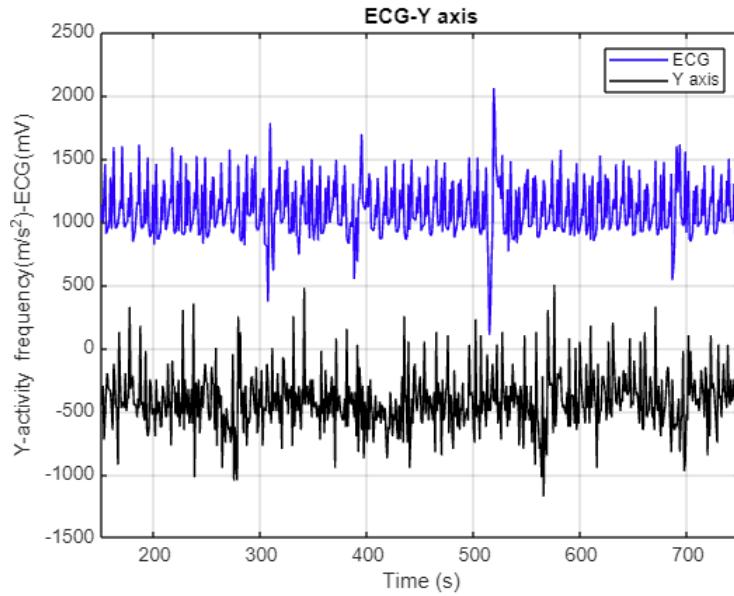


Figure 4.6 ECG Running Pattern and Y axis

when running is explained in figure 4.6. R signal from figure 3.3 can be utilized to compute heart rate. Every peak value of the ECG signal is considered based on this R signal. Finding the peak count in 10 seconds and increasing it by 6 yields results in 1 minute. An approximate beats-per-minute (BPM) will be provided by this. In general, running action yields a heart rate of 84 BPM. Y activity is the measure of how the running action alters the motion.

4.1.2 Jumping Pattern measurement

One of the most important ways to observe movement variability and the sensor comparison section is to jump. Jumping activities cause some shaking of the sensors, which makes obtaining measurements a little difficult. For this test, 70 jumps were performed. The data collected as a result of these 70 jumps are shown in figure 4.7a , 4.7b , 4.7c. Due to the fact that figure 4.2 and figure 4.9 exhibit distinct variations in acceleration and ECG readings between running and jumping.

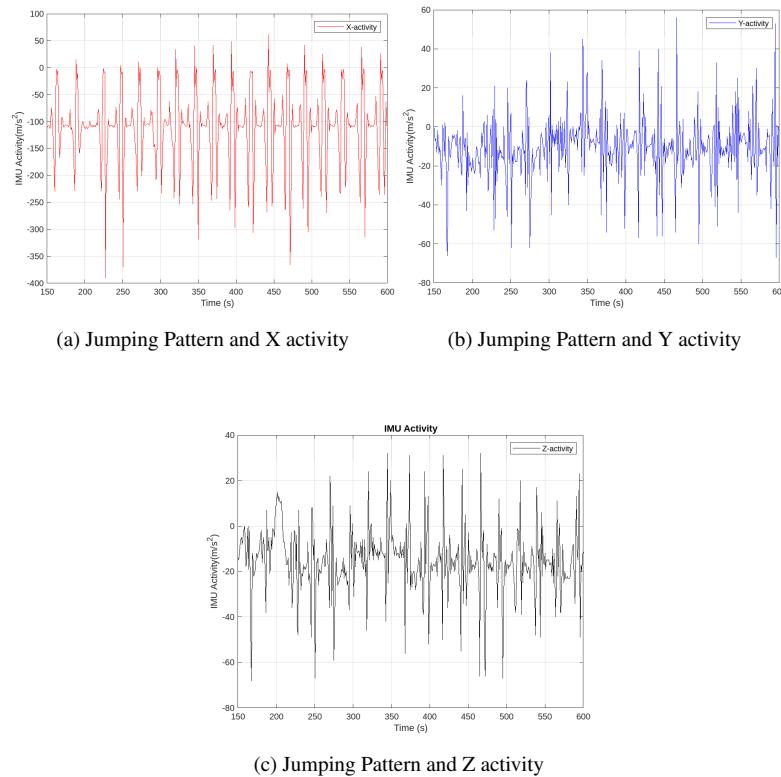


Figure 4.7 Jumping Pattern and Activity Diagram

The large differences in vertical movement during the activity under analysis are indicated by the significant changes in acceleration recorded on the Z-axis. This may be associated with periods of abrupt elevation or decrease, during which the body undergoes significant variations in speed when moving either in opposition to or in tandem with gravity. Determining the activity's intensity and biomechanics requires an understanding of these acceleration variations. Periods with lower acceleration could represent times of relative rest or decreased movement, while peaks or spikes in acceleration could represent times of higher muscular force output or deceleration.

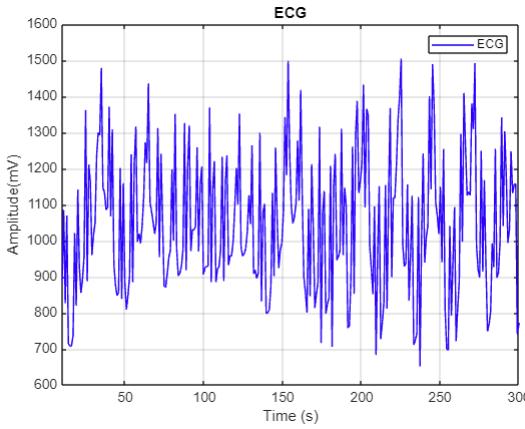


Figure 4.8 Jumping Pattern ECG

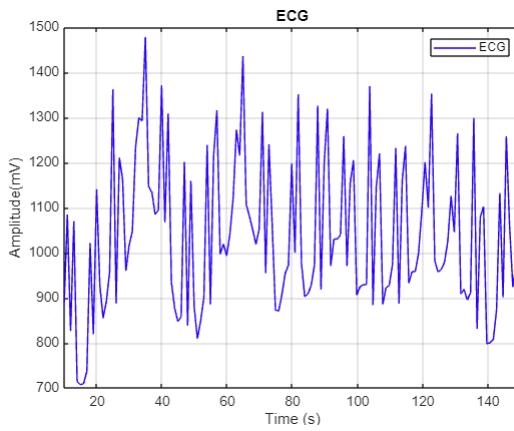


Figure 4.9 Jumping Pattern ECG (Zoom)

When the measurement results are analyzed in figure 4.10, it can be seen that the approximate heart rate varies between 78 and 84 BPM. This is the best possible result. Due to shaking, the heart rate fluctuates by 84 beats per minute. The graph bounces at the Z-activity and is the point at which jumping occurs.

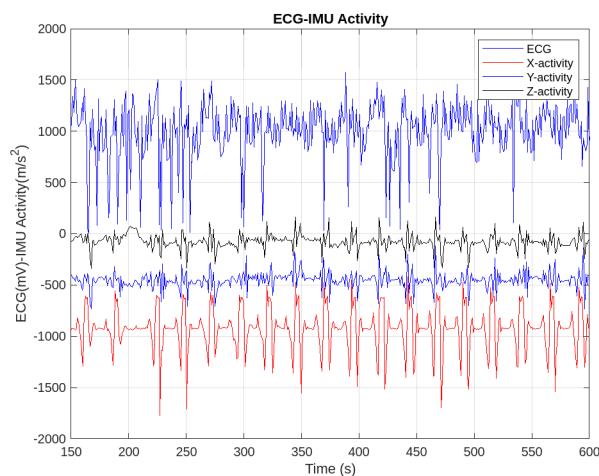


Figure 4.10 Jumping Pattern ECG and X,Y and Z axis

4.1.3 Stairs Pattern measurement

Descending and ascending the stairs were analysed under 2 different headings. X, Y and Z data from these titles and ECG data were drawn and interpreted under their own headings. When the graphs are analyzed, it can be seen that there are roughly differences between them. Consequently, there are also visible changes in heart rate. Figures show that heart rate varies with the ascending and falling stairs conditions and that the frequency of the activity diagrams for X and Y activities is increasing.

4.1.3.1 Stairs Down Pattern

Acceleration movements on the X, Y and Z axes are shown in figure 4.11a , 4.11b and 4.11c. When we look at these graphs, a change is observed in all 3 axes during descent from the stairs.

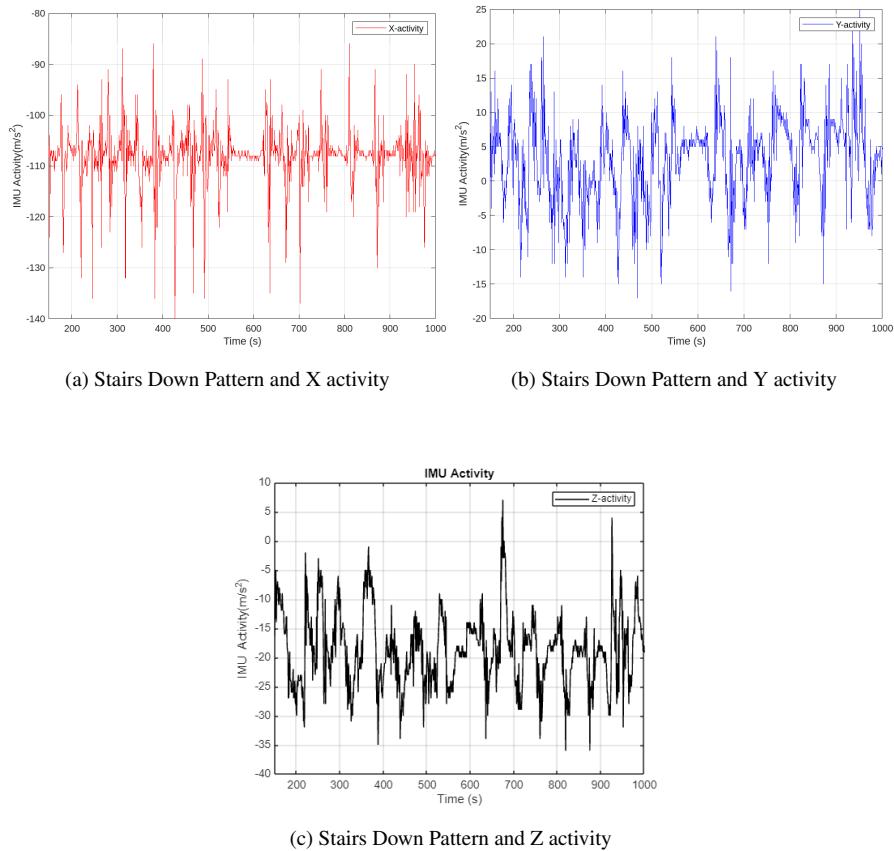


Figure 4.11 Stairs Down Pattern and Activity Diagram

ECG data corresponding to the acceleration movement were also observed for

comparison purposes in figure 4.12.

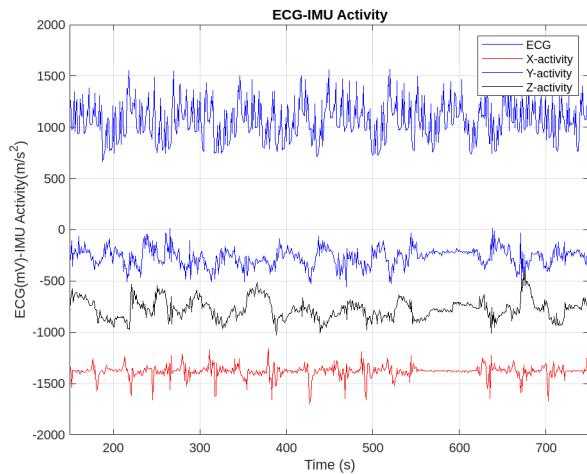


Figure 4.12 Stairs Down ECG Pattern and X,Y and Z axis

Heartbeat data of about 78 BPM is seen when looking at the entire 10-second image of the collected ECG data, as shown in figure 4.13 and 4.14. This BPM number ranges from 72 to 84 BPM when measurement errors are taken into account.

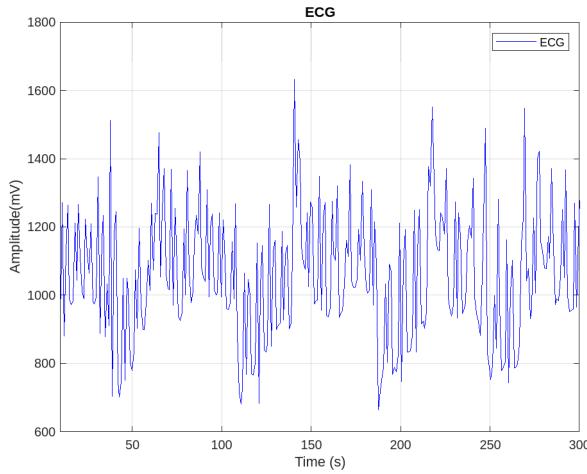


Figure 4.13 Stairs Down ECG Pattern

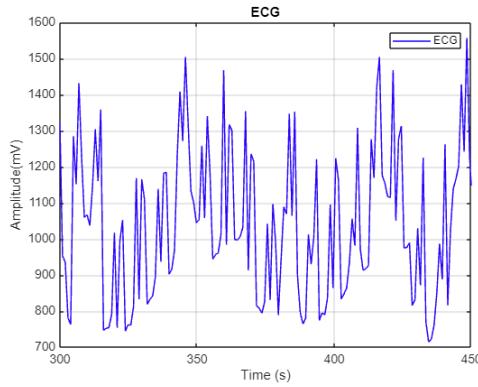
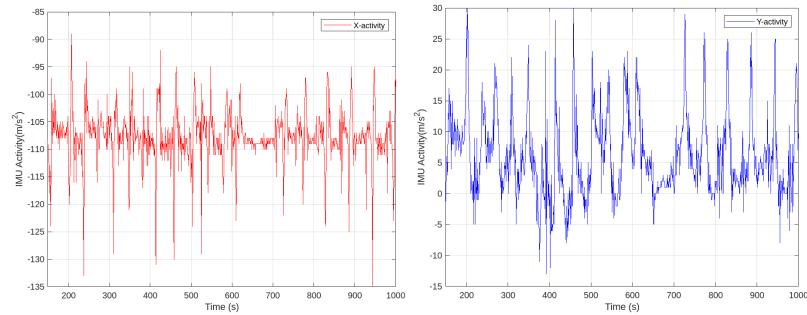


Figure 4.14 Stairs Down ECG Pattern

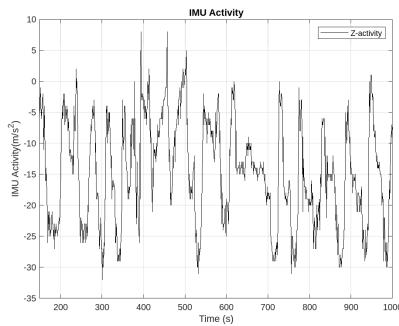
4.1.3.2 *Stairs Up Pattern*

The stair ascent data has the same characteristics as the stair descent data. However, slight changes are observed in the results obtained in figure 4.15a, 4.15b, and 4.15c. The body fights gravity to ascend stairs, which causes the heart rate to increase. The increased effort needed to raise the body is the cause of this rise. However, due to variations in muscle engagement and energy expenditure, the rate of growth can differ from the descent of the stairs.



(a) Stairs Up Pattern and X activity

(b) Stairs Up Pattern and Y activity



(c) Stairs Up Pattern and Z activity

Figure 4.15 Stairs Up Pattern and Activity Diagram

Acceleration data during ascent of stairs may exhibit patterns similar to descent of stairs, but with some variations. Although both activities involve changes in vertical acceleration, the magnitude and timing of these changes could differ. It is critical to remember that each physiological measurement, including heart rate monitoring, has a margin of error. Variability in recorded data might be introduced by factors that include signal noise, location, and sensor accuracy.

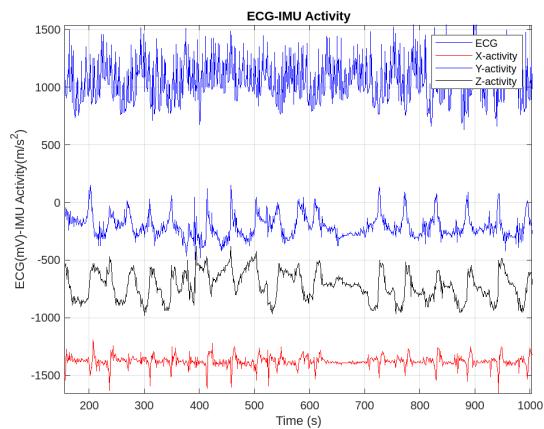


Figure 4.16 Stairs Up ECG Pattern and Activity Diagram

The variability of heart rate within the 72 to 90 BPM range indicates that the person's cardiovascular response varies during this exercise. These variations can be caused by variables such as the level of individual fitness, the intensity of exercise, the surrounding environment, and the state of one's hydration.

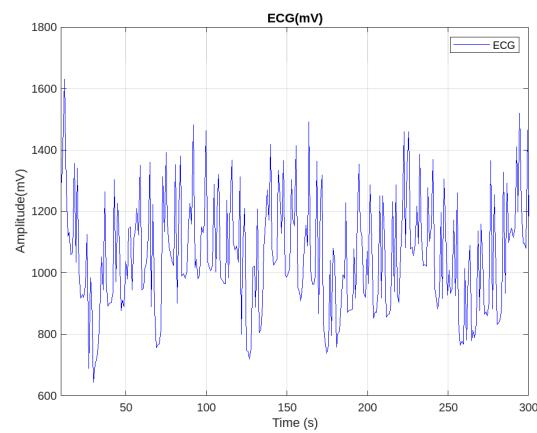


Figure 4.17 Stairs Up ECG Pattern

It is best to gather information from several trials and compare the outcomes to

guarantee the correctness of the observations. Furthermore, minimizing variability and improving the dependability of the results can be achieved by managing external variables and preserving constant settings during data collection.

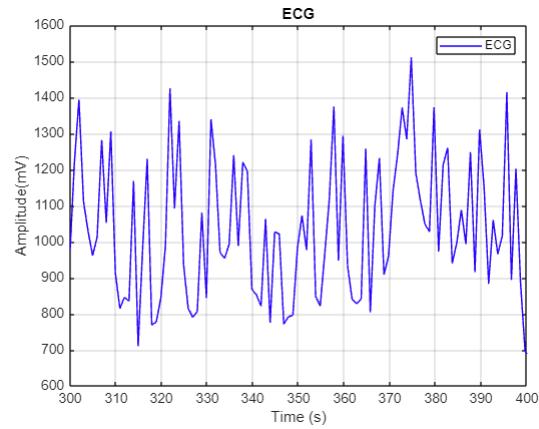


Figure 4.18 Stairs Up ECG Pattern

CHAPTER FIVE

ANALYSIS

The analyzes were performed using statistical analysis methods and machine learning algorithms.

5.1 Statistical Analysis

In these analyzes, mean, variance, standard deviation, root mean square, covariance, and correlation calculations, which are the basic components of the analysis, were used to determine similarity and dissimilarity within movement patterns to distinguish them from each other.

Mean is a statistical parameter that indicates a central trend, characterized by;

$$\mu = \frac{\sum_{i=0}^{N-1} x_i}{N} \quad (5.1)$$

Variance is the measure of the anticipated squared discrepancy from the mean, as defined by;

$$\sigma^2 = \frac{\sum_{i=0}^{N-1} (x_i - \mu)^2}{N} \quad (5.2)$$

Standard deviation represents the typical degree of variability within a dataset. It indicates, on average, the extent to which each value deviates from the mean.

$$\sigma = \sqrt{\frac{\sum_{i=0}^{N-1} (x_i - \mu)^2}{N}} \quad (5.3)$$

Covariance quantifies the combined change of x and y, which is described as:

$$Cov(x, y) = \frac{\sum_{i=1}^N |(x_i - \bar{x})(y_i - \bar{y})|}{N - 1} \quad (5.4)$$

Correlation evaluates the direction and intensity of the connection between A and B, which is described as:

$$Cor(x, y) = \frac{Cov(x, y)}{\sigma_x \sigma_y} \quad (5.5)$$

where σ_x and σ_y are the standard deviations of x and y.

The results obtained using the above formulae are as shown in table 5.1. As can be seen in this table, the movements are compared with each other and the calculation results obtained are marked on the table.

Table 5.1 Statistical data for measurement values

Correlation	Results	Variance(Run)	Variance(Jump)	Covariance
Run(x) and Jump(x)	-0.0063	784.5	3528.4	-10.6
Run(y) and Jump(y)	0.0028	2652.8	172.2	1.9
Run(z) and Jump(z)	-0.0325	106.226	140.9265	-3.9739
		Variance(Run)	Variance(Down)	
Run(x) and Stairdown(x)	-0.0021	804.7435	30.2717	-0.3263
Run(y) and Stairdown(y)	0.0222	2831.5	372	7.2
Run(z) and Stairdown(z)	-0.028	104.4689	32.5354	-1.2493
		Variance(Run)	Variance(Up)	
Run(x) and Stairup(x)	-0.0021	812.3511	31.7577	-0.3436
Run(y) and Stairup(y)	0.0024	2743.4	472	8.1
Run(z) and Stairup(z)	-0.0181	104.4689	61.579	-1.4544
		Variance(Jump)	Variance(Down)	
Jump(x) and Stair Down(x)	-0.0178	3637.9	34.8	-6.3
Jump(y) and Stair Down(y)	-0.002	174.3616	41.4081	-0.1731
Jump(z) and Stair Down(z)	-0.0101	143.088	31.513	-0.6806
		Variance(Jump)	Variance(Up)	
Jump(x) and Stair Up(x)	0.0245	3637.9	34.6	8.7
Jump(y) and Stair Up(y)	-0.0121	174.36	49.73	-1.131
Jump(z) and Stair Up(z)	0.0403	143.08	62.784	3.816
		Variance(Down)	Variance(Up)	
Stair Down(x) and Stair Up(x)	-0.0316	34.767	34.581	-1.0962
Stair Down(y) and Stair Up(y)	0.0294	41.408	49.733	1.3326
Stair Down(z) and Stair Up(z)	-0.0335	31.513	62.784	-1.49

5.1.1 Running-Jumping Pattern Analysis

When the data in table 5.1 are analysed, it can be clearly seen in the comparison of variance between run and jump that the change in the x-axis is greater in the case of jump. However, it is necessary to look at the other axes before we can say that this is directly jump. When the change in the y-axis is also considered, a larger value is obtained on the y-axis in the run case. So if we want to make a movement classification, when we check these two values, we can roughly determine which one

is jump and which one is run. In this case, as can be seen in the table, the data on the x and y axis are available for the classification between jump and run.

There are also correlation results obtained as can be seen in the results section in the table. When we look at these results, it can be said that there is a completely uncorrelated situation in the case of run-and-jump. Because correlation can take values between -1, 0 or +1. It can be called uncorrelated because it is very close to zero. In addition, when we examine the covariance values, although the movements on the x-axis are in the opposite direction, the movement on the y-axis is similar. This part can be used as a decision mechanism.

When Table 5.1 and Table 5.2 are compared, it can be seen that it is possible to make a classification for run and jump. In the case of jump, the variance value on the y-axis is large, while in the case of run, the variance value on the x-axis is large. In addition, when we look at the covariance value, the covariance on the y-axis is obtained as a positive value and this value can be used as a classification data.

5.1.2 *Running-Stairs Up and Down Pattern Analysis*

When run and stairs down are compared, it can be clearly seen that the variance values of running activity are higher in all axes. In addition, when the correlation status is examined, there is almost an uncorrelated situation. In this case, we can determine whether the movement is a running activity or a downhill task. When looking at the values, the result of comparing the two movements is easily determined, and the run movement is easily determined.

When the data is examined in table 5.1, the result obtained is almost similar to the stair descent case, and the values are almost similar. In this case, it is easily seen that the movement is running. In addition, the variance values on the y-axis in the run and stair ascent and descent cases are positive. In this case, it can be predicted that there may be confusion with the jump situation, but the way to separate this is to compare the values on the x axis. When we compare the variance of the x-axis in the jump state with the value of the x-axis in the ascending and descending state, it is seen that

it is in the ascending and descending state.

In order to make a decision between itself for stair ascent and descent, its x, y, and z values are compared again. Since the movements on the x-axis are very close to each other, the decision is made on the basis of the movements on the y and z axes.

5.1.3 Jumping-Stair Up and Down Pattern Analysis

Comparison of the jumping and climbing parameters is given in table 5.1. When you look at this table, When comparing the results obtained in the x, y and z axis, the variance movements of the axis in the jumping state were obtained higher. This was done to avoid confusion between the data obtained and the working state. The calculated variance values on the y and x axes can be compared.

In addition, the y and z parameters can be used to distinguish between up and down, and when we look at the covariance value, a positive x value is obtained. When viewed according to this value, the stairs-up status can be easily observed.

The jumping and stairs situation is also investigated in table 5.2. Jumping and stairs down and up in the jumping and stairs down and up situation has been completely obtained a result in the match case. In this case, a classification can be obtained by looking at the three data x, y, and z. Although some values are very close to each other, large differences between the values on the x and y axes will play a decisive role.

5.1.4 Stairs Up and Down Analysis

As can be seen in the last part of table 5.1, only up-and-down states were observed. Accordingly, we can classify the up situation using the data on the y and z axes. Here, the variance values are higher than the down situation, and these values can be used to make a distinction. When we look at the comparisons in the Jump and Run situations, we can easily understand that it is in the stair case since the values are small.

It will be a bit of a challenge to make the classification in the case of stairs down and up. However, considering the variance value and covariance value of the y axis, a rough inference of movement can be made. The covariance value was obtained as positive on the y-axis. When table 5.1 and table 5.2 are compared, considering the margin of error of these values, the most appropriate axis for classification is the y axis.

Table 5.2 Second measurement and statistical data analysis

Correlation	Results	Variance(Run)	Variance(Jump)	Covariance
Run(x) and Jump(x)	-0.0108	1382.4	3262.8	-0.023
Run(y) and Jump(y)	0.0226	183.04	161.76	3.8941
Run(z) and Jump(z)	0.0011	141.5	59.02	0.1027
		Variance(Run)	Variance(Down)	
Run(x) and Stairdown(x)	0.0082	1212.5	81.3	2.6
Run(y) and Stairdown(y)	0.0106	181.46	191.36	-1.244
Run(z) and Stairdown(z)	0.0027	128.94	58.08	0.235
		Variance(Run)	Variance(Up)	
Run(x) and Stairup(x)	-0.0019	1212.5	150.7	-0.8
Run(y) and Stairup(y)	-0.005	181.46	137.37	-0.78
Run(z) and Stairup(z)	0.0082	128.94	259.61	8.51
		Variance(Jump)	Variance(Down)	
Jump(x) and Stair Down(x)	0.000199	3262.8	76.9	0.1
Jump(y) and Stair Down(y)	-0.0018	161.76	76.87	-0.2
Jump(z) and Stair Down(z)	-0.017	59.02	58.94	-1
		Variance(Jump)	Variance(Up)	
Jump(x) and Stair Up(x)	0.0286	3268.4	77	14.4
Jump(y) and Stair Up(y)	0.021	162.03	71.51	2.26
Jump(z) and Stair Up(z)	-0.0073	59.11	59.02	-0.43
		Variance(Down)	Variance(Up)	
Stair Down(x) and Stair Up(x)	-0.0047	150.6	81.25	-0.52
Stair Down(y) and Stair Up(y)	0.0242	137.3	191.3	3.92
Stair Down(z) and Stair Up(z)	0.0572	259.6	58.08	7.027

5.1.5 *Running BPM analysis*

When we look at the heartbeat graph in figure 5.1 against the frequency graph on the x, y and z axis in the running state, it is seen that the movement on the y and z axis occurs more frequently. When we look at the heartbeat against this, it usually oscillates between 80BPM and 70BPM. Assuming that a normal human heartbeat is between 60 and 66 BPM, there is a noticeable increase.

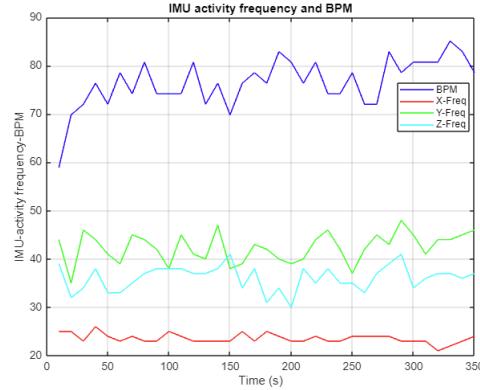


Figure 5.1 Running Pattern vs BPM comparision

5.1.6 Jumping BPM analysis

When we look at the comparison in the jumping movement in figure 5.2, while we expect a movement in the z axis, the results are very close to each other in the x, y and z axis. However, the determining factor here is the heartbeat. Due to the effort spent in the jumping movement, the heart rate can reach around 100 BPM. According to these results, it is seen that the similarity of the movement in the x, y, and z axes is not due to the absence of any displacement or acceleration.

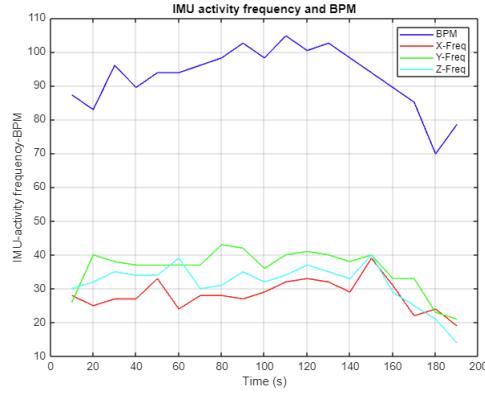


Figure 5.2 Jumping Pattern vs BPM comparision

5.1.7 Stairs BPM analysis

5.1.7.1 Stairs Up BPM analysis

When we look at the comparison in the stair up activity in figure 5.3, the heart rate in the stair climbing situation is close to the running situation. Although there is a slight increase in heart rate at the beginning, it is almost suitable for the running state as a

result of the measurement taken. In addition, when the x, y and z axes are considered, the frequency value on the y axis is higher because there is movement in that direction when climbing stairs. Since there will be no excessive acceleration change in the x and z axes during stairs climbing, they are closed to each other.

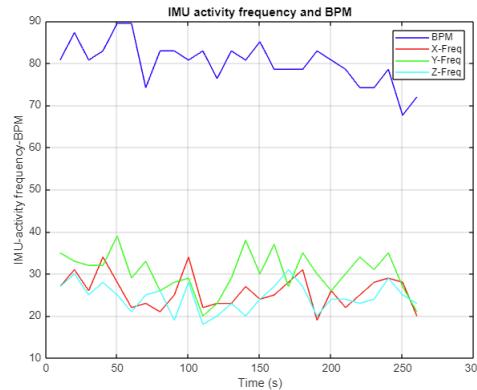


Figure 5.3 Stairs Up Pattern vs BPM comparision

5.1.7.2 Stairs Down BPM analysis

In the stair down movement, in contrast to the stair up movement in figure 5.4, a decrease in heart rate is observed. The main reason for this is that the effort spent in the up stairs is less in the down stairs. When we compare stair up and down, it can be seen that the heartbeat in the down state will be observed between 70BPM and 80BPM in longer measurements. In addition, when we observe the movements on the x, y, and z axes, the frequency of movements is very close to each other, but the movement on the y axis is observed to be slightly higher.

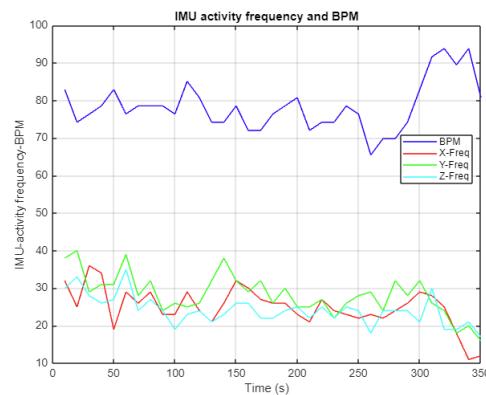


Figure 5.4 Stairs Down Pattern vs BPM comparision

5.2 Machine Learning Analysis and Discussion

In order to conduct the categorization procedure, the time series of activity collected for each type of activity were separated into smaller segments using sliding windows. To ascertain the activity type over a duration of approximately 15 seconds, the window width is set to 15 steps and the offset value is set to 1 step for the sliding windows applied to the necessary time series. Therefore, a total of 4209 15-step time series were collected for the activity of Jumping, 8971 for Running, 8394 for Stairs Down, and 7263 for Stairs Up. Due to the significance of changes in each axis during Stairs Down and Up activities, the time series analysis of these activities included the average of the three axes. Every divided time series is classified according to the corresponding motion. For training purposes, 70% of the labeled data from each class was chosen at random, while the remaining 30% was reserved for testing.

In order to classify ECG data, the ECG time series of the relevant activity were subdivided into smaller segments using sliding windows of 15 lengths and 1 step offset. A total of 34446 time series were collected, including 4207 for Jumping, 13868 for Running, 8658 for Stairs Down, and 7713 for Stairs Up. Like the time series of activity, 70% of the random data was allocated for training, while the remaining 30% was reserved for testing.

Figure 5.5 displays the confusion matrices for the activity and ECG time series that were generated by classification using the decision tree method. Furthermore, table 5.3 presents the classification metrics that were obtained.

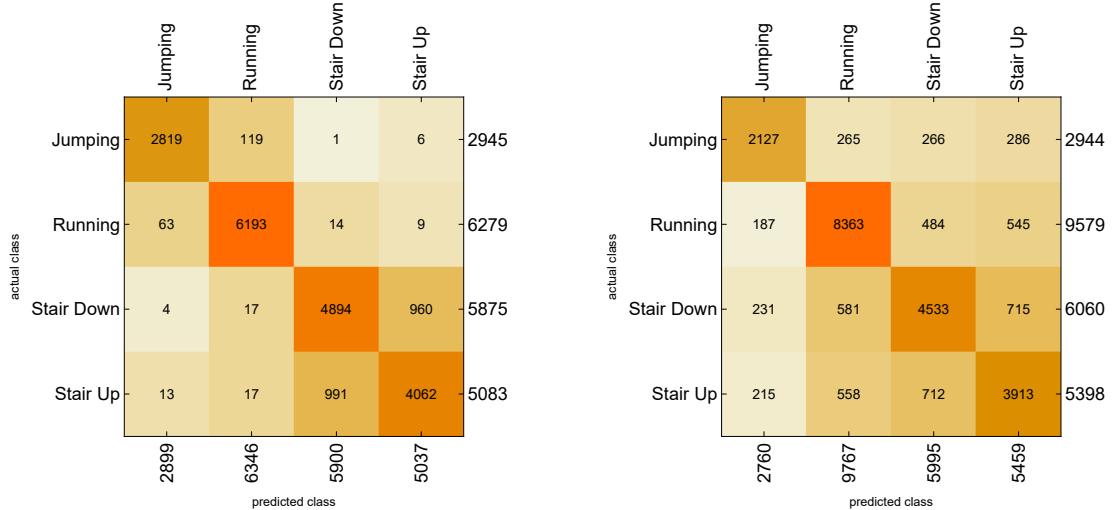


Figure 5.5 Confusion matrices for activity time series classification (on the left) and ECG time series (on the right) for decision tree.

Table 5.3 Classification performance metrics for decision tree method.

	Activity				ECG			
	Jumping	Running	Stair Down	Stair Up	Jumping	Running	Stair Down	Stair Up
Area Under ROC	0.984968	0.99503	0.948484	0.942771	0.909348	0.930176	0.88552	0.877156
Mean Cross Entropy	0.182689	0.130714	0.608811	0.662104	0.947932	0.6434	0.899856	0.991949
Class Rejection Rate	0.	0.	0.	0.	0.	0.	0.	0.
F1	0.96475	0.981069	0.831253	0.802767	0.745792	0.864571	0.752053	0.720825
False Discovery	0.0275957	0.0241097	0.170508	0.193568	0.229348	0.143749	0.24387	0.283202
False Negative Rate	0.0427844	0.0136964	0.166979	0.200866	0.277514	0.126944	0.25198	0.275102
False Positive Rate	0.00464118	0.0110048	0.0703152	0.0645738	0.0300898	0.0974865	0.0815803	0.0831943
MCC	0.958824	0.972483	0.761754	0.736786	0.711997	0.773131	0.668831	0.639158
Negative Predictive Value	0.99271	0.993784	0.931312	0.932585	0.9615	0.914451	0.915101	0.919825
Precision	0.972404	0.97589	0.829492	0.806432	0.770652	0.856251	0.75613	0.716798
Recall	0.957216	0.986304	0.833021	0.799134	0.722486	0.873056	0.74802	0.724898
Specificity	0.995359	0.988995	0.929685	0.935426	0.96991	0.902514	0.91842	0.916806

Figure 5.6 displays the confusion matrices for the activity and ECG time series that were generated by classification using the gradient boosted tree method, in particularly XGBoost method. Furthermore, table 5.4 presents the classification metrics that were obtained.

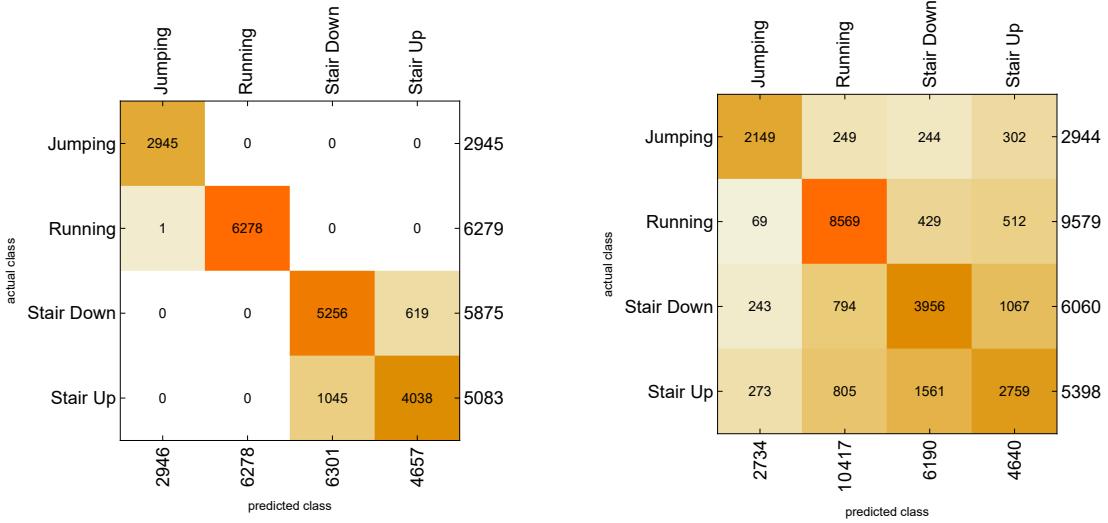


Figure 5.6 Confusion matrices for activity time series classification (on the left) and ECG time series (on the right) for XGBoost.

Table 5.4 Classification performance metrics for XGBoost method.

	Activity				ECG			
	Jumping	Running	Stair Down	Stair Up	Jumping	Running	Stair Down	Stair Up
Area Under ROC	1.	1.	0.974825	0.972429	0.959952	0.952789	0.885536	0.857324
Mean Cross Entropy	0.000194838	0.001214	0.325713	0.405877	0.86274	0.436881	0.878273	1.03969
Class Rejection Rate	0.	0.	0.	0.	0.	0.	0.	0.
F1	0.99983	0.99992	0.863338	0.829158	0.756957	0.857071	0.645878	0.549711
False Discovery	0.0000339443	0.	0.165847	0.132918	0.213972	0.177402	0.360905	0.405388
False Negative Rate	0.	0.0000159261	0.105362	0.205587	0.270041	0.105439	0.347195	0.488885
False Positive Rate	0.0000580147	0.	0.0730412	0.0409961	0.0278081	0.128316	0.124658	0.101222
MCC	0.999801	0.999884	0.805419	0.776248	0.725009	0.757137	0.524477	0.433358
Negative Predictive Value	1.	0.999928	0.955407	0.932689	0.962583	0.925538	0.881738	0.863554
Precision	0.999661	1.	0.834153	0.867082	0.786028	0.822598	0.639095	0.594612
Recall	1.	0.999841	0.894638	0.794413	0.729959	0.894561	0.652805	0.511115
Specificity	0.999942	1.	0.926959	0.959004	0.972192	0.871684	0.875342	0.898778

Figure 5.7 exhibits the confusion matrices for the activity and ECG time series that were produced by classification using the logistic regression technique. Moreover, table 5.5 displays the classification metrics that were acquired.

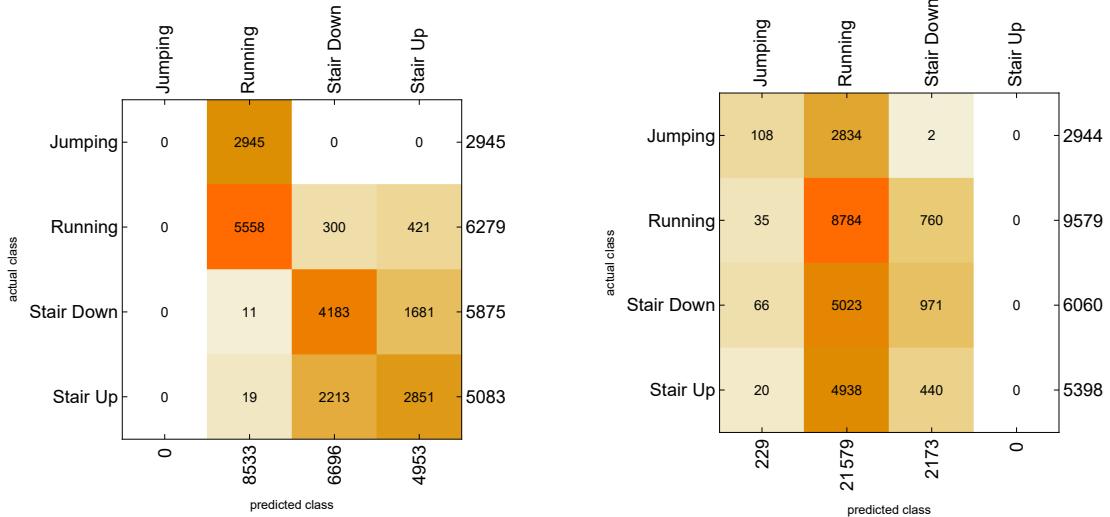


Figure 5.7 Confusion matrices for activity time series classification (on the left) and ECG time series (on the right) for logistic regression.

Table 5.5 Classification performance metrics for logistic regression method.

	Activity				ECG			
	Jumping	Running	Stair Down	Stair Up	Jumping	Running	Stair Down	Stair Up
Area Under ROC	0.929509	0.886907	0.874423	0.854187	0.727614	0.562867	0.59641	0.5954
Mean Cross Entropy	1.1257	0.79607	0.632569	0.826695	1.83694	0.904267	1.32623	1.46961
Class Rejection Rate	0.	0.	0.	0.	0.	0.	0.	0.
F1	0.	0.750473	0.6655	0.568155	0.0680744	0.563836	0.23588	0.
False Discovery	Indeterminate	0.348646	0.375299	0.424389	0.528384	0.592938	0.553152	Indeterminate
False Negative Rate	1.	0.114827	0.288	0.439111	0.963315	0.082994	0.839769	1.
False Positive Rate	0.	0.213983	0.175648	0.139215	0.00575177	0.888418	0.0670721	0.
MCC	Indeterminate	0.628999	0.517462	0.425346	0.104379	0.0466389	0.141028	Indeterminate
Negative Predictive Value	0.854078	0.938106	0.874537	0.853438	0.8806	0.669026	0.766645	0.774905
Precision	Indeterminate	0.651354	0.624701	0.575611	0.471616	0.407062	0.446848	Indeterminate
Recall	0.	0.885173	0.712	0.560889	0.0366848	0.917006	0.160231	0.
Specificity	1.	0.786017	0.824352	0.860785	0.994248	0.111582	0.932928	1.

Figure 5.8 provides the confusion matrices for the activity and ECG time series that were produced by classification using the hidden Markov models. Moreover, table 5.6 displays the classification metrics that were acquired.

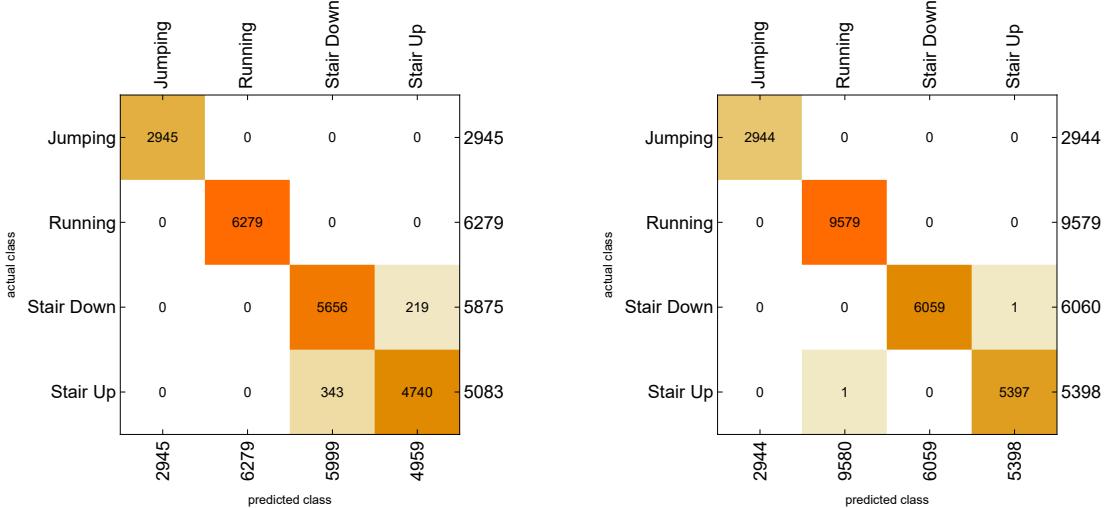


Figure 5.8 Confusion matrices for activity time series classification (on the left) and ECG time series (on the right) for hidden Markov model.

Table 5.6 Classification performance metrics for hidden Markov method.

	Activity				ECG			
	Jumping	Running	Stair Down	Stair Up	Jumping	Running	Stair Down	Stair Up
Area Under ROC	1.	1.	0.99655	0.996222	1.	1.	1.	1.
Mean Cross Entropy	3.09078E-9	1.56183E-14	0.102278	0.183789	1.15138E-10	5.33893E-7	0.000940715	0.000205988
Class Rejection Rate	0.	0.	0.	0.	0.	0.	0.	0.
F1	1.	1.	0.95267	0.944035	1.	0.999948	0.999917	0.999815
False Discovery	0.	0.	0.0571762	0.0441621	0.	0.000104384	0.	0.000185254
False Negative Rate	0.	0.	0.0372766	0.0674798	0.	0.	0.000165017	0.000185254
False Positive Rate	0.	0.	0.0239743	0.0145043	0.	0.0000694348	0.	0.0000538126
MCC	1.	1.	0.933049	0.925629	1.	0.999913	0.99989	0.999761
Negative Predictive Value	1.	1.	0.984559	0.977468	1.	1.	0.999944	0.999946
Precision	1.	1.	0.942824	0.955838	1.	0.999896	1.	0.999815
Recall	1.	1.	0.962723	0.93252	1.	1.	0.999835	0.999815
Specificity	1.	1.	0.976026	0.985496	1.	0.999931	1.	0.999946

Figure 5.9 displays the confusion matrices for the activity and ECG time series that resulted from classification using the Naive Bayes approach. Furthermore, Table 5.7 presents the obtained classification metrics.

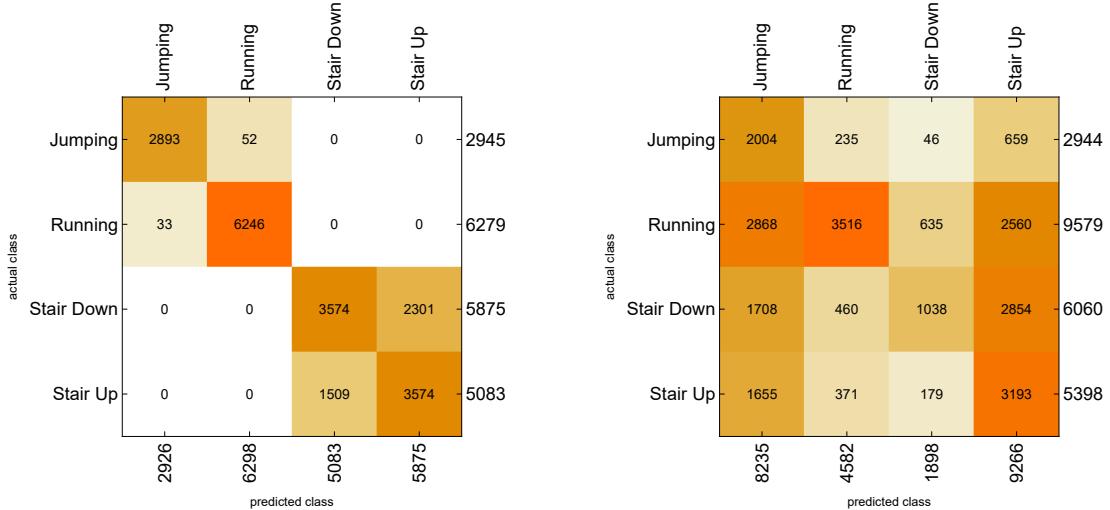


Figure 5.9 Confusion matrices for activity time series classification (on the left) and ECG time series (on the right) for Naive Bayes approach.

Table 5.7 Classification performance metrics for Naive Bayes approach.

	Activity					ECG			
	Jumping	Running	Stair Down	Stair Up	Jumping	Running	Stair Down	Stair Up	
Area Under ROC	0.999417	0.999665	0.907212	0.898295	0.791906	0.777545	0.66633	0.6964	
Mean Cross Entropy	0.0938986	0.0516214	0.608161	0.622479	1.14777	1.10641	1.30831	1.21467	
Class Rejection Rate	0.	0.	0.	0.	0.	0.	0.	0.	
F1	0.985522	0.993242	0.652309	0.652309	0.358529	0.496575	0.26087	0.435488	
False Discovery	0.0112782	0.00825659	0.296872	0.39166	0.756648	0.232649	0.453109	0.655407	
False Negative Rate	0.017657	0.00525561	0.39166	0.296872	0.319293	0.632947	0.828713	0.408485	
False Positive Rate	0.00191449	0.0037402	0.105473	0.152394	0.296192	0.0740175	0.0479884	0.326804	
MCC	0.983065	0.990185	0.526257	0.526257	0.26574	0.365069	0.198472	0.22705	
Negative Predictive Value	0.996987	0.997623	0.847606	0.894527	0.940302	0.687458	0.772585	0.850153	
Precision	0.988722	0.991743	0.703128	0.60834	0.243352	0.767351	0.546891	0.344593	
Recall	0.982343	0.994744	0.60834	0.703128	0.680707	0.367053	0.171287	0.591515	
Specificity	0.998086	0.99626	0.894527	0.847606	0.703808	0.925983	0.952012	0.673196	

Figure 5.10 exhibits the confusion matrices for the activity and ECG time series that were produced by classification using the Random Forest method. Moreover, table 5.8 displays the classification metrics that were acquired.

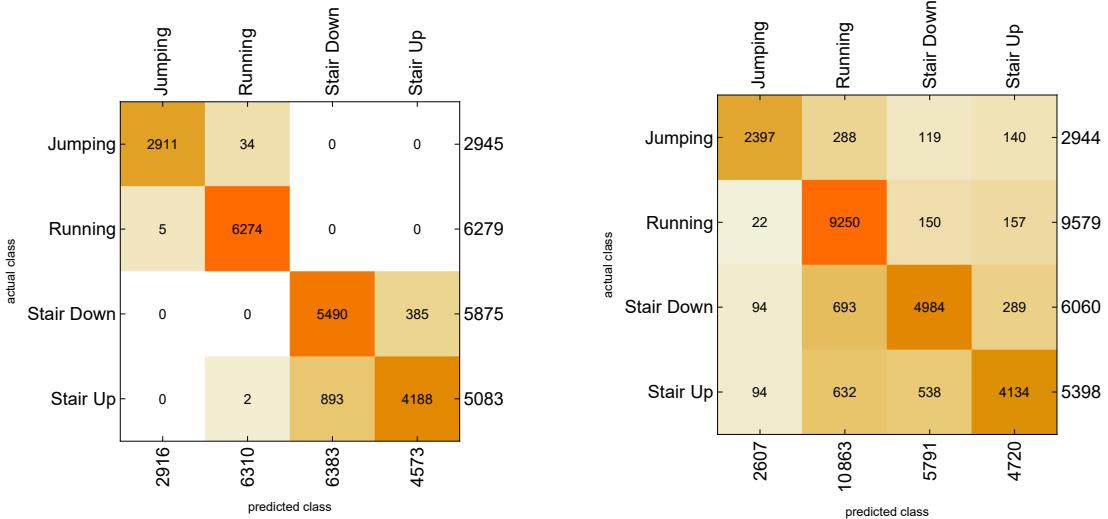


Figure 5.10 Confusion matrices for activity time series classification (on the left) and ECG time series (on the right) for random forests.

Table 5.8 Classification performance metrics for random forest method.

	Activity					ECG			
	Jumping	Running	Stair Down	Stair Up	Jumping	Running	Stair Down	Stair Up	
Area Under ROC	0.999916	0.999946	0.98279	0.98124	0.982742	0.981571	0.96199	0.961598	
Mean Cross Entropy	0.0471246	0.0125656	0.205457	0.382575	0.638939	0.22755	0.595098	0.717563	
Class Rejection Rate	0.	0.	0.	0.	0.	0.	0.	0.	
F1	0.993346	0.996743	0.895742	0.86744	0.863628	0.905	0.84111	0.817158	
False Discovery	0.00171468	0.00570523	0.139903	0.0841898	0.0805524	0.148486	0.139354	0.124153	
False Negative Rate	0.011545	0.000796305	0.0655319	0.176077	0.185802	0.034346	0.177558	0.234161	
False Positive Rate	0.000290074	0.00258937	0.062417	0.0254984	0.00998241	0.111998	0.045031	0.0315342	
MCC	0.992232	0.995273	0.851891	0.827904	0.847852	0.839935	0.78936	0.77133	
Negative Predictive Value	0.998031	0.99964	0.972099	0.942661	0.974408	0.97492	0.940847	0.934375	
Precision	0.998285	0.994295	0.860097	0.91581	0.919448	0.851514	0.860646	0.875847	
Recall	0.988455	0.999204	0.934468	0.823923	0.814198	0.965654	0.822442	0.765839	
Specificity	0.99971	0.997411	0.937583	0.974502	0.990018	0.888002	0.954969	0.968466	

Figure 5.11 exhibits the confusion matrices for the activity and ECG time series that were produced by classification using the Support Vector Machine with RBF kernel. Moreover, table 5.8 displays the classification metrics that were acquired.

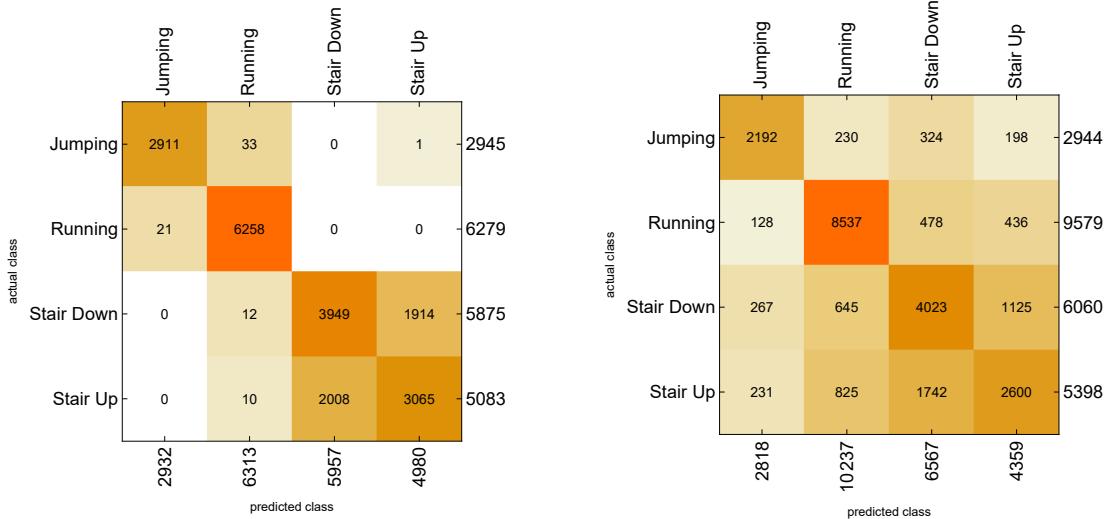


Figure 5.11 Confusion matrices for activity time series classification (on the left) and ECG time series (on the right) for support vector machines.

Table 5.9 Classification performance metrics for support vector machine method.

	Activity				ECG			
	Jumping	Running	Stair Down	Stair Up	Jumping	Running	Stair Down	Stair Up
Area Under ROC	0.999544	0.999404	0.895008	0.884881	0.954903	0.949435	0.873249	0.850105
Mean Cross Entropy	0.0813252	0.0155505	0.701713	0.69647	0.924909	0.37166	0.901233	1.15719
Class Rejection Rate	0.	0.	0.	0.	0.	0.	0.	0.
F1	0.990641	0.993964	0.667512	0.609162	0.760847	0.861627	0.637206	0.532951
False Discovery	0.00716235	0.00871218	0.337082	0.384538	0.222143	0.166064	0.387392	0.403533
False Negative Rate	0.011545	0.00334448	0.32783	0.39701	0.255435	0.10878	0.336139	0.51834
False Positive Rate	0.00121831	0.00395598	0.140351	0.12683	0.0297571	0.118039	0.141956	0.0946564
MCC	0.98905	0.991236	0.529666	0.479427	0.728436	0.765614	0.5086	0.419106
Negative Predictive Value	0.998029	0.998486	0.864605	0.867254	0.964466	0.924185	0.883025	0.857405
Precision	0.992838	0.991288	0.662918	0.615462	0.777857	0.833936	0.612608	0.596467
Recall	0.988455	0.996656	0.67217	0.60299	0.744565	0.89122	0.663861	0.48166
Specificity	0.998782	0.996044	0.859649	0.87317	0.970243	0.881961	0.858044	0.905344

5.2.1 Discussion

The table labeled as Table 5.3 presents a summary of the classification outcomes achieved by implementing the Decision Tree method on two sets of data: Activity and ECG. Each dataset has four distinct classes: Jumping, Running, Stairs Down, and Stairs Up. The AUC values measure the model's capacity to differentiate across classes by evaluating the ROC curve. Both the Activity and ECG data exhibit high AUC values, which indicate exceptional discriminative capability. Running demonstrates the greatest AUC values in both datasets, indicating improved categorization ability for this activity. The mean cross entropy quantifies the average level of uncertainty in predictions for each class. Smaller values indicate higher levels of confidence in the predictions. Running regularly has the lowest mean cross entropy, indicating a better level of confidence in its classification compared to other classes. This is particularly visible in the Activity data. The lack of class rejection suggests that the Decision Tree algorithm successfully classified all instances without any rejections. Indications imply that the model generated predictions for every event in both the Activity and ECG datasets. The F1 score achieves equilibrium between precision and recall. Running consistently yields the highest F1 scores for both datasets, indicating a favorable equilibrium between precision and recall for this particular class. The rates of False Discovery, False Negative, and False Positive quantify distinct forms of misclassifications. In general, lower rates are indicative with superior performance. The Stairs Up class exhibits a considerably higher false discovery rate in both datasets, suggesting a greater number of erroneous positive predictions for this class compared to other classes. The MCC quantifies the degree of correlation between the actual and anticipated classifications, where values closer to 1 indicate superior performance. Consistently running shows higher Matthews Correlation Coefficient (MCC) values, indicating superior overall categorization performance for this exercise. The measures of Negative Predictive Value, Precision, Recall, and Specificity offer valuable information about the model's ability to accurately identify true negatives, true positives, and minimize the occurrence of false positives and false negatives. Overall, Running exhibits superior values in these parameters, indicating superior performance in comparison to other classes. To

summarize, the Decision Tree algorithm demonstrates robust performance in accurately classifying both Activity and ECG data into their respective categories. Consistently, running is identified as the most highly ranked activity, exhibiting high AUC values, low prediction uncertainty, and a fair trade-off between precision and recall. Nevertheless, Stairs Up exhibits comparatively inferior performance measures, indicating possible difficulties in reliably differentiating this activity from others.

The classification results obtained by applying the XGBoost algorithm to the activity and ECG data are presented in Table 5.4. Each dataset contains four classes: Jumping, Running, Stairs Down, and Stairs Up. Both the Activity and ECG data regularly exhibit high AUC values, with values approaching 1 for Jumping and Running. This suggests exceptional categorization ability. Nevertheless, there is a marginal decline in the AUC values for Stairs Down and Stairs Up, indicating a somewhat weaker capacity to distinguish between these groups in comparison to Jumping and Running. Jumping and Running demonstrate remarkably low mean cross entropy values, suggesting a high level of certainty in their categorization. Nevertheless, the classifications Stairs Down and Stairs Up exhibit comparatively elevated mean cross entropy values, which suggest a greater level of uncertainty in their predictions. The lack of class rejection suggests that the XGBoost algorithm successfully classified all instances without any rejection. This implies that the model made predictions for all instances in both the Activity and ECG datasets. Jumping and running regularly produce high F1 scores, indicating a favorable equilibrium between precision and memory for these categories. Nevertheless, the F1 scores for Stairs Down and Stairs Up indicate a less balanced performance for these classes. Jumping and Running typically demonstrate extremely low rates of false discovery, false negatives, and false positives, suggesting few instances of misclassification. Nevertheless, the classifications of Stairs Down and Stairs Up have comparatively elevated rates, suggesting difficulties in accurately differentiating between these categories. The activities of jumping and running consistently demonstrate high Matthews Correlation Coefficient (MCC) values, indicating a strong overall performance in classification. However, the MCC values for Stairs Down and Stairs

Up are considerably lower, suggesting that these classes have worse performance in comparison. These metrics offer valuable information about the model’s ability to accurately identify true negatives and true positives and minimize the occurrence of false positives and false negatives. Jumping and Running often exhibit superior values for Negative Predictive Value, Precision, Recall, and Specificity, showing better performance in comparison to Stairs Down and Stairs Up. To summarize, the XGBoost algorithm exhibits robust performance in accurately classifying both Activity and ECG data into their respective categories. Jumping and running consistently stand out as highly distinguished activities, exhibiting high AUC values, low prediction uncertainty, and a well-balanced trade-off between precision and recall. Nevertheless, Stairs Down and Stairs Up demonstrate relatively inferior performance measures, indicating possible difficulties in reliably differentiating these tasks from others.

Table 5.5 presents a summary of the classification findings obtained by utilizing logistic regression to classify both Activity and ECG data. The AUC values in both the Activity and the ECG data vary from moderate to high, suggesting that they have an acceptable discriminative ability. Running typically has lower AUC values in comparison to other classes, indicating difficulties in reliably differentiating Running from other activities. In this scenario, it can be observed that all classes in both the activity and the ECG data have rather high mean cross-entropy values, indicating a notable level of uncertainty in the predictions for all classes. The lack of class rejection suggests that the logistic regression model successfully classified all cases without any rejection. This implies that the model made predictions for all occurrences in both the Activity and ECG datasets. The F1 score offers an equilibrium between precision and recall. Most of the classes in this scenario demonstrate F1 scores that are moderate to poor, suggesting an imbalance between precision and recall. Significantly, the Jumping activity has an F1 score of 0, indicating inadequate performance in accurately categorizing instances of this action. In general, false discovery, false negative and false positive rates are often elevated for most categories, highlighting the difficulties in effectively differentiating between

these categories. The MCC metric quantifies the correlation between the actual and projected classifications, where values closer to 1 signify superior performance. However, most classes demonstrate MCC values that are around 0 or ambiguous, suggesting inadequate overall classification performance. Many classes exhibit different levels of imbalance between precision and recall, indicating less than ideal performance. In summary, the logistic regression model has varied performance in accurately classifying both Activity and ECG data into their respective categories. Although many classes have satisfactory discriminatory capability and a well-balanced trade-off between accuracy and memory, others show considerable uncertainty in prediction and disparities between precision and recall, suggesting difficulties in effectively differentiating across activities. Additional optimization or investigation of alternative categorization methods may be necessary to enhance the accuracy of classification.

The classification results obtained using the Hidden Markov Model (HMM) to classify both activity and ECG data, each including four classes, are presented in Table 5.6. Both the activity and the ECG data sets exhibit flawless discriminative capacity, as all classes earn perfect AUC scores of 1. This indicates that the Hidden Markov Model (HMM) successfully distinguishes various activities with a high level of precision. The model's cross-entropies for most classes in both activity and ECG data are remarkably low, indicating highly confident predictions. The lack of class rejection suggests that the Hidden Markov Model (HMM) successfully classified all instances without any rejection. This shows that the model made confidence predictions for all instances in both the activity and the ECG datasets. The HMM demonstrates great accuracy in properly detecting instances of each activity, as seen by the high F1 scores close to 1 for most classes in both datasets. The model demonstrates resilience, as evidenced by the remarkably low values for false discovery, false negative rate, and false positive rate in most classes, indicating few misclassifications. The high values of Matthews Correlation Coefficient (MCC) near to 1 for most classes indicate an excellent overall performance of the Hidden Markov Model (HMM) in classification. The HMM demonstrates great accuracy and efficacy

in identifying activities, as seen by the high values for Negative Predictive Value, precision, recall, and specificity across most classes in both data sets. In summary, the hidden Markov model exhibits outstanding efficacy in accurately categorizing both activity and ECG data into their appropriate classes. The Hidden Markov Model (HMM) demonstrates strong and precise classification abilities across all activities, as evidenced by its flawless AUC values, low mean cross entropy, and few misclassifications.

The classification results obtained by applying the Naive Bayes algorithm to categorize Activity and ECG data, each including four classes, are presented in Table 5.7. Within the realm of Activity data, the AUC values exhibit a remarkably high level for the Jumping and Running classes, hence signifying an exceptional capacity for differentiation. Nevertheless, the AUC values for the Stair Down and Stair Up classes are comparatively lower, indicating less successful discriminating in comparison to the other classes. Similarly, the area under the curve (AUC) values are high for the Jumping and Running classes in ECG data, but lower for the Stair Down and Stair Up classes. For both the Activity and ECG data, the mean cross entropy is low for the Jumping and Running classes, which suggests that the predictions are very confident. Nevertheless, in the case of the Stair Down and Stair Up classes, the average cross entropy is elevated, suggesting a greater level of uncertainty in the forecasts. The Naive Bayes approach successfully classified all instances without any instances being rejected, indicating that the model provided predictions for all occurrences in both the Activity and ECG datasets. Across different courses, the F1 scores exhibit variation in this situation. Jumping and Running classes typically exhibit higher F1 ratings, indicating superior performance in these classes compared to Stair Down and Stair Up classes. Stair Down and Stair Up classes exhibit higher rates of false discovery and false negative compared to Jumping and Running classes in this scenario. For Jumping and Running classes, the MCC values are often high, indicating a strong classification performance. However, for Stair Down and Stair Up classes, the MCC values are substantially lower, suggesting a less consistent classification performance in these classes. The interpretation adheres to analogous

patterns as previously described, wherein the Jumping and Running classes typically exhibit superior performance in contrast to the Stair Down and Stair Up classes. To summarize, the Naive Bayes technique exhibits diverse levels of performance across several classes in both the Activity and ECG datasets. Jumping and Running courses typically demonstrate superior classification performance, whereas Stair Down and Stair Up classes often exhibit poorer performance metrics, suggesting difficulties in effectively differentiating between these activities.

The classification results obtained by utilizing the Random Forest approach to classify Activity and ECG data, each including four classes, are presented in Table 5.8. Both the Activity and ECG data show high AUC values for the Jumping and Running classes, which suggests exceptional discriminative capability. However, the AUC values for the Stair Down and Stair Up classes exhibit a little decrease, indicating a relatively less efficient differentiation ability compared to the other classes. For both the Activity and ECG data, the mean cross entropy is relatively low for the Jumping and Running classes. This suggests that the predictions made for these classes are very confident. However, in the case of the Stair Down and Stair Up classes, the average cross entropy is elevated, suggesting a greater level of uncertainty in the forecasts. The lack of class rejection suggests that the Random Forest approach successfully classified all instances in both the Activity and ECG datasets, indicating that the model made predictions for every case. For the Jumping and Running divisions, the F1 scores are consistently high, which suggests excellent performance. Nevertheless, the F1 ratings for the Stair Down and Stair Up classes exhibit a little decrease, indicating difficulties in accurately differentiating between these activities. Greater values for false discovery, false negative rate, and false positive rate suggest a higher number of misclassifications. Stair Down and Stair Up classes exhibit higher rates of false discovery and false negative compared to Jumping and Running classes in this scenario. In this instance, the MCC (Matthews Correlation Coefficient) values indicate that the classification performance is generally high for Jumping and Running classes, but significantly lower for Stair Down and Stair Up classes. This suggests that there are varied levels of classification performance for each class. The

interpretation exhibits analogous trends as previously noted, with the Jumping and Running categories often demonstrating superior performance in comparison to the Stair Down and Stair Up categories. To summarize, the Random Forest approach exhibits differing levels of effectiveness across several categories in both the Activity and ECG datasets. Jumping and Running classes typically provide superior classification performance, however Stair Down and Stair Up classes show significantly lower performance metrics, suggesting difficulties in effectively differentiating between these activities.

The classification results obtained by utilizing the Support Vector Machine (SVM) approach to classify Activity and ECG data, each including four classes, are presented in Table 5.9. The AUC values for both Activity and ECG data are consistently high for the Jumping and Running classes, suggesting a strong ability to distinguish between them. Nevertheless, the AUC values for the Stair Down and Stair Up classes exhibit a little decrease, indicating a relatively less efficient ability to differentiate compared to the other classes. For both Activity and ECG data, the mean cross entropy is low for the Jumping and Running classes, which suggests a high level of confidence in the predictions. However, in the case of the Stair Down and Stair Up classes, the average cross entropy is elevated, suggesting a greater level of uncertainty in the forecasts. The lack of class rejection suggests that the SVM model successfully classified all instances without any rejection, indicating that the model made predictions for all instances in both the Activity and ECG datasets. For both the Jumping and Running classifications, the F1 scores are consistently high, suggesting excellent performance. Nevertheless, the F1 ratings for the Stair Down and Stair Up classes exhibit a little decrease, indicating difficulties in accurately differentiating between these activities. Greater values for false discovery, false negative rate, and false positive rate suggest a higher number of misclassifications. Stair Down and Stair Up classes exhibit greater rates of both false discovery and false negative compared to Jumping and Running classes in this scenario. For Jumping and Running classes, the MCC values are generally high, indicating a strong classification performance. However, for Stair Down and Stair Up classes, the MCC values are slightly lower,

suggesting a slightly weaker classification performance in these classes. The interpretation exhibits analogous patterns as previously mentioned, with Jumping and Running classes typically manifesting. Jumping and Running classes typically provide superior classification performance, however Stair Down and Stair Up classes show significantly lower performance metrics, suggesting difficulties in effectively differentiating between these activities.

CHAPTER SIX

CONCLUSION

Based on our observations, the AD8232 microchip is suitable for the AFE function, since it generated a signal that proved advantageous for a prolonged use of a single-lead ECG tracking application. Using the MPU6050 inertial measurement sensor, it is also possible to determine the movement of the x, y, and z axes based on the study findings. The prototype motion detector developed for this study is represented by the x, y, and z axes. Due to the volume of data collected and the need for the ECG to be extremely precise (little errors could signal major problems), a doctor might not be able to accurately diagnose or evaluate heart function with the suggested device. However, a fundamental approach for assessing an athlete's performance is feasible given the data. Activity data make it possible to pinpoint an athlete's movements, including sprinting and jumping. Heart rate is also used in athletic activities. Heart rate and activity data were compared to determine the frequency of activities and fluctuations in BPM. Activity and ECG data can be used to determine an athlete's sprint performance and to spot signs of overtraining or athlete burnout. The amount of accelerations and decelerations that an athlete does can be determined using activity data gathered during their exercise.

Even if movement classification cannot be made completely precisely as a result of the measurements taken, it can be roughly estimated from the data obtained which movement corresponds to what. When we look at the variance and covariance values in the tables, according to the results we obtained, run and jump movements show a distinct feature and accordingly, we can easily distinguish these movements from stair down and up movements. In the comparison of run and jump in the table, we can roughly classify the jump movement by looking at the x-axis and the run movement by looking at the y-axis. In the case of stair up and down, the decisive part of the results obtained is the variance value on the y-axis and the positivity of the covariance value.

It seems possible that better results can be obtained with future improvements. To

briefly mention these, the electrode connection points and conductive threads on the T-shirt should be fixed at a sufficient level. In addition, a socket connection is required in the part of the PCB design where the electrode connections are located. In further studies, the number of ECG electrodes can be increased and the measurement values can be obtained with higher resolution.

When we look at the results obtained through machine learning, running activity can be easily observed due to the high AUC value and low prediction uncertainty in the results obtained with the decision tree.

A similar situation applies to the XGboost method, where jumping and running activities can be classified. However, it does not seem possible to classify any movement in the results obtained in the logistic regression method. In this case, optimization or an alternative categorization method should be developed.

In the HMM method, AUC values and low mean cross entropy in the results obtained help us to provide a strong and accurate classification. This method was able to classify all movements with a small error.

In the Naive Bayes method, jumping and running movements are easily classified, while stair up and down are not at a sufficient level.

The random forest method also gave very good results in jumping and running, but showed a low classification performance in stair up and down activities.

A similar situation applies to the support vector machine. Here too, jumping and running activities can be easily distinguished, but stair up and down activities are very difficult to classify.

When all the methods are examined, it is explained that the stair up and down

activity is difficult to classify and cannot be done properly, but any of the described methods can be used to classify jumping and running.

REFERENCES

- Zysset, C., Kinkeldei, T. W., Munzenrieder, N., Cherenack, K., & Troster, G. (2012). Integration method for electronics in woven textiles. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 2(7), 1107-1117.
- Tao, X. (Ed.). (2005). *Wearable electronics and photonics*. Elsevier..
- Hu, E., Kaynak, A.,& Li, Y. (2005). Development of a cooling fabric from conducting polymer coated fibres: Proof of concept. *Synthetic metals*, 150(2), 139-143.
- Khan, A., Hussain, M., Nur, O., & Willander, M. (2014). Fabrication of zinc oxide nanoneedles on conductive textile for harvesting piezoelectric potential. *Chemical Physics Letters*, 612, 62-67.
- Mehta, D. D., Nazir, N. T., Trohman, R. G., & Volgman, A. S. (2015). Single-lead portable ECG devices: Perceptions and clinical accuracy compared to conventional cardiac monitoring. *Journal of electrocardiology*, 48(4), 710-716.
- Sun, F., Yi, C., Li, W., & Li, Y. (2017). A wearable H-shirt for exercise ECG monitoring and individual lactate threshold computing. *Computers in industry*, 92, 1-11.
- Guzik, P., & Malik, M. (2016). ECG by mobile technologies. *Journal of electrocardiology*, 49(6), 894-901.
- Jung, J., Lee, J., Lee, J., & Kim, Y. T. (2015). A smartphone-based U-Healthcare system for real-time monitoring of acute myocardial infarction. *International Journal of Communication Systems*, 28(18), 2311-2325.
- Baig, M. M., Gholamhosseini, H., & Connolly, M. J. (2013). A comprehensive survey of wearable and wireless ECG monitoring systems for older adults. *Medical & biological engineering & computing*, 51, 485-495.
- Spanò, E., Di Pascoli, S., & Iannaccone, G. (2016). Low-power wearable ECG monitoring system for multiple-patient remote monitoring. *IEEE sensors journal*,

16(13), 5452-5462.

- Ozkan, H., Ozhan, O., Karadana, Y., Gulcu, M., Macit, S., & Husain, F. (2019). A portable wearable tele-ECG monitoring system. *IEEE Transactions on Instrumentation and Measurement*, 69(1), 173-182.
- Dias, D., & Paulo Silva Cunha, J. (2018). Wearable health devices vital sign monitoring, systems and technologies. *Sensors*, 18(8), 2414.
- Patel, S., Park, H., Bonato, P., Chan, L., & Rodgers, M. (2012). A review of wearable sensors and systems with application in rehabilitation. *Journal of neuroengineering and rehabilitation*, 9, 1-17.
- Van Loon, K., van Zaane, B., Bosch, E. J., Kalkman, C. J., & Peelen, L. M. (2015). Non-invasive continuous respiratory monitoring on general hospital wards: a systematic review. *PLoS One*, 10(12), e0144626.
- Beck, S., Laufer, B., Krueger-Ziolek, S., & Moeller, K. (2020). Measurement of respiratory rate with inertial measurement units. *Current Directions in Biomedical Engineering*, 6(3), 237-240.
- Di Tocco, J., Raiano, L., Sabbadini, R., Massaroni, C., Formica, D., & Schena, E. (2021). A wearable system with embedded conductive textiles and an imu for unobtrusive cardio-respiratory monitoring. *Sensors*, 21(9), 3018.
- Guo, L., Peterson, J., Qureshi, W., Kalantar Mehrjerdi, A., Skrifvars, M., & Berglin, L. (2011). Knitted wearable stretch sensor for breathing monitoring application. In *Ambience'11*, Borås, Sweden, 2011.
- Casale, P., Pujol, O., & Radeva, P. (2011). Human activity recognition from accelerometer data using a wearable device. In *Pattern Recognition and Image Analysis: 5th Iberian Conference, IbPRIA 2011, Las Palmas de Gran Canaria, Spain, June 8-10, 2011. Proceedings* 5 (pp. 289-296). Springer Berlin
- Nweke, H. F., Teh, Y. W., Mujtaba, G., Alo, U. R., & Al-garadi, M. A. (2019). Multi-sensor fusion based on multiple classifier systems for human activity

- identification. Human-centric Computing and Information Sciences, 9, 1-44.
- Long, X., Yin, B., & Aarts, R. M. (2009, September). Single-accelerometer-based daily physical activity classification. In 2009 Annual International Conference of the IEEE Engineering in Medicine and Biology Society (pp. 6107-6110). IEEE.
- Yang, J. Y., Chen, Y. P., Lee, G. Y., Liou, S. N., & Wang, J. S. (2007). Activity recognition using one triaxial accelerometer: A neuro-fuzzy classifier with feature reduction. In Entertainment Computing—ICEC 2007: 6th International Conference, Shanghai, China, September 15-17, 2007. Proceedings (pp. 395-400). Springer Berlin Heidelberg.
- ÇALIŞAN, M., & Talu, M. F. (2020). Comparison of methods for determining activity from physical movements. Politeknik Dergisi, 24(1), 17-23.
- Yen, C. T., Liao, J. X., & Huang, Y. K. (2020). Human daily activity recognition performed using wearable inertial sensors combined with deep learning algorithms. Ieee Access, 8, 174105-174114.
- Arif, M., & Kattan, A. (2015). Physical activities monitoring using wearable acceleration sensors attached to the body. PloS one, 10(7), e0130851.
- Bayat, A., Pomplun, M., & Tran, D. A. (2014). A study on human activity recognition using accelerometer data from smartphones. Procedia Computer Science, 34, 450-457.
- Devices, A. (2012). Analog. Single-Lead, Heart Rate Monitor Front End Data Sheet AD8232http://www.analog.com/media/en/technical-documentation/data-sheets/AD8232.
- Fasate, A. (2023, October 27). “Mastering Motion Sensing: A Comprehensive Guide to the MPU6050 and MPU9250.” Medium.https://medium.com/@fasateaniket5/motion-sensors-mpu-6050-and-mpu9250
- Potter, L. (2023, November 24). Understanding an ECG | ECG Interpretation | Geeky

- Medics. Geeky Medics. <https://geekymedics.com/understanding-an-ecg/>
- Ashley, E. A., & Niebauer, J. (2004). Conquering the ECG-cardiology explained-NCBI bookshelf. Cardiology Explained.
- Manullang, M. C. T., Simanjuntak, J., & Ramdani, A. L. (2019). Implementation of AD8232 ECG Signal Classification Using Peak Detection Method For Determining RST Point. Indonesian Journal of Artificial Intelligence and Data Mining, 2(2), 61-66.
- Bluetooth 5.2 radio system-on-chip (SOC) - onsemi. (n.d.). <https://www.onsemi.com/download/data-sheet/pdf/rsl10-d.pdf>