New University of Lisbon

Faculty of Sciences and Technology

Physics Department

PLUX, Wireless Biosignals

Algorithms for Time Series Clustering Applied to Biomedical Signals

Neuza Filipa Martins Nunes

Lisbon, 2011

Algorithms for Time Series Clustering Applied to Biomedical Signals

Advisor: Prof. Dr. Hugo Gamboa

Thesis submitted in the fulfillment of the requirements for the Degree of Master in Biomedical Engineering.

Physics Department

Faculty of Sciences and Technology,

New University of Lisbon

March 2011

Acknowledgments

At the end of this chapter in my life, many people deserve my sincere gratitude for all the emotional or scientific support that allowed me to achieve my goals and dreams.

My biggest thanks goes to Professor Hugo Gamboa for the excellent guidance provided and all the support given through this project. Thanks for giving me the opportunity to work with such amazing people and for helping me to grow and evolve so much in this short period of time.

During the last months I worked with many groups of enthusiastic people that accepted me and always had a word of appreciation for my work. I acknowledge Professors Filomena Carnide, Vera Pereira, Ana Conceição, Sandra Amado, Bjorn Olstad and Havard Myklebust. I really enjoyed working with all of you.

I thank *PLUX - Wireless Biosignals* and all its collaborators for making me feel part of the team. A special thanks to José Medeiros, Rui Martins, Susana Palma, Joana Sousa, Lídia Fortunato, Nuno Santos, Paulo Aires, Gonçalo Martins and Nuno Cardoso for all the help and for making me laugh every day! I also thank Tiago Araújo for the great companionship developed during these months and his enthusiastic approach every time we worked together.

My friends, that accompanied me for the last years, also have my gratitude. Inês, Mafalda, Claudia and Sofia, you made these years great. Ricardo Gomes, thanks for always finding the time to ask how I was doing and offer your help. *Will*, *Dred* and *Misai*, thank you for the nice motivation words. *Rodri*, I can't thank you enough!

Agradeço à minha família pelo apoio recebido e orgulho demonstrado, especialmente à minha avó por ter lutado para que eu conseguisse ter um futuro melhor.

And last I thank Tiago Reis. Thank you for being the person who supported me the most through the last years and for your unconditional belief in me.

Abstract

The increasing number of biomedical systems and applications for human body understanding creates a need for information extraction tools to use in biosignals. It's important to comprehend the changes in the biosignal's morphology over time, as they often contain critical information on the condition of the subject or the status of the experiment. The creation of tools that automatically analyze and extract relevant attributes from biosignals, providing important information to the user, has a significant value in the biosignal's processing field.

The present dissertation introduces new algorithms for time series clustering, where we are able to separate and organize unlabeled data into different groups whose signals are similar to each other.

Signal processing algorithms were developed for the detection of a *meanwave*, which represents the signal's morphology and behavior. The algorithm designed computes the *meanwave* by separating and averaging all cycles of a cyclic continuous signal. To increase the quality of information given by the *meanwave*, a set of wave-alignment techniques was also developed and its relevance was evaluated in a real database. To evaluate our algorithm's applicability in time series clustering, a distance metric created with the information of the automatic *meanwave* was designed and its measurements were given as input to a K-Means clustering algorithm. With that purpose, we collected a series of data with two different modes in it. The produced algorithm successfully separates two modes in the collected data with 99.3% of efficiency. The results of this clustering procedure were compared to a mechanism widely used in this area, which models the data and uses the distance between its cepstral coefficients to measure the similarity between the time series.

The algorithms were also validated in different study projects. These projects show the variety of contexts in which our algorithms have high applicability and are suitable answers to overcome the problems of exhaustive signal analysis and expert intervention.

The algorithms produced are signal-independent, and therefore can be applied to any type of signal providing it is a cyclic signal. The fact that this approach doesn't require any prior information and the preliminary good performance make these algorithms powerful tools for biosignals analysis and classification.

Keywords: Biosignals, Algorithms, Signal-Processing, Alignment Techniques, Clustering.

Resumo

O aumento de sistemas e aplicações biomédicas para o conhecimento do corpo humano cria a necessidade de ferramentas para extracção de informação relevante de biosinais. É importante perceber as alterações da morfologia de um biosinal ao longo do tempo, pois elas frequentemente contêm informação crítica sobre o estado de um sujeito ou de um estudo. A criação de ferramentas que automaticamente analisam e extraem atributos relevantes dos biosinais, providenciando informação importante ao utilizador, tem um valor significativo na área do processamento de biosinais.

A presente dissertação introduz novos algoritmos para *clustering* em séries temporais, em que conseguimos separar e organizar dados não classificados em diferentes grupos cujos sinais são semelhantes entre si.

Algoritmos de processamento de sinal foram desenvolvidos para a detecção automática de uma onda média, representativa da morfologia e comportamento do sinal. O algoritmo projectado cria uma onda média através da separação e cálculo da média ponto-a-ponto de todos os ciclos de um sinal cíclico contínuo. Um conjunto de técnicas de alinhamento de ondas foi desenvolvido para aumentar a qualidade de informação dada pela onda média e a sua relevância foi avaliada numa base de dados real. Para avaliar a aplicabilidade do nosso algoritmo em de séries temporais de dados, escolhemos uma medida de distância criada com a informação da onda média automática, e essas medidas foram introduzidas num algoritmo de *clustering* K-Means. Com esse propósito, adquirimos uma série de biosinais com dois modos distintos. O algoritmo produzido separa com sucesso os dois modos destes sinais com 99.3% de eficiência. Os resultados deste procedimento foram comparados com um mecanismo muito usado nesta área, que modela os dados e usa a distância entre os seus coeficientes cepstrais

para medir a similaridade entre as séries temporais.

Também avaliámos a aplicabilidade dos algoritmos desenvolvidos em diferentes casos de estudo concretos. Estes projectos mostram a variedade de contextos em que os nossos algoritmos têm grande aplicabilidade e são soluções adequáveis para ultrapassar os problemas da análise exaustiva de sinais ou intervenção do perito.

Os algoritmos produzidos podem ser aplicados a qualquer tipo de sinal, desde que este seja cíclico. O facto deste método não requerer prévia informação e o seu bom desempenho fazem com que esta seja uma ferramenta poderosa para a análise e classificação de biosinais.

Palavras-chave: Biosinais, Algoritmos, Processamento de Sinal, Técnicas de Alinhamento, *Clustering*.

Contents

List of Figures

List of Tables

List of Abbreviations and Units

AAL Ambient Assisted Living

ACC Accelerometry

ADC Analog to Digital Converter

AR Autoregressive

ARMA Autoregressive Moving Average

BVP Blood Volume Pressure

BB *Biceps Brachii*

CT Cycle Time

ECG Electrocardiography

EEG Electroencephalography

EMG Electromyography

EP Evoked Potentials

f_0 Fundamental Frequency

f_s Sampling Frequency

FFT Fast Fourier Transform

IR Infrared

LED Light-emitting Diode

LPC Linear Predictive Coding

MA Moving Average

OLR Overlap Left and Right side

NaN Not a Number

PT Pushing Time

PPG Photopletysmography

RT Recovery Time

sEMG Surface Electromyography

std Standard Deviation Error

SYM Symmetry between each side

SNR Signal-to-Noise Ratio

TB *Triceps Brachii*

XC Cross Country

Chapter 1

Introduction

1.1 Motivation

The constant chase for human wellbeing has led medicine and engineering to increasingly develop new systems and applications for a continuous monitoring of patients through their body signals. These techniques generate large amounts of data, creating the need for information extracting tools. For a more rigorous and precise medical care, the intrinsic dynamic character of the analyzed physiological parameters should be taken into account [78]. Medical imaging and biomedical signal analysis are becoming methods of the utmost importance for visualization and interpretation in biology and medicine, as the manipulation and processing of data provide the researcher with vital information on the condition of the subject or the status of the experiment. An awareness of the power of computer-aided techniques, coupled with a continuing need to derive more information from biomedical signals, has led to a growing application of signal processing techniques in medicine, sports and research [97].

Signal processing techniques have been developed in order to help the examination of many different biosignals and to find new information embedded in them not easily observable in the raw data. When watching a biosignal's evolution it's important to know and understand the changes in its morphology over time. Abrupt changes often contain critically important information from various perspectives, and hence, the problem of discovering time points where changes occur has received much attention on statistics and data mining [9] [38].

The creation of tools that automatically analyze and extract relevant attributes

from biosignals, providing important information to the user, represents a significant step in the biosignal's processing field.

This dissertation was developed in *PLUX - Wireless Biosignals* [73]. One of the main goals of its Research and Development (R&D) department is the creation of new solutions for more comfortable and ergonomic biosignals monitoring. Having the opportunity to create new tools that can help to extract and analyze information from those signals strongly encouraged this research work. Also, researching in a business environment led to interactions with other institutions and developments of interesting and exciting projects. The resulting scientific publications of some researches and the eagerness of the institutions to use the resulting work on a regular basis instilled great motivation for the development of this study.

1.2 State of the Art

Biosignal processing and classification tools have been studied and developed in several research projects and are in constant adaptation to the new advanced acquisition systems. Following, we list a few state of the art examples that represent major advances in this area.

Human activity tracking techniques focus on observation of people and their behavior; in the past, such examination was mostly done with a great amount of cameras [11]. However, the use of wearable sensors have been increasingly sought because it allows continuous acquisitions in different locations, being independent from the infrastructures. Activity recognition with wearable sensors has a vast applicability, particularly in sports and healthcare. In the sports field, there is a need for wearable sensors to assess physiological signals and body kinematics during free exercise. The inclusion of textile-based sensors in sports is already a practice [26]. Wearable sensors have major utility in healthcare, particularly for monitoring elderly and chronically ill patients in their homes, through Ambient Assisted Living (AAL) (figure 1.1) [76].

Frameworks using wearable sensors are usually based on accelerometers, electrocardiography (ECG), blood volume pressure (BVP) or electromyography (EMG) sensors [7] [90] [31] [76]. This provides an effective means of monitoring the physiological and activity status of the subject. The recognition of the daily life activity is a sig-

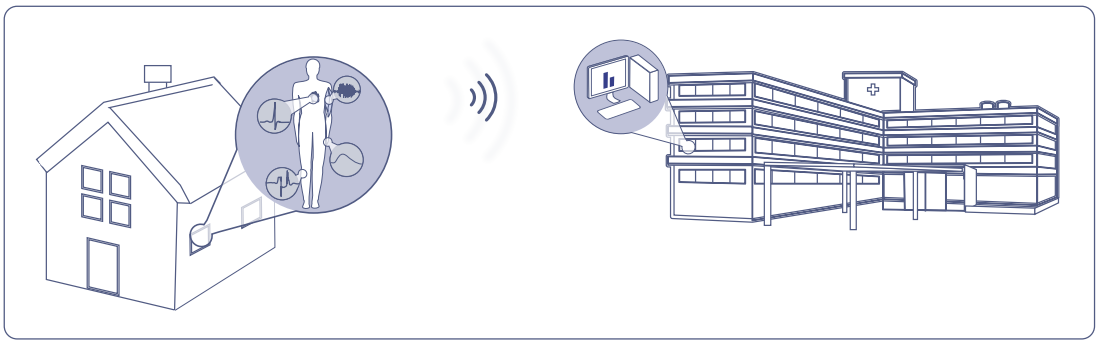

Figure 1.1: A representation of AAL: a person at home instrumented with wearable sensors and his/hers vital signs being transmitted to the Hospital.

nificant challenge in human caring systems, especially in elderly care. Recently, some researchers have investigated these issues with the use of accelerometers and signal processing algorithms to classify human activity [95]. The introduction of wearable sensors covers a wide range of topics, such as the recognition of activities of daily living in the context of healthcare and elderly care, unsupervised learning of activities or the combination of several sensor modalities to improve recognition performance [44]. Wearable solutions are being required for health and medical treatment applications; the goal is the reduction of costs on Health Services while maintaining a high quality of the supplied services, thus providing an easy access from various possible sites, at any time, and towards a large number of users. These devices possess the capability of extending health services, support the traditional applications, caring of disaster victims and monitor emergency operators [17].

Accelerometers embedded in mobile phones are also beginning to be used to recognize the user's activity in real time and promote more physical activity and a healthier lifestyle. Real time applications in mobile phones with accelerometers are used to estimate and summarize a person's activity levels in order to motivate and encourage a reflection on the daily activities [42], combine activity and location to suggest spontaneous exercises [14] and even for user-interaction by sharing the information on the presence and activity of the individual on social networks [61]. Unlike the approach we will present, the majority of these systems uses a supervised learning classification to differentiate the users activity, by extracting some features from the data (such as the mean, energy, frequency-domain entropy and correlation) [28].

Another interesting and urging research is the study of human-centered cognitive systems to improve human-machine interaction as well as machine-mediated human

communication. At Cognitive Systems Lab, in Germany, studies have been focused on biosignals acquired from the human being and its interpretation, classification and measurement by machines. This group is working in interfaces for silent speech recognition, that rely on articulatory face muscle movements [48], interfaces that use brain activity to determine the users' mental states (task activity, cognitive workload and emotion) [87] [27] and interfaces for airwritting recognition that identify text writing from free movements of the hand [3].

Dr. Rodrigo Quiroga [80] also presents an interesting research on clustering neuron activity. The clustering procedure, like the one we will present, is based on the similarity of the waveshapes in a single time series data, given the principle that each neuron tends to fire spikes of a particular shape. However, as those spikes do not occur periodically, unlike our approach, the procedure is based in threshold methods to extract relevant attributes of each wave.

Other recent clustering and pattern recognition techniques include fatigue detection in athletes' EMG signals [30] and detection of cardiac arrhythmia in ECG signals [91].

The approach described in this thesis contributes for the biosignal processing and clustering/classification field, presenting a new way to cluster varied types of biosignals in a wide range of different scenarios.

1.3 Objectives

The main goal of the thesis is to cluster time series signals. To achieve that aim we needed signals, tools to process and extract information from those signals and clustering algorithms to separate them. We detained our attention on time varying signals collected from the human being, as EMG, ECG, BVP, accelerometry (ACC) or motion signals obtained from reflective markers and infrared cameras. The physiological manifestations and morphology of the biosignals were observed and robust signal processing algorithms were designed and implemented. Alignment techniques and distance metrics were defined to apply to the signals and to use the clustering algorithms. It was also a goal to test and validate each of these tools separately - which resulted in some independent contributions that will be discussed in this thesis (figure 1.2).

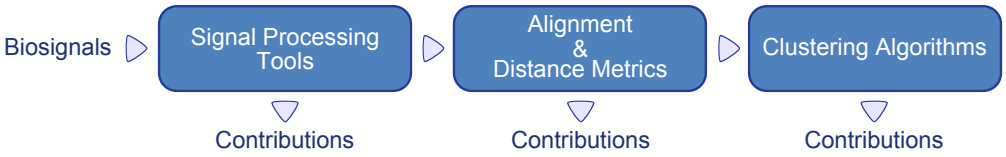

Figure 1.2: Independent tools developed for the thesis.

1.4 Thesis Overview

This thesis is divided into four parts composed with six chapters and one appendix, as schematized in figure 1.3.

In the present chapter, the thesis context is exposed, providing some insight on the motivation and objectives which led to the development of this work and a state of the art is also presented. A review of the theoretical concepts is made in the second chapter. These two chapters form the basis for the development of the thesis.

Chapter 3 focuses on the signal processing algorithms developed for this study. We developed an algorithm which computes a *meanwave* from a continuous cyclic signal and also wave-alignment techniques as a complement to the *meanwave*. The processing steps of both procedures will be described in chapter 3. This chapter alone molds the methods part of the thesis.

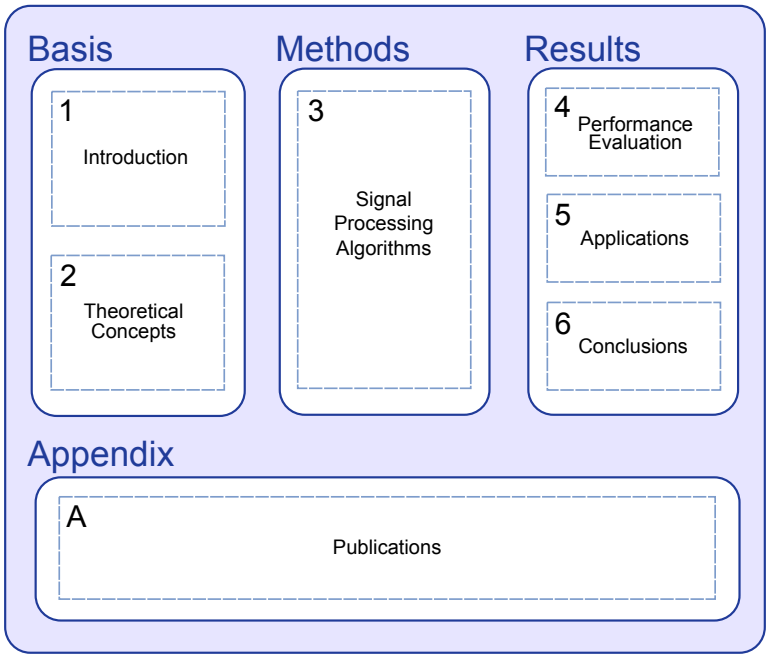

Figure 1.3: Thesis Overview

Chapter 4 exposes three methods for an evaluation performance of our work. These evaluation projects test the direct performance of our algorithms, validating them in real datasets and testing its efficiency to cluster time series data. Chapter 5 presents some projects which were developed in parallel with our algorithms, working both as motivation and possible applications for them. In this chapter we compare the methods created specifically for those projects with our approaches, evidencing its efficiency and value. Although both of these chapters present clear applications of our algorithms, we choose to present them separately to emphasize its individual importance. The last chapter of the thesis presents the conclusion of the work, with a view over the general results and contributions achieved and some suggestions of future work. These three chapters constitute the results part of the thesis.

The thesis has one additional appendix. Appendix A presents the three articles published during this research.

The implemented algorithms were developed using *Python* and *Eclipse* as an integrated development environment [98]. The *Python* packages used were the numpy [66], scipy [65], matplotlib [43] and scikits talkbox [25].

Chapter 2

Concepts

This chapter reviews the theoretical concepts relevant for the present research. In this chapter, contextual information about biosignals, biosignal processing and clustering mechanisms will be provided.

2.1 Biosignals

The term biosignal is used for all kinds of signals which can be continuously measured from biological beings. The biosignals represent space-time records that capture some aspect of a biological event and are usually divided into the following groups [97] [20]:

- **Bioelectrical signals:** Represented by changes in electric currents produced by the sum of electrical potential differences across a specialized tissue, organ or cell system.
- **Biomechanical signals:** Generated by tissue motion that produces force.
- **Biomagnetic signals:** When electrically active tissue produces a bioelectric field, it simultaneously produces a biomagnetic field as well.
- **Biochemical signals:** Represents the levels and changes of various biochemicals like glucose and metabolites.

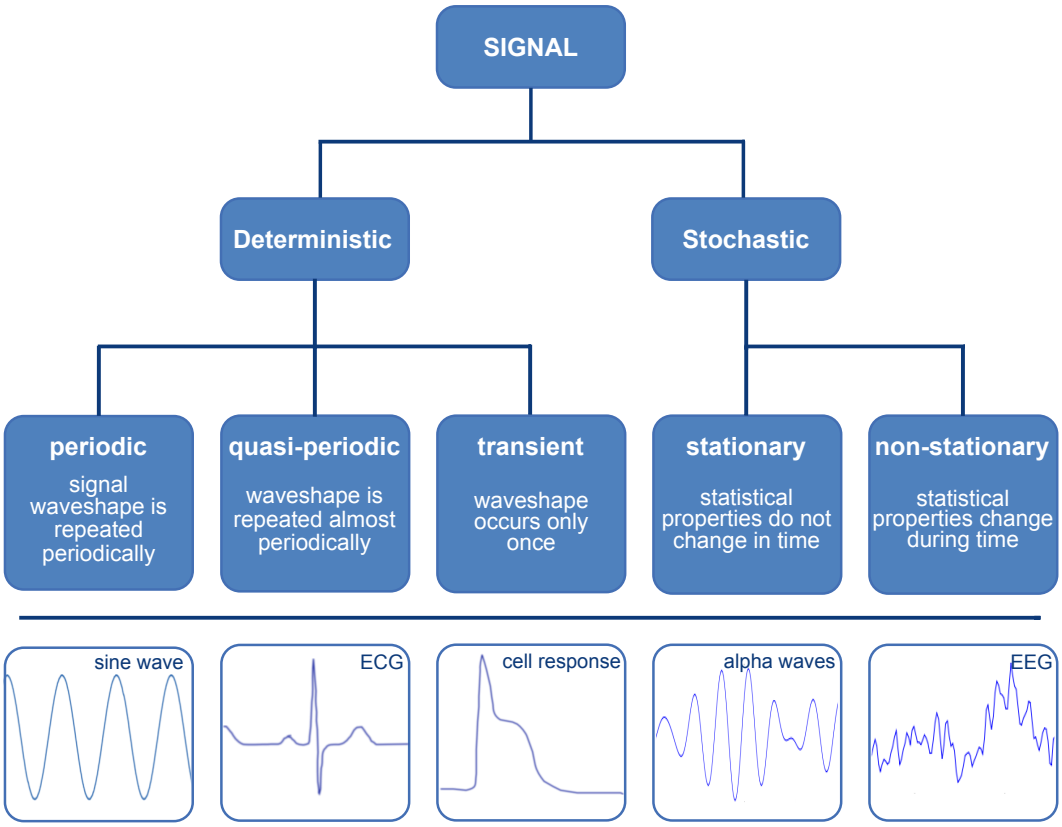

Figure 2.1: Biosignal's Types and waveshapes examples. From [45].

These biosignals can be either deterministic or stochastic in nature, as shown by some examples in figure 2.1. The deterministic group includes the periodic, quasi-periodic, and transient signals. The stochastic signals are subdivided into stationary and non-stationary signals [10]. This research focuses on quasi-periodic biosignals and transient biosignals periodically repeated. In this dissertation the term "time series" is also used, referring to the biosignals to process. A time series is a sequence of data points, measured at successive times and spaced with uniform time intervals.

2.1.1 Biosignals Types

Probably the most familiar electrical measures of biosignals are the electrocardiography (ECG) and electroencephalography (EEG), which are the recordings of the electrical activity of the heart and the brain, respectively. These are examples of continuous biosignals, but there are also biosignals with values only defined at some discrete non-uniform times, for example, the successive heart beat intervals (heart rate variability) [96].

The signals most used in this research were the electrocardiography, blood volume

pressure (BVP), electromyography (EMG), accelerometry (ACC) and movement, which are briefly described next.

Electrocardiography

The ECG signal is the heart's electric activity recording, describing the repolarization and depolarization of the atrial and ventricular chambers of the heart. Depolarization is the sudden influx of cations when the membrane becomes permeable, and repolarization is the recovery phase of the ion concentrations returning to normality. The ECG works mostly by detecting and amplifying the tiny electrical changes on the skin, with surface electrodes, that are caused when the heart muscle "depolarizes" during each heart beat. A normal ECG pattern consists of a P wave, a QRS complex, and a T wave. The P wave depends on electrical currents generated when the atria depolarize before contraction, and the QRS complex is produced by currents arising when the ventricles depolarize prior to contract. Therefore, the P wave and the components of the QRS complex correspond to depolarization waves. The T wave, which is caused by currents arising when the ventricles recover from the depolarization state, is known as the repolarization wave. Figure 2.2 illustrates these concepts.

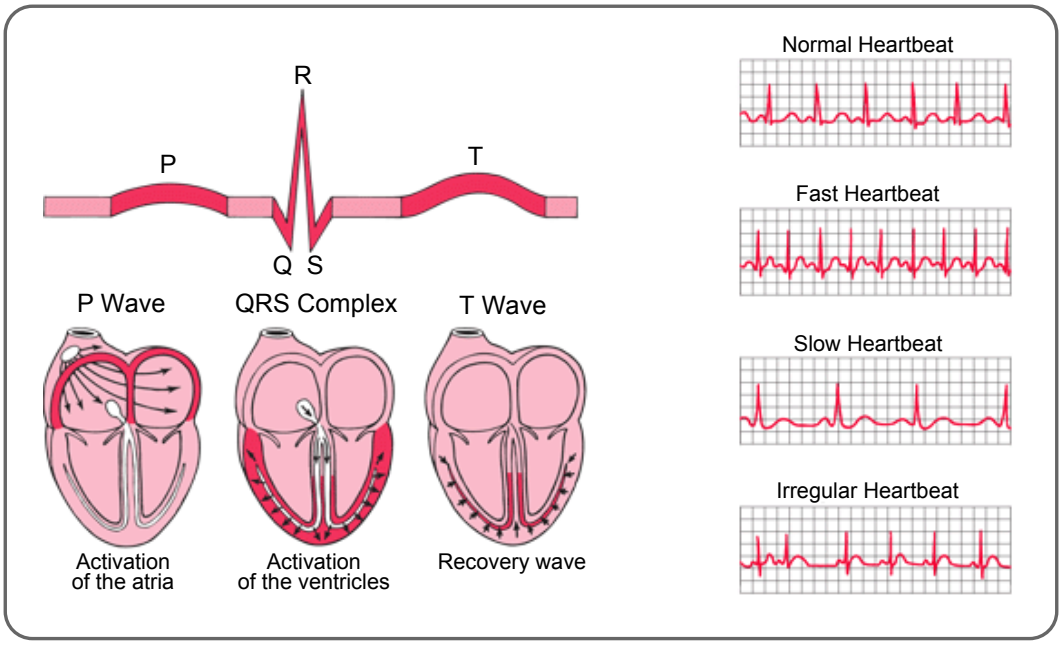

Figure 2.2: The origin of the electrical activity of the heart and pattern of a normal, fast, slow, and irregular ECG signal. From [91].

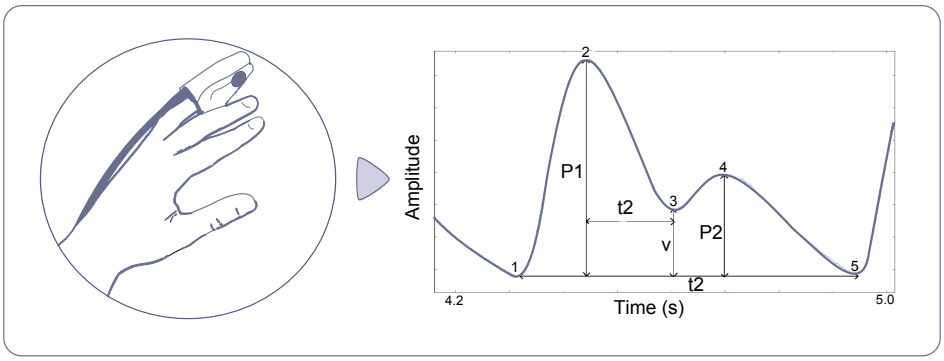

Figure 2.3: Example of a BVP finger sensor and a BVP signal with amplitude and timing markers. Time t1 (between markers 1 and 5) indicates the inter-beat interval and is used to compute the heart rate. P1 and P2 (marker 1 and 4) are measures of pressure peak amplitudes. Volume at V (marker 3) is the indicator of the blood volume influenced by the end of a systole. From [41].

Blood Volume Pressure

The pressure exerted by blood circulating upon the walls of blood vessels, constitute one of the principal vital signs, the blood volume pressure. During each heartbeat, the blood pressure varies between a maximum (systolic) and a minimum (diastolic) pressure. BVP sensors can be used to detect heart beats, based on a principle called photopletysmography (PPG) which measures changes in blood volume in arteries and capillaries by shining an infrared light - a light-emitting diode (LED) - through the tissues. At each contraction of the heart, blood is forced through the peripheral vessels, producing engorgement of the vessels under the light source. The amount of light that returns to a PPG sensor's photodetector is proportional to the volume of blood in the tissue. The PPG signal represents an average of all blood volume in the arteries, capillaries and any other tissue through which the light passed, and depends on the thickness and composition of the tissue beneath the sensor and the position of the source and receiver of the infrared light [79] [58]. An example of a BVP signal acquired with this PPG method is presented in figure 2.3.

Electromyography

The electromyography (EMG) signal is a record of the electrical activity generated by muscle cells when these cells are electrically or neurologically activated. The signals can be analyzed to detect clinical abnormalities, activation level, recruitment order or to analyze the movement biomechanics. From EMG it is possible to determine whether

a particular muscle is responding appropriately to stimulation and whether a muscle remains inactive when not stimulated. Muscles are stimulated by signals from nerve cells called motor neurons. A motor unit is defined as one motor neuron and all of the muscle fibers it innervates. When a motor unit fires, the impulse (called action potential) goes through the motor neuron and to the muscle. The electrophysiologic activity from multiple motor units is the signal typically evaluated during an EMG. There are two types of EMG: intramuscular and surface EMG (sEMG). Intramuscular EMG is performed by inserting a needle into the muscle, serving as an electrode. The sEMG is obtained by a non-invasive method, placing an electrode on the skin over a muscle in order to detect the electrical activity of this muscle [59]. In this research, all EMG signals referenced used a surface and non-invasive acquisition method. Figure 2.4 makes a brief representation of the explained concepts and sEMG signal.

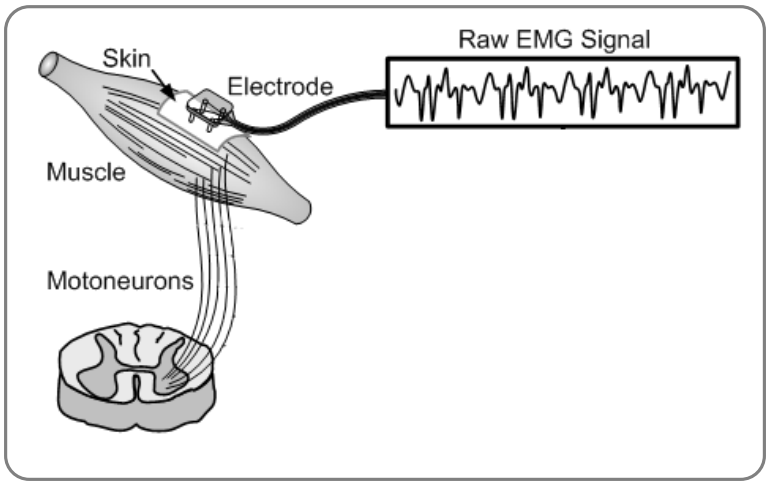

Figure 2.4: Representation of a sEMG sensor placed on the surface of the skin above the muscle and an example of the collected signal. From [100].

Accelerometry

The accelerometry is a method for movement kinematic analysis which allows, through the use of an accelerometer sensor, the quantification of caused or suffered accelerations by the human body. The accelerometer sensor detects the acceleration magnitude or direction change rate. As the majority of human motion occurs in more than one axis, triaxial accelerometers are used to measure the acceleration in each orthogonal axis [84]. In this dissertation, all accelerometry signals were acquired from triaxial accerelometers - a representation of the acceleration signals from each axis is shown in

figure 2.5. Accelerometers are widely used, for example, for vibration and tilt analysis and to obtain motion patterns of various motion tasks. Gravity is the acceleration toward the center of the Earth, and accelerometers that have a DC response are sensitive to gravity. An accelerometer at rest with its sensitive axis pointing toward the center of the Earth will have an output equal to 1 g. This property is commonly used to calibrate the gain and zero offset of an accelerometer [1].

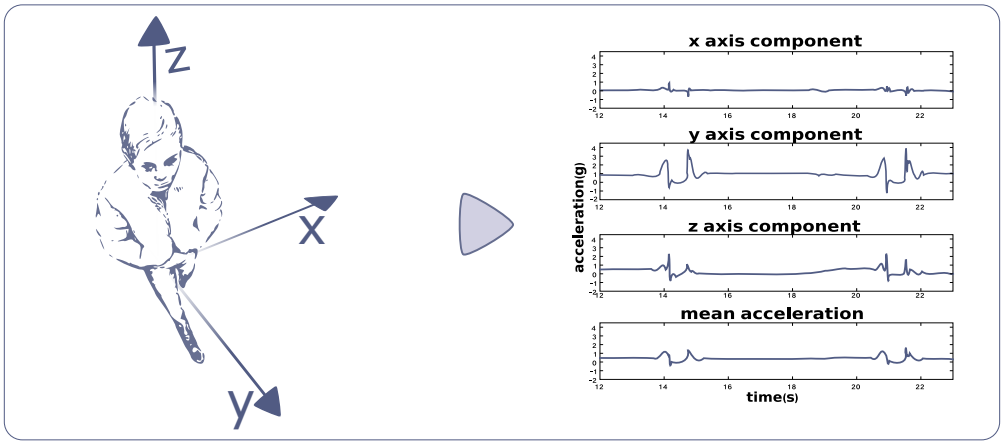

Figure 2.5: Representation of the orthogonal axis and resulting acceleration signals for the three axis components. The mean acceleration curve can be computed from the x, y and z acceleration components.

Movement

Optoelectronic systems are widely used to capture movement over time. Those systems use video cameras to track the motion of markers attached to particular locations of a subject's body. The operating principle of the optical system is based on the detection of light (visible or not) emanated from the surface of the moving markers into the surrounding space. Such radiation may be originated either by an external light source that then is retro-reflected by the markers or produced by the markers itself. The former case is classified as a passive detector and the later as an active detector. The active detectors rather than reflecting light back that is generated externally have markers which are themselves powered to emit their own light; the optical motion capture systems based on this pulsed-LED's markers measure the IR light emitted by the LEDs. The reflective optical motion capture systems use infrared (IR) LEDs mounted around the camera lens, along with IR pass filters placed over the camera lens and measure the light reflected from the markers. Specific computer software

detects and triangulates the position of each visible marker relative to the camera position, tracing movement signals with the changes of the markers positions [99]. An illustration of this motion signals' acquisition method is presented in figure 2.6.

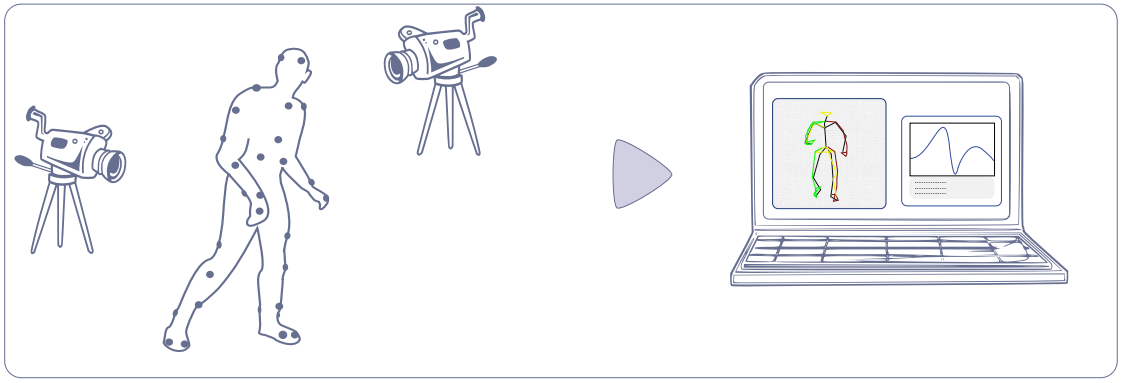

Figure 2.6: Acquisition of motion signals by reflective markers and infrared cameras and a representation of the software for data analysis.

2.1.2 Biosignals Acquisition

The naturally occurring biosignals are analog signals, i.e., signals that present infinite values between a defined interval, varying continuously in time and in amplitude. A digital signal, in opposition, is a discrete and quantized signal, with a finite number of values in time and amplitude. Computers store and process values in discrete units, therefore, before processing is possible, the analogue signals must be converted to discrete units and sampled. The conversion process is called analogue-to-digital conversion (ADC). ADC comprises sampling and quantization - the continuous value is observed (sampled) at fixed intervals and rounded (quantized) to the nearest discrete unit, as observable in figure 2.7. Each time value the discrete signal assumes is called "sample" [88].

In this conversion, there are two parameters that determine how closely the digital data resembles the original analogue signal: the precision and the frequency with which the signal is recorded and sampled [60]. Precision describes the accuracy degree of a sample observation of a signal. It is derived from the number of bits (quantization) used to represent a signal - the higher number of bits, the greater is the number of levels that can be distinguished. The sampling frequency is another parameter that affects the correspondence between an analogue signal and its digital representation. The sampling frequency defines the number of samples per second (or per other unit)

taken from an analog signal to make a discrete signal. A sampling rate that is too low relative to the rate at which a signal changes value will produce a poor representation. On the other hand, oversampling increases the expense of processing and storing the data. The sampling theorem mathematically expressed by Nyquist states that a signal must be sampled at a rate at least twice the rate of the highest frequency component present in the signal. If a signal is sampled at a higher frequency than the Nyquist frequency, the complete syntactic information content of the signal is retained [82]. Figure 2.8 illustrates three cases of a signal's sampling with different discrete signal results.

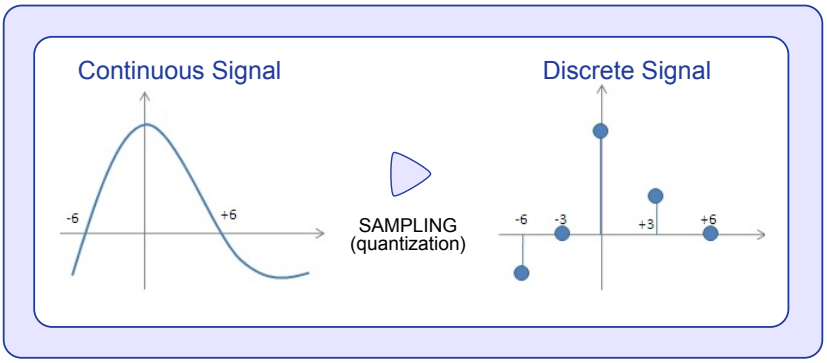

Figure 2.7: Sampling and quantization of an analog signal. From [70].

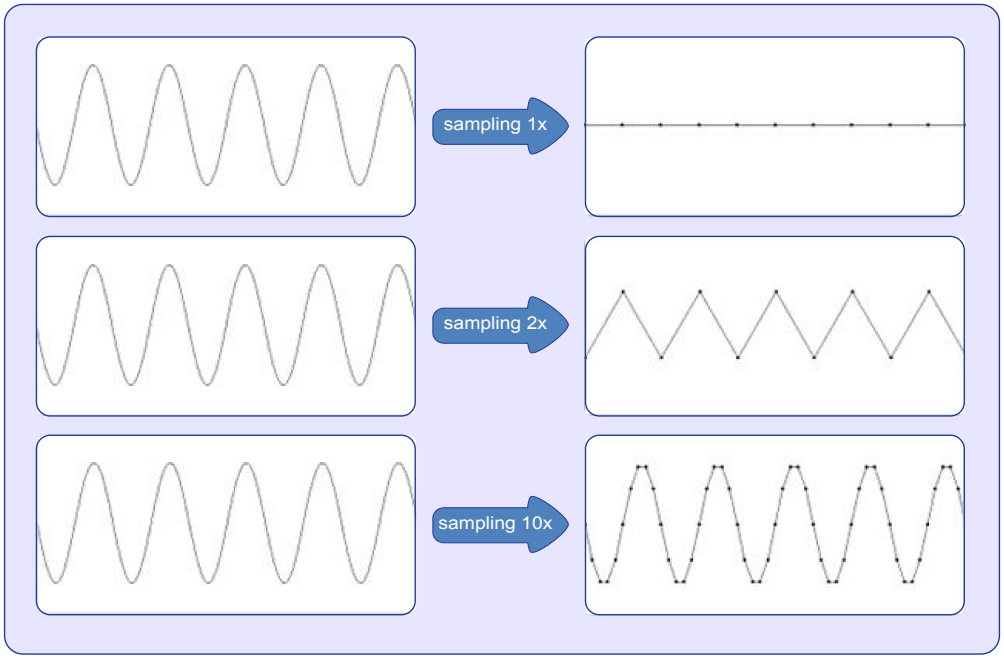

Figure 2.8: Examples of sampling at 1x, 2x, and 10x of signal's maximum frequency. Notice that 2x only shows the frequency information of the desired signal, not amplitude or shape. From [69].

2.1.3 Biosignals Processing

Biomedical signal processing aims at the representation, transformation and manipulation of biosignals and extraction of significant information. With the aid of biomedical signal processing, sophisticated medical monitoring systems that give a real time view over the human body's functioning can help physicians to monitor distinct illnesses.

The primary focus of biomedical signal processing was on filtering signals to remove noise [21] [35] [54]. While these denoising techniques are well established, the field of biomedical signal processing continues to expand with the development of various novel biomedical instruments.

The processing of biomedical signals usually consists of at least four stages [45]:

- Signal observation and measurement or signal acquisition;
- Transformation and reduction of the signal;
- Computation and extraction of parameters that are significant for signal interpretation;
- Signal interpretation or/and classification.

Typically after the acquisition, filtering and needed transformations of the data, there's a necessity to lower the data volume and to obtain a more abstract information view. Often, the data are analyzed to extract important signal's parameters, and then are interpreted and classified. In clinics, classification is used to distinguish a normal behavior from one with some pathology or abnormality. For instance, by monitoring physiological recordings, clinicians judge if patients suffer from illness [40]; cardiologists detect which region of the myocardium experiences failure by watching cardiac magnetic resonance scans [19]; and geneticists infer the likelihood that the children inherit disease from their parents, analyzing gene sequences of a family [15].

The field of signal processing has always benefited from a close coupling between theory, applications and technologies for implementing signal processing systems. The growing number of applications and demand for increasingly sophisticated algorithms goes hand-in-hand with the rapid pace of device technology for implementing signal processing systems. In many ways, the importance and role of signal processing is accelerating and expanding [74].

2.2 Clustering and Classification

Clustering is the process of finding structural information in data without labeling. Clustering mechanisms separates and organizes unlabeled data into different groups whose members are similar to each other in some metric. Those groups are called clusters. Being a method of unsupervised learning, the learner only receives unlabeled inputs with no information on class of each sample. A good clustering procedure will produce clusters in which the intra-class similarity is high and the inter-class similarity is low. The clustering quality depends on both the similarity measure used by the method and its implementation [34].

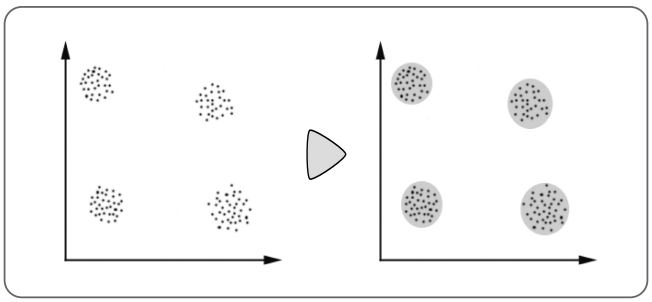

Figure 2.9: Graphical example of data clustering. From [68].

In figure 2.9, the four clusters into which the data can be divided are easily identified. In this particular case, the similarity criterion for the data clustering is the Euclidean distance, so it is a *distance-based clustering*: objects belong to the same cluster if they are similar according to a given distance metric. Another type of clustering is the *conceptual clustering*: objects belong to the same cluster if it defines a concept common to all those objects. In other words, objects are grouped according to their fit to descriptive concepts, not according to simple similarity measures [89].

Classification is a technique used to predict group membership for data instances and is a supervised learning process, as opposed to the unsupervised learning clustering mechanisms. Unlike clustering, in which we do not know the class labels and may not know the number of classes, the classification process aims to identify unknown data, on the basis of a training set of data containing observations whose classes are known. Classification in biomedicine faces several difficulties: researchers spend a long time to accumulate enough knowledge to distinguish different related cases, as normal and abnormal. Manual classification requires intensive labor and is time consuming - signal characteristics may not be prominent and therefore not easily discernible by the

researcher. Automated methods for classification in signal processing hold the promise for overcoming some of these difficulties and to assist and improve biomedical decision making [18].

In this dissertation we focus our attention in distance-based clustering mechanisms applied to time series.

2.2.1 Clustering Phases

The procedure of cluster analysis can be divided into four phases [103] [104], illustrated in figure 2.10.

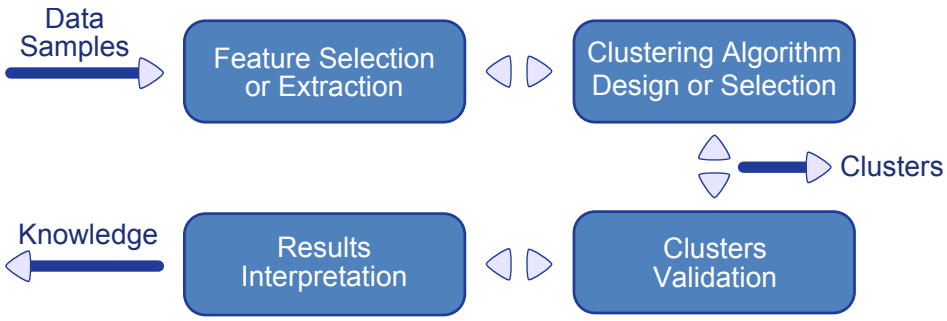

Figure 2.10: The phases of cluster analysis. From [103] and [104].

1. **Feature selection or extraction.** Feature selection chooses distinguishing features from a set of candidates and feature extraction uses data transformations to generate useful and novel features. Both are extremely crucial to the effectiveness of clustering applications. Elegant selection of features can greatly decrease the workload and simplify the subsequent design process. Ideal features should belong to different clusters, are immune to noise and are easy to extract and interpret.

2. **Clustering algorithm design or selection.** This step is usually combined with the selection of a corresponding proximity measure. Patterns are grouped according to whether they resemble each other and the proximity measure directly affects the formation of the resulting clusters. There is no clustering algorithm that can be universally used to solve all problems. Therefore, it is significant to carefully investigate the characteristics of the problem at hand, in order to use an appropriate strategy.

3. **Cluster validation.** Different approaches generally lead to different clusters and even for the same algorithm, parameter identification or the sequence of input patterns may affect the final results. Consequently, effective evaluation standards and criteria are important to provide the researcher with a degree of confidence for the results derived from the used algorithms. These assessments should be objective and have no preferences and should be useful for answering questions on how many clusters are hidden in the data, whether the clusters obtained are meaningful, or why we choose some algorithm as an alternative of another.

4. **Results interpretation.** The ultimate goal of clustering is to provide meaningful insights from the original data. Further analysis may be required to guarantee the reliability of extracted knowledge.

2.2.2 Clustering Methods

Time series clustering can be divided into two main categories [52]: "Whole Clustering" and "Subsequence Clustering". "Whole clustering" is the clustering performed on many individual time series to group similar series into clusters. "Subsequence clustering" is based on sliding window extractions of a single time series and aims to find similarity and differences among different time windows of a single time series.

The clustering algorithms are usually classified as Hierarchical or Partition-Based [105]. These categories will be briefly described below.

Hierarchical Algorithms

This clustering mechanism creates a hierarchical decomposition of the dataset using some criterion.

The hierarchical clustering algorithm groups data to form a tree shaped structure and find successive clusters by using ones previously established. It can be divided into agglomerative hierarchical clustering ("bottom up") and divisive hierarchical clustering ("top down"). In agglomerative approach, each data points are considered to be a separate cluster and on each iteration clusters are joined based on a criteria. In divisive approach all data points are considered as a single cluster and then they are divided into a number of clusters based on certain criteria [29].

Partition-Based Algorithms

In this method various partitions are constructed and then evaluated by some criterion.

The partition clustering algorithm splits the data points into k partitions, where each partition represents a cluster. The cluster must exhibit two properties: (1) each group must contain at least one object; (2) each object must belong to exactly one group. The main weakness of this algorithm is that whenever a point is close to another cluster's center, it gives weak results due to overlapping data points [29] [46].

In this dissertation, a partition based algorithm, the K-Means, is used. The main idea of the K-Means algorithm is to define a loop with k centroids (centers of the cluster) far away from each other, taking each point belonging to a given dataset and associate it to the nearest centroid. Repeating the loop, the centroids position will change because they are re-calculated by averaging all the points in the cluster, and after several iterations the position will stabilize, achieving the final clusters [57]. The main advantages of this algorithm are its simplicity, efficiency and speed which are good attributes for large datasets [101].

2.2.3 Distance-based Methods

As mentioned before, a distance based clustering is built upon the similarity between the data calculated with a chosen distance metric. Various distance measures can be applied to the data which will determine how the similarity of two elements is calculated. A simple Euclidean distance metric is sufficient to successfully group similar data instances. However, sometimes a chosen distance metric can be misleading, and therefore it's important to know which will suit best the input data - this will influence clustering results, as some elements may be close to one another according to one distance and farther away according to another. [68].

Several approaches for time series comparison have been proposed in literature. The most straightforward approach relies on similarity measures which directly compare observations or features extracted from raw data. In the time domain, the autocorrelation and cross-correlation functions are used to this end. Besides the measurements made directly between time series, distances can also be computed from transformations of the data [36]. In the frequency domain, techniques using discrete fourier transform of data and wavelets are also used [23].

Warren Liao [55] presents a survey on time series data clustering, exposing past researches on the subject. He organizes clustering in three groups: whether they work directly with the raw data, indirectly with features extracted or indirectly with models built from the raw data. By modeling the raw data with a stochastic model, similarities are detected in the dynamics of different time series.

Modeling

A model is a convenient way of summarizing the observations of a time series and increase understanding of the underlying process. Models for time series data can have many forms and represent different stochastic processes. Kashyap and Rao [51] state that the basic premise in modeling is that complicated systems can be expressed using simple models [49].

Linear Predictive Coding (LPC) is one of the methods of model compression and is widely used in speech analysis. Linear prediction filters attempt to predict future values of the input signal based on past signals. The model is created with p poles and q zeros in the general *pole-zero* case, which means that a synthetic sample, $\hat{s}(n)$, can be modeled by a linear combination of the p previous output samples and $q+1$ previous input samples of an LPC synthesizer:

$$\hat{s}(n) = \sum_{k=1}^{p} a_k \hat{s}(n-k) + G \sum_{l=0}^{q} b_l u(n-l) \tag{2.1}$$

And where G is a gain factor for the input data and the coefficients a_k, b_l are constant [13] [81].

Most LPC works assume an all-pole model (also known as an *autoregressive*, or AR, model), where $q = 0$. An all-zero model, $p = 0$, is also called *moving average* (MA) model, since the output is a weighted average of the q prior inputs. The more general LPC model with both poles and zeros ($q > 0$) is known as an autoregressive moving average (ARMA) model. The order of an LPC model is the number of poles in the filter. When constructing a model from the data, the order number is usually defined by the user [75].

The process of clustering time series models is usually a three-step procedure. Primarily, each time series is represented by a dynamical model, which is estimated using the given data. Secondly, a distance between the dynamical models is defined and

computed over all the models estimated in the first stage - this distance measure can be the same used to cluster data or features extracted from the data. And finally, a clustering and/or a classification mechanism is performed based on the distance metric defined [12].

This general methodology has been applied previously in different application areas, by estimating similarity measures between the LPC coefficients [92] [5]. However, other method which estimates the cepstral coefficients (which will be defined below) from the LPC model and computes the distance between those coefficients, has been widely used achieving better results [8] [49] [102] [50] [23] [86].

Cepstral Coefficients

Cepstrum analysis is a nonlinear signal processing technique with a variety of applications in areas such as speech and image processing. The term "cepstrum" is a coined word which includes the meaning of the inverse transform of the spectrum. The cepstrum is defined as the inverse Fourier transform of the short-time logarithmic amplitude spectrum.

It is possible to compute the cepstral coefficients (c_n) from the linear prediction coefficients of the data. Consider a time series X_t defined by an AR(p) model:

$$X_t + a_1 X_{t-1} + ... + a_p X_{t-p} \tag{2.2}$$

Where $a_1, a_2, ..., a_p$ are the autoregression coefficients. The LPC cepstral coefficients for a time series of lenght N can be obtained from the autoregression coefficients as follows [32]:

$$A(n) = \begin{cases} -a_1, \text{if n} = 1 \\ -a_n - \sum_{m=1}^{n-1} (1 - \frac{m}{n}) a_m c_{n-m}, \text{if } 1 < \text{n} \leq \text{p} \\ -\sum_{m=1}^{p} (1 - \frac{m}{n}) a_m c_{n-m}, \text{if p} < \text{n} \end{cases} \tag{2.3}$$

Kalpakis et al. [49] compared the clustering results obtained using LPC cepstrum with other methods such as Discrete Fourier Transform, Discrete Wavelet transform, Principal Component Analysis, and more, and achieved better results using the distance between cepstral coefficients.

In this chapter we presented the base concepts that will be used in the next sec-

tions of the thesis. The clustering algorithm used in this research was a distance-based K-Means procedure. Two different distance metrics were used for the clustering mechanism: the root mean square distance measured directly from the data and the euclidean distance between the cepstral coefficients of the data. In the respective sections the procedures will be thoroughly described.

Chapter 3

Signal Processing Algorithms

This chapter exposes the signal processing algorithms created for this study. The concept of a *meanwave* computed from a continuous signal is presented and its procedure described. A set of wave-alignment techniques is also characterized. These algorithms enable the extraction of higher level information from the raw signal.

3.1 *Meanwave*

3.1.1 Concepts

Having a set of signal's cycles, we define *meanwave* as a wave constructed by the mean value of those cycles, for each time-sample. The *meanwave* sums the information from all signal's waves, giving the user a notion of the signal's behavior. Following the same principle, a deviation wave is also defined, computing the standard deviation error value for each sample [64]. This wave presents the variability of all signal's cycles.

For visualization purposes, upper and lower deviation waves are traced, using the *meanwave* and deviation wave information:

$$u_{wave} = meanwave + deviationwave \tag{3.1}$$

$$l_{wave} = meanwave - deviationwave \tag{3.2}$$

The area filling the space between the lower and upper deviation waves, is our "deviation area". This area is relevant for visualization purposes as it shows the variability of the signal's cycles regarding to the *meanwave*.

An example of these concepts is produced in figure 3.1. This figure shows the application of the *meanwave* for signals which are already divided into cycles normalized in time. All the those cycles have the same number of samples - that way, it is possible to compute the mean value for each sample without having missing points. However, for cyclic signals acquired continuously, it's impossible to compute directly the *meanwave*, because prior to that a separation of each cycle is needed, noting that a cycle may last longer than others.

The cycles separation challenge resides in the detection of the quasi-periodic cycle events. Figure 3.2 shows an example of a cyclic signal whose cycles can be separated and represented by a *meanwave*. The cycles separation and the *meanwave* computation is done with the *autoMeanwave* algorithm, which will be presented and described in the following section.

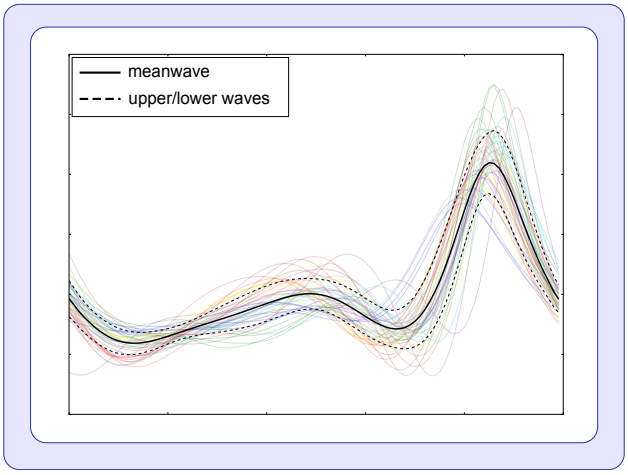

Figure 3.1: Representation of a *meanwave*, upper and lower deviation waves.

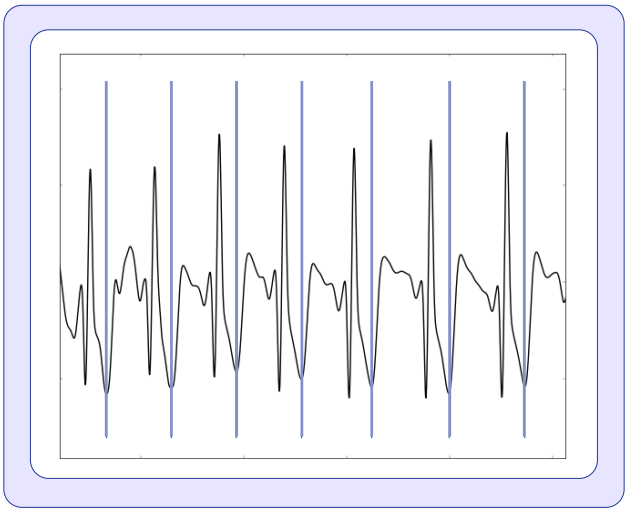

Figure 3.2: Cycles of an ECG signal with events marked.

3.1.2 Algorithm Design

The *autoMeanwave* algorithm is the base function to identify the individual waves. It receives, by input, a cyclic signal, its sampling frequency (f_s) and a trigger mode - the trigger mode is a parameter that can be omitted from the input and will be explained further on this chapter.

The algorithm is divided into sub-functions that return relevant signal's information, such as its fundamental frequency, the cycle's events and the actual *meanwave* and deviation wave. The *autoMeanwave* algorithm also returns the distance of each signal's cycle to the resulting *meanwave*.

As we're working with cyclic signals, the first step of the procedure is to compute the signal's fundamental frequency, to estimate the cycles' size. Then, the signals enter a correlation function to find the instants corresponding to the cycles' events. With those points, the cycles can be separated and the *meanwave* computed. These steps will be described in more detail below.

Fundamental Frequency

The fundamental frequency (f_0) of a periodic or quasi-periodic signal is the inverse of the repeating period pattern length, which is the smallest repeating unit of a signal. Considering the signal as a superposition of sinusoids, the f_0 is the lowest frequency harmonic. The estimation of the f_0 is still a current research topic (particularly in the speech processing field); there isn't any ultimate method to compute it, as a procedure that returns good results for one type of signal can perform poorly for others [33] [77].

For the purpose of this study, the estimation of the f_0 was based on the extraction of the first signal's harmonic. The first step was to compute the absolute value of the fast fourier transform (FFT) signal. Then, a smoothing filter with a moving average window of 5% of signal's length was applied. The first peak's position of the smoothed signal is assumed as the f_0 (figure 3.3).

With the fundamental frequency (f_0) and the sampling frequency (f_s) it's possible to compute the period of the signal (in number of samples). We call that value "window size", opening the window by 20% to use more samples than a cycle:

$$window_{size} = \frac{f_s}{f_0} \times 1.2 \tag{3.3}$$

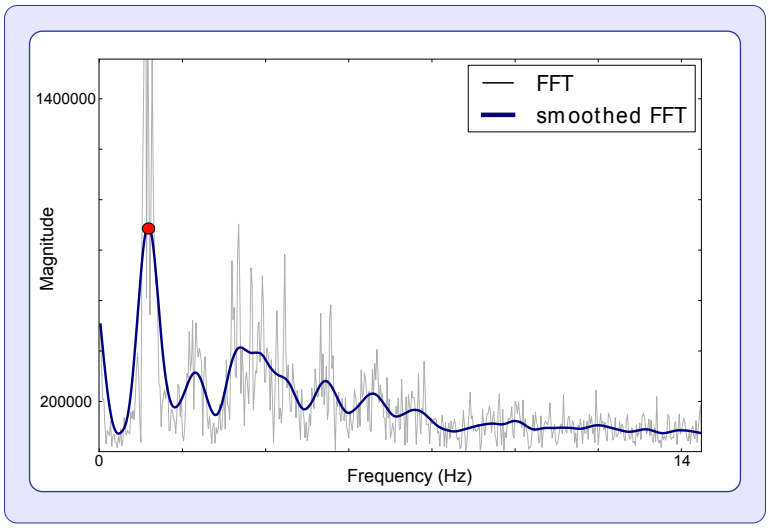

Figure 3.3: FFT raw and smoothed signal with and f_0 estimation.

Although there are more robust methods to determine a signal's f_0, this approach is adequate for this study, as the purpose was to have a close idea of a cycle's size. In fact, more than one exact cycle is used, as a 20% margin is forced. Further on, a correlation function is used to detect the cycles' events, so the f_0 is just a preliminary estimation to support the following algorithms.

Correlation Function

Correlation refers to a measure of the relation between two signals. For the detection of the cycle's events, a correlation function was defined. This function receives a signal, the window size (previously computed) and a distance measuring function.

The function selects a random part of the signal ($window$), with the same number of samples as the window size (N). That small wave slides through the original signal, one sample at a time ($sig_{[i:i+N]}$), and the distance between the two overlapped waves is computed for each sample (i). The expression used to compute the distance value for each sample is exposed in equation 3.4. This was the measuring function given to the algorithm as input, but others can be added and used.

$$distance_i = \frac{\sum_{j=1}^{N} \left| sig_{[i:i+N]j} - window_j \right|}{N} \tag{3.4}$$

With i ranging from 1 to N, the number of signal's samples minus the window size.

The result of this algorithm is a signal composed by distance values. Those values show the difference between the selected sliding window and the overlapped wave in

that instant. The distance signal is composed with minimum peaks, which represent the instant where the two waves are more similar. In a cyclic signal we can predict that this happens once per cycle. A smoothing filter with a moving average window of 0.1% of signal's length is applied to the inverted distance signal and its peaks positions are detected and assumed as our cycle's events.

The process described is illustrated in figure 3.4.

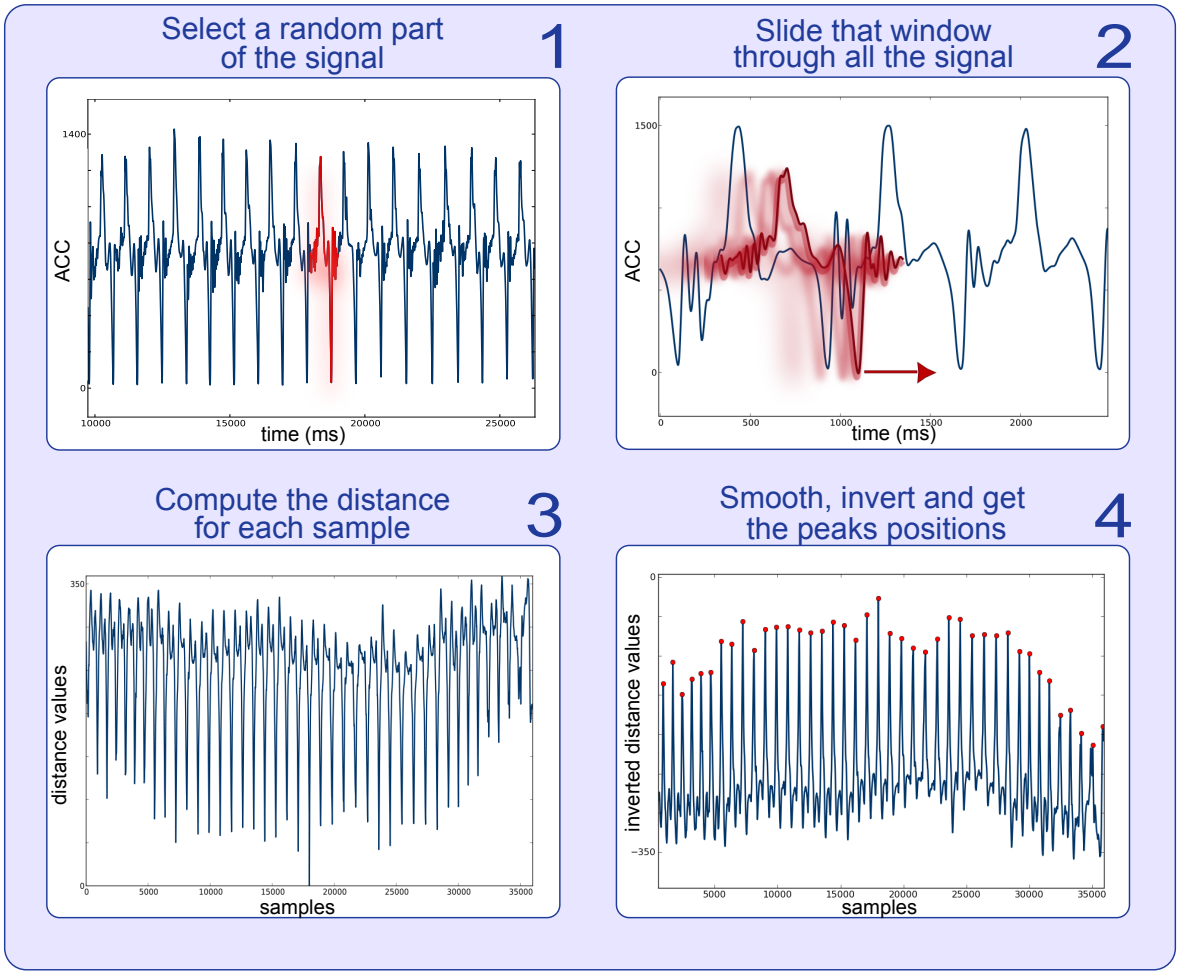

Figure 3.4: Illustration of the correlation process. Left at the top: selection of a random part of the signal; Right and top: slide the selected part through the signal; Left at the bottom: the distance between overlapped waves for each sample traces a signal of distances; Right at the bottom: the signal of distances is smoothed, inverted and its peaks are computed.

Meanwave Computation

As mentioned above, to compute a *meanwave*, the signal's cycles must be separated. This is possible with the information of the cycles' events positions and the window

size (previously computed):

$$cycle_i = signal[event_i - \frac{window_{size}}{2} : event_i + \frac{window_{size}}{2}] \tag{3.5}$$

With i ranging from 1 to the number of events/cycles. Some cycles may last longer than others, causing the ending of a cycle to be equal to the start of another, so a normalization of all the cycles in time isn't necessary to compute the *meanwave*.

With the cycles separated it's possible to compute the mean and standard deviation error value for each cycle sample, creating a *meanwave* and a deviation wave.

For visualization purposes, a final adjustment is made: a rearrangement of the events' positions. The events were defined as the time instant in which the two overlapped waves of the correlation procedure were more similar. In this procedure, as the window that goes through the signal is randomly chosen, the resulting events may not be the best for the visualization of the *meanwave*.

A trigger position is computed to rearrange the cycles' events. The trigger position is a notable point of the previously traced *meanwave* and it's computed after selecting a trigger mode (input of the *autoMeanwave* algorithm). A few possibilities for this variable were designed and are exposed in table 3.1.

Table 3.1: Trigger mode options.

input string	Trigger Position
'max'	*meanwave*'s maximum point
'min'	*meanwave*'s minimum point
'zeropos'	*meanwave*'s zero transition to positive values
'zeroneg'	*meanwave*'s zero transition to negative values
'diffpos'	maximum point of the *meanwave*'s derivative signal
'diffneg'	minimum point of the *meanwave*'s derivative signal

The maximum point of the *meanwave* is the default selection if an option is omitted in the input. After the calculation of the *meanwave*'s chosen trigger position, the events can be recalculated:

$$event_i = event_i + trigger_{position} - \frac{window_{size}}{2} \tag{3.6}$$

With i ranging from 1 to the number of events. A new *meanwave* and deviation wave is computed after the rearrangement of the cycles' events. An example of the trigger influence in the rearrangement of the events is exposed in figure 3.5.

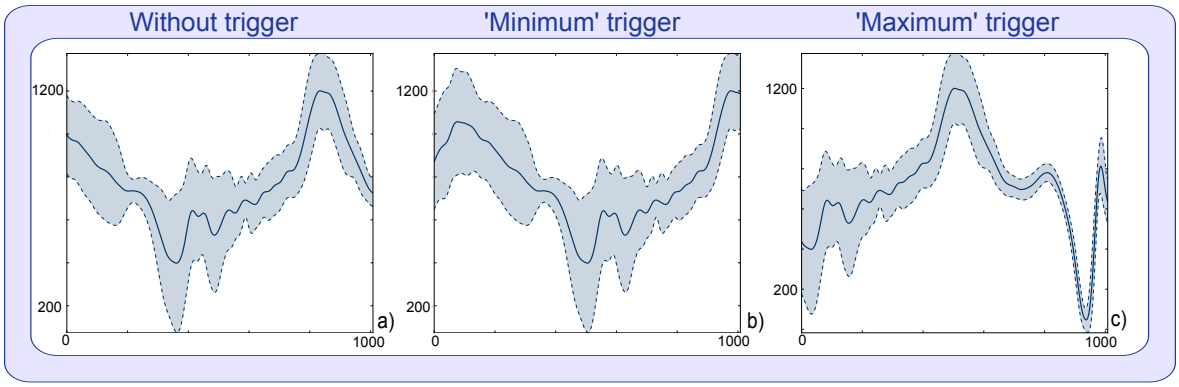

Figure 3.5: Example of trigger influence in *meanwave*'s visualization: a) no trigger; b) trigger in the minimum point; c) trigger in the maximum point.

The *autoMeanwave* algorithm also returns an array with the distance of each cycle to the final *meanwave*. The root mean square error formula was used to compute the distance between the waves:

$$distance = \sqrt{\frac{\sum_{i=1}^{l}(cycle_i - meanwave_i)^2}{l}} \tag{3.7}$$

With i ranging from 1 to l, the length of the *meanwave*. This is computed for all the cycles, composing a distances signal.

3.2 Signal Alignment Techniques

3.2.1 Concepts

In our research, the idea of a wave-alignment algorithm arises to improve the information given by the *meanwave*, minimizing its deviation area. Given a set of separated cycles, this algorithm aligns all waves in a specific point. This is done by making a shift in time to all the waves until its notable point coincides with the position chosen for the alignment.

In the following section, the algorithm created for waves-alignment will be described. Figure 3.6 shows a *meanwave* computed with non-aligned cycles. The fixed position of alignment is traced (the *meanwave*'s peak position) and the shift needed from each wave is illustrated, considering the position of its maximums as the notable point.

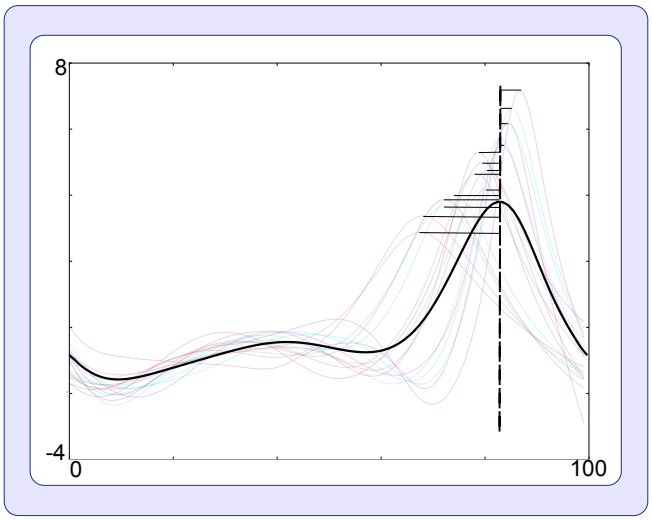

Figure 3.6: Waves alignment in the maximum point of the *meanwave*.

3.2.2 Algorithm Design

This algorithm receives as input the cycles to align and the type of alignment desired and returns a *meanwave* created with the aligned data. As this algorithm has its basis in the *meanwave*, the first step is to compute it. A *meanwave*'s notable point is chosen as the position of alignment and all the waves are moved to align with that point. After this procedure, a new *meanwave* is traced. These steps will be described in more detail below.

Position of Alignment

The position of alignment is calculated after the *meanwave*'s computation and with the desired alignment type's information. The type of alignment chosen is represented by a string given as input to the algorithm. In this study, a few options were designed for this variable and are exposed in table 3.2. This table shows some resembles with table 3.1 (from page 28), however, the trigger mode only rearranges the events for a better visualization of the *meanwave*, not changing its information. The alignment mode produces different displacements in each wave and modifies the *meanwave*'s information.

The reference wave for the alignment is the *meanwave*, and all the waves are moved to align with it. The first six alignment modes stated in table 3.2 correspond to the first and second order derivative zeros. The *'corrAlign'* mode uses a function that computes the correlation signal between two signals and finds the correlation's maximum point; that point is where the signals are more similar, so it's chosen as the alignment point.

Table 3.2: Alignment types options.

input string	Position of alignment for the two waves
'maxAlign'	Absolute maximum points
'minAlign'	Absolute minimum points
'peakAlign'	Relative maximum point
'peakNegAlign'	Relative minimum point
'diffMaxAlign'	Absolute maximum point of the derivative signals
'diffMinAlign'	Absolute minimum point of the derivative signals
'corrAlign'	Maximum point of the waves' correlation signal
'bestAlign'	Point that minimizes the distance between waves

The last alignment mode, *'bestAlign'*, makes each wave go through the fixed *meanwave*, computing the distances for each sample. The position which gives a lower distance value is the position of alignment for that wave. The formula used to estimate the distance between the waves for each sample an adapted root mean square error distance (equation 3.8). The distance is only computed for the time periods in which the wave has at least 40% of its samples overlapped with the *meanwave*.

$$distance = \sqrt{\frac{\sum_{i=f}^{l}(wave_i - meanwave_i)^2}{PO}} \tag{3.8}$$

With i ranging from f, the first sample in which the waves overlap, to l, the last overlapped sample (see figure 3.7). The result of the square root is divided by PO, the percentage of overlap. This variable is 1 when the wave is totally overlapped with the *meanwave*, and the minimum value it can assume is 0.4, as we only compute the distance for waves with 40% of samples overlapped.

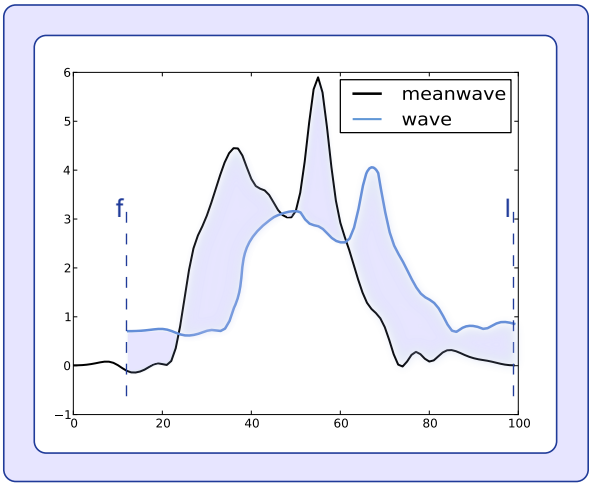

Figure 3.7: Illustration of the overlapped area between two waves. The distance value is computed from sample f to sample l.

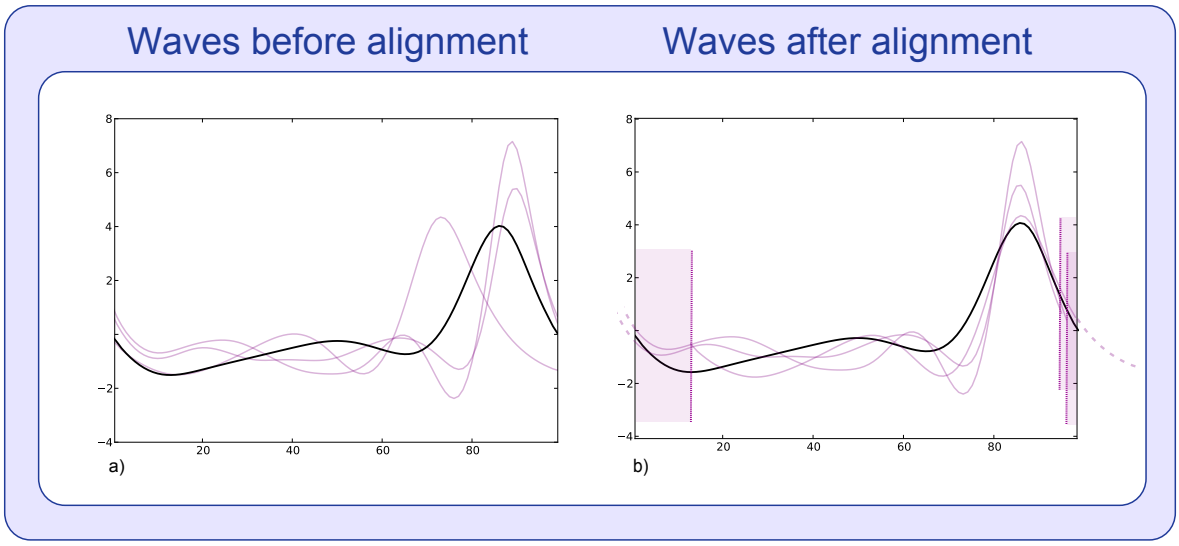

Figure 3.8: Waves a) before and b) after alignment and its effect on the borders.

Wave Alignment

The alignment's positions are computed for the waves and the *meanwave*; the difference between those points gives the resulting number of samples each wave has to shift to align with the *meanwave*. If the wave displacement is positive (the wave has to move forward), the last samples of this wave are removed, and the first border points become "empty" values. If the displacement is negative (the wave moves backwards), the initial border is removed and the final border presents "empty" samples. Figure 3.8 illustrates this case. The number of the "empty" samples in the borders is defined by the number of samples each wave has to shift in the alignment.

To overcome the problems given by the existence of "empty" samples in some waves, it was necessary to manipulate masked arrays. Masked arrays are arrays that have missing or invalid entries [39]. Its usage is needed in datasets which are incomplete or tainted by the presence of invalid data. We use these arrays because the shifting in the waves creates the necessity to introduce "empty samples" in one wave's border. With the masked arrays it is possible to attribute a NaN (not a number) to a sample, instead of a real value, allowing the processing and visualization of the aligned waves.

After the wave alignment it is possible to compute a new *meanwave*. This *meanwave* is traced using just the samples in which all the waves overlap; therefore, the new aligned *meanwave* has a part of its borders removed. The number of samples missing from the aligned *meanwave* will depend on the type of alignment chosen. Figure 3.9 shows an example of a *meanwave* before and after alignment.

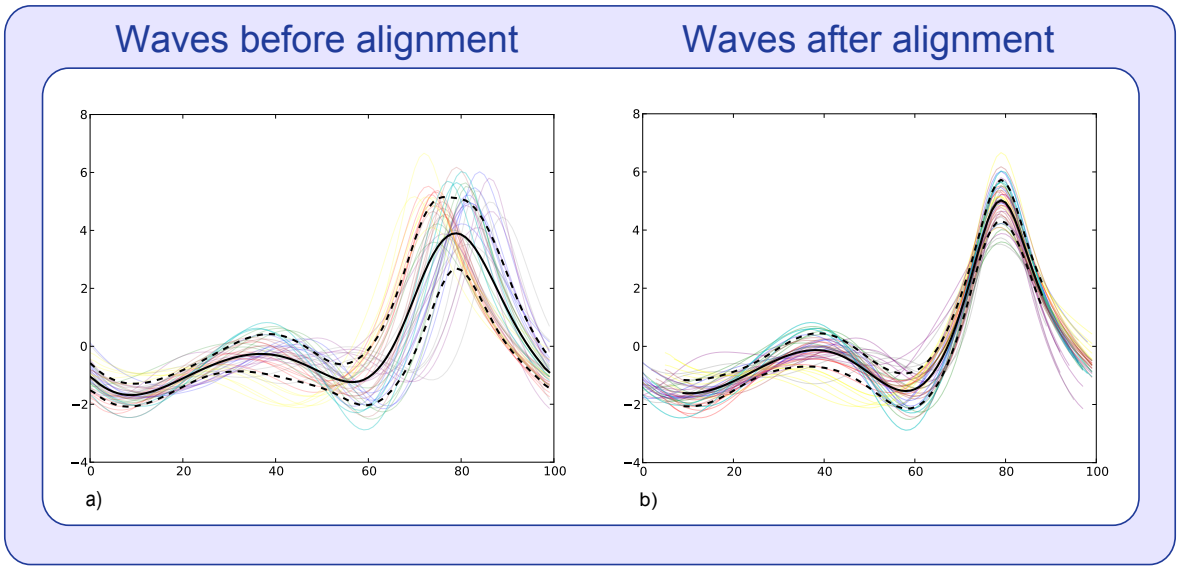

Figure 3.9: Example of a *meanwave* a) before and b) after alignment.

Chapter 4

Performance Evaluation

In this chapter, an evaluation of the signal processing algorithms previously designed is made. The results obtained for the *autoMeanwave* evaluation procedure were compared to the results obtained with a state of the art algorithm which models and clusters the data with a distance metric obtained from its cepstral coefficients.

4.1 *autoMeanwave* Evaluation

4.1.1 Overview

For the evaluation of the *autoMeanwave* algorithm, we acquired several cyclic biosignals with the subjects performing tasks composed by two different modes. We also composed a synthetic signal with the distinctive modes. To distinguish the different modes in the same acquired signal, a function which receives the information returned from the *autoMeanwave* algorithm and measures the distances between the cycles was used. The distances computed with that function were the input of a K-Means clustering algorithm, used to differentiate the signals.

In the following sections the methods for the signal acquisition and processing will be described and the results obtained will be presented.

4.1.2 Signal Acquisition

To acquire the signals necessary to this study three sensors were used: an electromyography sensor (*emgPLUX*), a triaxial accelerometer (*xyzPLUX*) and a finger blood

Figure 4.1: bioPLUX research system.

volume pressure sensor (*bvpPLUX*).

The signals analog to digital conversion and bluetooth transmission to the computer was made with a wireless signal acquisition system, the bioPLUX research system (figure 4.1). This system is portable, small sized and light-weighted, has a 12 bit ADC and a sampling frequency of 1000 Hz [73].

All sensors were connected to the channels of the bioPLUX system and the signals were acquired continuously in real time. In the acquisitions with accelerometers, just the axis with inferior-superior direction was connected to the bioPLUX. The signals were saved and processed offline.

Several tasks were designed and executed in order to acquire signals with two distinctive modes. We conceived a synthetic digital signal and collected signals from six different activities scenarios with the accelerometer, EMG and BVP sensors. The signals acquired are available online at the Opensignals Website [71].

Synthetic Signal

A synthetic cycle was traced using a random walk [93] of 100 samples, low-pass filtered with a moving average smoothing window of 10% of signal's length and multiplied by a hanning window. This synthetic cycle was repeated 30 times, so all the cycles were identical for the first mode. For the second mode, a small change of 40 samples was introduced in the syntethic cycle, and that new cycle was repeated 20 more times.

This procedure generated a synthetic wave which has identical cycles in each mode. The resulting wave and corresponding cycles are illustrated in figure 4.2.

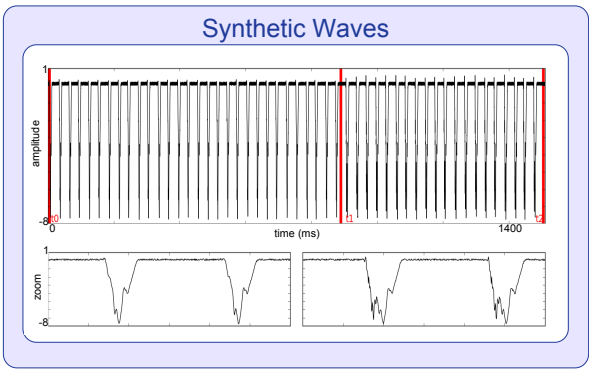

Figure 4.2: Synthetic signal with identical waves from t_0 to t_1 (first mode) and from t_1 to t_2 (second mode) and corresponding zoomed waves of each mode.

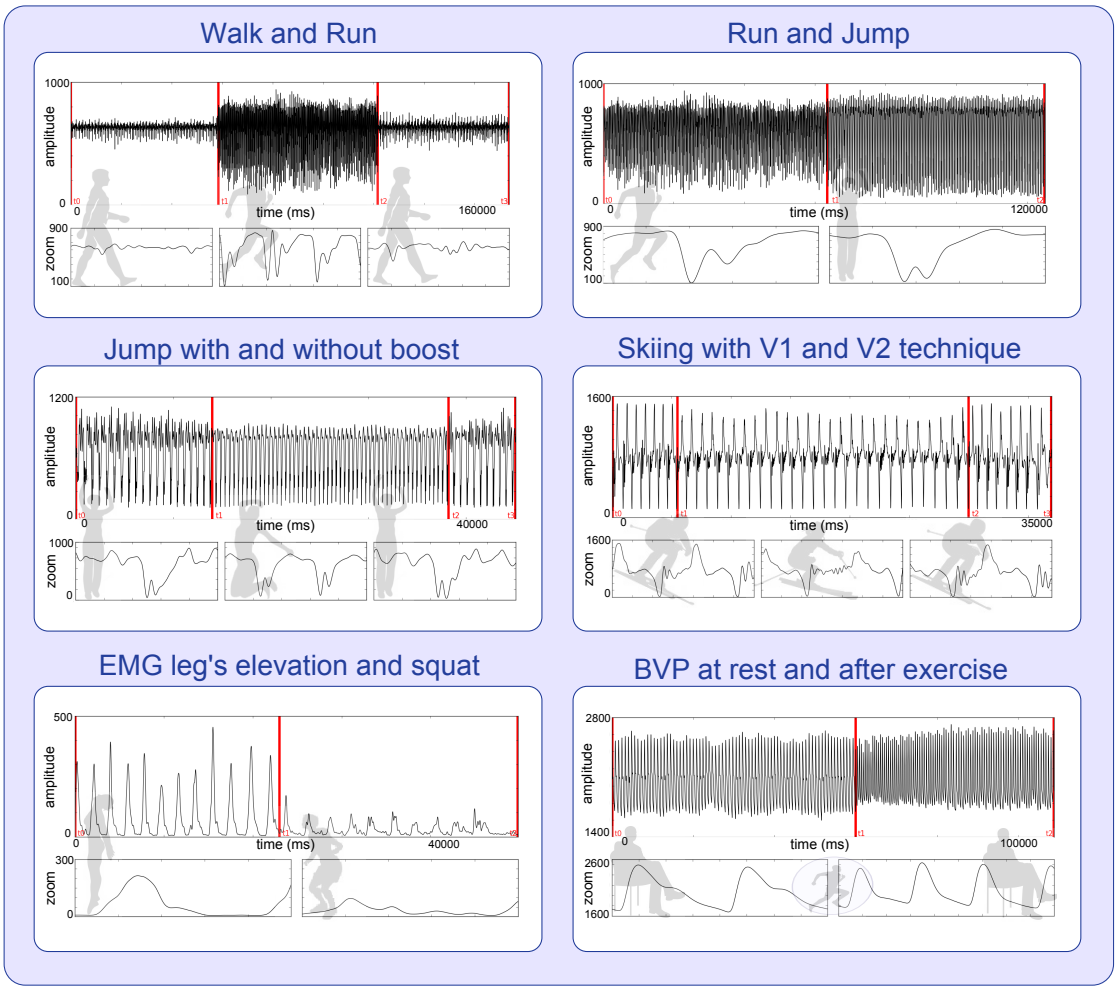

Figure 4.3: Acquired signals with two different modes and corresponding zoomed waves.

Acquired Signals

All six acquired signals and its corresponding zoomed cycles are exposed in figure 4.3.

The tasks "Walk and Run", "Run and Jump" and "Jump with and without boost" were acquired using an accelerometer located at the right hip of the subjects and oriented so that the accelerometer's used axis was pointing upward.

In the first task, "Walk and Run", the subjects were asked to walk and run continuously in large circles. The subjects walked for about one minute at a slow speed, then spent another minute running and ended walking again for one more minute.

For the "Run and Jump" task the subjects performed a task of running and jumping continuously. The subjects spent one minute running, followed by one minute jumping.

In the "Jump with and without boost" task, the subject performed the following procedure: approximately 14 seconds of "normal" jumping (small jumps without a big impulsion), followed by 24 seconds jumping with a big boost, ending with 7 more seconds of normal jumps.

The last of the acceleration signals is the "Skiing" task, acquired during a cross-country (XC) skiing study with the accelerometer attached to the subject's ski pole, below the hand grip[1]. For this task, the subject performed two different skiing techniques, called V1 and V2. V1 skate is an asymmetrical uphill technique involving one poling action over every second leg stroke. V2 skate is used for moderate uphill slopes and on level terrain, involving one poling action for each leg stroke [4]. The first 7 cycles of the signal were produced through a V2 technique, the next 27 cycles a V1 technique was used and in the final 8 cycles the technique was V2 again.

With the EMG sensor, the tasked performed was the "Leg's Elevation and Squat". The subject, standing straight with both feet completely on the ground, performed 12 elevations of the legs - getting on the tiptoes and back with both feet completely on the ground -, followed by 11 squats - bending the knees and back standing straight. The EMG data was collected using bipolar electrodes at the *gastrocnemius* muscle of the right leg.

In the last task, "At Rest and After Exercise", the subjects were instrumented with a BVP sensor on the fourth finger of the left hand and were siting with their left forearm

[1]The skiing signal was acquired by Havard Myklebust and Jostein Hallén, researchers from Norwegian School of Sport Sciences. We were allowed to use their signal in this study [62].

resting on a platform. One acquisition was made with the subjects sitting at rest. Then, the subjects were asked to perform intensive exercise which was not collected to avoid undesirable movement artifacts. Immediately after the exercise, another BVP signal was acquired with the subject sitting again, but tired. For the purpose of this study both signals (at rest and after exercise) were used in the same file, cutting a part of each signal and concatenating them offline.

4.1.3 Signal Processing

Figure 4.4 describes the method used to process the signals. All biosignals were submitted to a signal-specific pre-processing phase and then to a generic signal-independent phase (composed by a *autoMeanwave* and clustering procedure) which was applied to all the signals of this study.

Pre-processing Phase

The pre-processing phase is signal specific, as each signal has different treatments depending on the type of sensor used, how much noise is present or if a calibration is needed. In this phase, the acceleration signals were low-pass filtered using a smoothing filter with a moving average window of 50 samples. The BVP signal was also low-passed filtered with the same moving average window. Random noise with 1/5 of original signal's amplitude was added to the synthetic signal. The EMG signal was centered at y axis zero (subtracting its mean value), rectified and filtered with a smoothing average window of 300 samples.

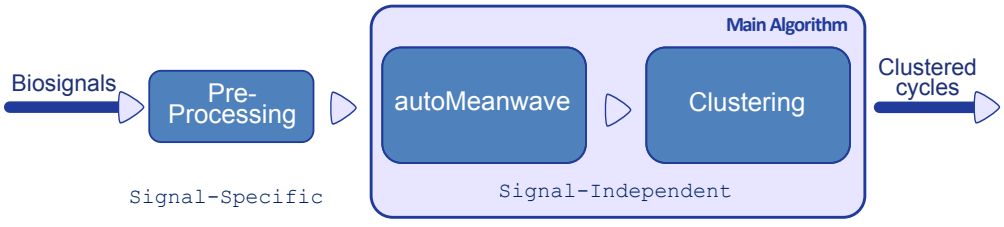

Figure 4.4: Signal's processing procedure schematics. The procedure is divided into two phases: one phase is signal specific (pre-processing block) and the other is signal independent (main algorithm composed by *autoMeanwave* and Clustering blocks).

autoMeanwave Procedure

The *autoMeanwave* algorithm, described in section 3.1, received by input the pre-processed signals. All signals had a sampling frequency of 1000Hz, and the trigger mode chosen was the minimum point of the *meanwave*. A set of *meanwaves* and deviation waves resulted from this procedure. The events and window size variables were also returned to be used in the clustering procedure.

Clustering Procedure

For the clustering procedure we developed a function that receives the signal to cluster, the cycles' window size and events produced with the *autoMeanwave* algorithm. This function goes through all the events and for each selects a period of the signal with center at that event and a number of samples to both sizes equal to the window size. That period is compared with each one of the others (with the center in the other events and the same window size), using the equation 4.1 for all cycles' samples:

$$distance = \sqrt{\frac{\sum_{i=1}^{l}(cycle1_i - cycle2_i)^2}{l}} \tag{4.1}$$

With i ranging from 1 to l, the cycles' length. This equation exemplifies the procedure only for two cycles. Applying equation 4.1 to all cycles, a matrix of distance values is created, as exemplified in figure 4.5.

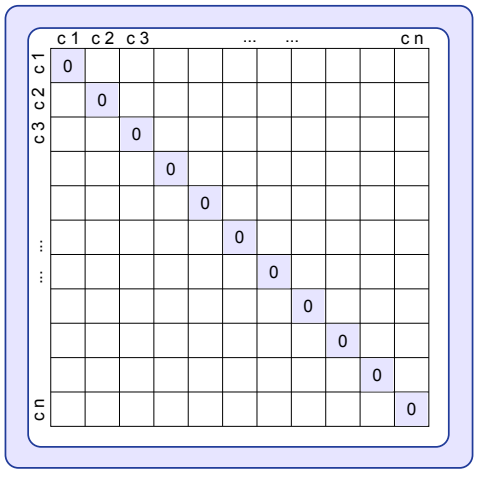

Figure 4.5: Representation of a matrix for the distances between each cycle. Each line represent the distance of one cycle to all the others. The diagonal line represent the distance of one cycle to itself (distance equal to zero).

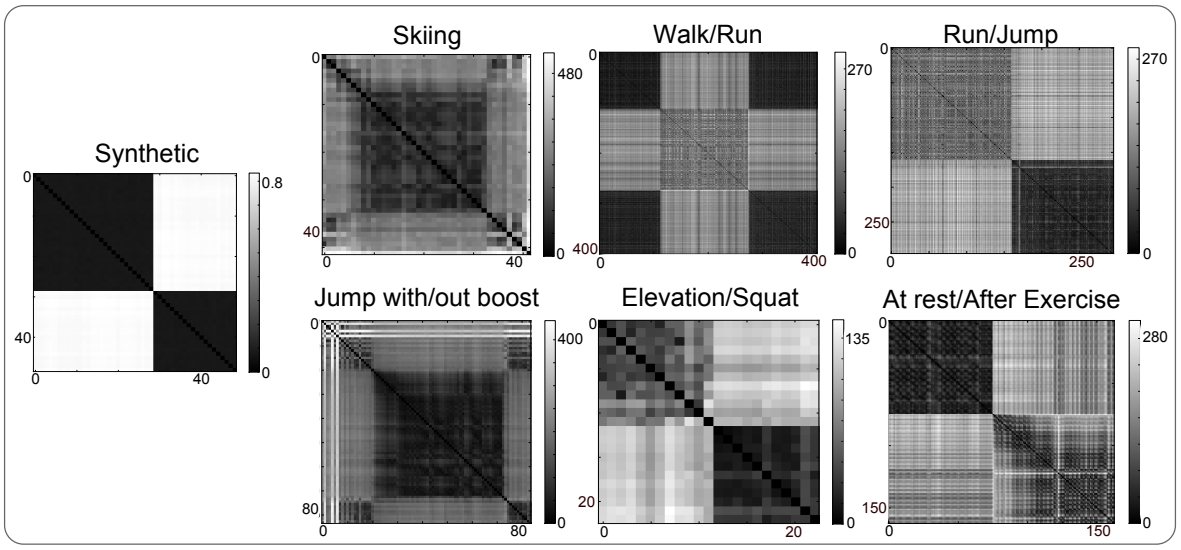

Figure 4.6: Illustration of the distances matrices obtained for each studied task.

Figure 4.6 illustrates the matrices of distances obtained for all tasks. These figures use colors to represent the distance values: white color represents maximum distances values and black color represents minimum distances - note that all matrices' diagonals are black, corresponding to the distance between a wave to itself. The synthetic waves matrix seems almost ideal because all the waves are equal in each mode - the distance values between waves are minimums or maximums. The other matrices show a greater variation of distances, as the cycles are not exactly identical for each mode. However, it's still visible the similarity between the cycles of the same mode.

A K-means algorithm was used to cluster the signals. This algorithm receives the distances matrices and the number of clusters expected in the data - which in this study was always two - returning the clusters and the distances to the clusters.

4.1.4 Results

Figure 4.7 shows the graphics of the resulting *meanwaves* and deviation areas, before and after the clustering procedure. According to the clustering results, the modes of the signal were separated and a *meanwave* was computed for each. Before the clustering, the *meanwaves* gather information about the behavior of the two signal's modes, showing changes in shape and frequency. The reshaped *meanwaves*, traced after the clustering procedure, show an overall reduction of the deviation area and a higher similarity to each mode's cycles.

Figure 4.7: At left: resulting *meanwaves* before the clustering procedure for each task. At right: resulting *meanwaves* gathered after the clustering procedure calculated separately for each mode and task.

Table 4.1: Clustering Results.

Task	Number of cycles	Cycles correctly clustered	Errors	Misses
Synthetic	50	49	0	1
Walk/Run	343	342	1	0
Run/Jump	296	295	1	0
Jumps with/without boost	85	84	1	0
Skiing V1/V2	42	41	0	1
Leg's Elevation/Squat	23	23	0	0
At Rest/After Exercise	165	159	4	2
All	1004	993	7	4

The results of the clustering procedure, for each task, are exposed in table 4.1.

It is important to note that some cycles weren't classified (*Misses* column in table 4.1) because the borders of some signals didn't had full cycles. The function created to compute the distances matrices cannot be used to compare a short cycle with the regular ones. Therefore, those 4 cycles were rejected for lack of signal quality.

In the "Walk and Run" activity there were some extra classification points. The cycles were correctly clustered - only 1 error encountered - but in the "Walking" mode, some extra points among the cycles were also clustered. That occurs due to a relatively large variation in the fundamental frequency from the walking to the running activity - despite one activity has all cycles well defined by the window size variable, the other has less than one cycle per window size. This condition shows a limitation of our algorithm: it doesn't allow big changes in the frequency domain for the different signal's modes.

Only 7 errors resulted from our procedure, and two of those errors occurred in transition periods. In the transition to another mode, the cycle is reshaping and the distance value to the *meanwave* or to the clusters mean values is bigger than anywhere else on the signal. This occurred in the "Jumps with and without boost" and in the "Walk and Run" activities.

This clustering algorithm, based on the *meanwave* information, only returned 7 errors out of 1000 cycles with pattern quality. Therefore, 99.3% of efficiency was achieved in this study. Our algorithm shows an effective detection of signal variations, tracing different patterns for distinct modes, whether it's an activity, synthetic or physiological signal.

This study was previously published and presented at the BIOSTEC 2011 conference [64][Appendix A].

4.2 Comparison with Cepstral Coefficients

4.2.1 Overview

A review of the literature was made to understand the state of the art in clustering time series data. As mencioned before in section 2.2.3, a method that achieved good results began to be widely used: clustering with a distance metric given by the data's cepstral coefficients. Some publications use the Euclidean Distance between the LPC

cepstrum of two time series as their dissimilarity measure to cluster a public database [49] [102] [8].

The algorithm described in those papers (which will be called "*cepstral algorithm*" from now on) was replicated in our research and applied to the public database, described in the following section, to achieve the same results as the ones documented. That guarantees the validity of our *cepstral algorithm*'s implementation. Our implementation was compared with the results exposed by Anthony Bagnall [8], as his publication was the only which used a k-means clustering procedure. After this preliminary comparison, the *cepstral algorithm* was applied to the signals used in our previous study (section 4.1) and the results of both algorithms were compared.

In the following sub-sections, the public database used will be described, as well as the creation of the algorithm and the results obtained for the comparisons.

4.2.2 Database

The studies referenced before used a public dataset of ECG signals. This dataset can be obtained from the ECG database at PhysioNet [72].

Three groups of those time series were used in these experiments. Group 1 is composed with 22 signals, each with 2 seconds of ECG recordings of people having malignant ventricular arrhythmia. Group 2 includes 13 signals with 2 seconds of normal ECG recordings of healthy people. Group 3 has 35 signals with 2 seconds of ECG recordings during supraventricular arrhythmia. Figure 4.8 shows one example of a time series in each group.

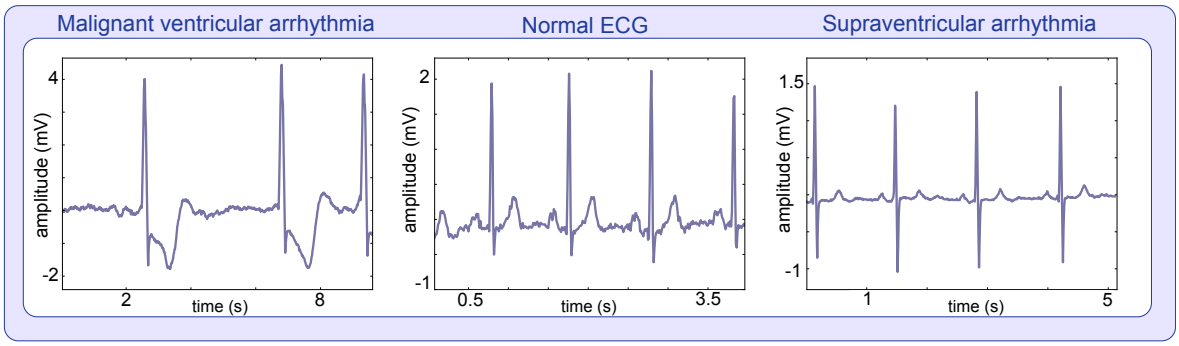

Figure 4.8: Examples of each group's signal from the public dataset: a) malignant ventricular arrhythmia; b) normal ECG and c) supraventricular arrhythmia.

Two collections were defined in these researches: collection one comprises the first two groups (35 signals), and collection two gathers group 2 and 3 (48 signals). These collections were submitted separately to the *cepstral algorithm* to find the two different clusters in each.

For the actual comparison between the performance of the *autoMeanwave* and the *cepstral algorithm*, the dataset used was the one already described in 4.1.2.

4.2.3 Algorithm Implementation

As mentioned before, the algorithm's implementation was described in the literature, and replicated in this study. The steps of the procedure will be presented below, as well its implementation in the continuously acquired data.

Cepstral Algorithm

The first step of this algorithm is to fit a LPC model to the raw data, with an order equal to 6. Among the direct transformations of LPC parameters, one is a filtering process to get the cepstral coefficients. Using the LPC coefficients estimation we computed the five cepstral coefficients (order - 1) of each time series. After that, the Euclidean distance between the signals' coefficients was estimated. Using equation 4.2 to all signals, a distances matrix is computed.

$$distance = \sqrt{\sum_{i=1}^{5}(s1cc_i - s2cc_i)^2} \tag{4.2}$$

This matrix is given to the K-means algorithm, which separates the time series into two different clusters.

Cepstral Algorithm in Continuous Signals

The *cepstral algorithm* was created to be applied in various separated time series, and not to a single signal which has two modes in it. So, for the comparison of both algorithms using the data collected for the *autoMeanwave* study, it was necessary to separate the signal's cycles prior to the application of the *cepstral algorithm*. The separated cycles were exactly the same for both algorithms.

Table 4.2: Comparison of the results obtained for the ECG Database.

ECG Database	Results described in literature	Results with our implementation
Collection 1	62.5%	100%
Collection 2	62.5%	62.5%

Table 4.3: Comparison of the results obtained by both algorithms.

Task	Accuracy of *cepstral algorithm*	Accuracy of *autoMeanwave* algorithm
Synthetic	100.0%	100.0%
Walk/Run	92.4%	99.7%
Run/Jump	68.2%	99.7%
Jumps with/without boost	82.1%	98.8%
Skiing V1/V2	90.2%	100.0%
Leg's Elevation/Squat	56.5%	100.0%
At Rest/After Exercise	68.7%	96.4%
All	80.0%	99.3%

4.2.4 Results

The results of the implemented *cepstral algorithm* and the one described by Anthony Bagnall applied in the ECG database are exposed in table 4.2. Looking at the results, it's reasonable to affirm that not only we were able to reach to Anthony's results, but actually surpass them. The accuracy percentage for each collection was computed with equation 4.3. For collection 2 we achieved the same accuracy as literature, but for collection 1 our accuracy was 100%.

$$accuracy = \frac{\text{number of correctly clustered signals}}{\text{total number of signals}} \times 100\% \tag{4.3}$$

With these results it's safe to say that the *cepstral algorithm* was successfully replicated and is valid for comparison with other algorithms.

Table 4.3 gathers the clustering accuracy results obtained for each task and algorithm used. The accuracy percentage was computed using the same expression referenced above (equation 4.3), but with signal' cycles instead of the whole time series.

Our *autoMeanwave* procedure presents a higher accuracy level for every signal but the synthetic waves, for which the accuracy is the same. Looking at the overall results, our algorithm achieved 99.3% of efficiency, and the cepstral algorithm only 80.0% for the same signals - which from the tests with this database makes our approach a better

option for clustering cyclic signals. To note also that our algorithm can be applied to a continuous signal with different modes in it, automatically separating the signal's cycles and computing a distance metric for each. The *cepstral algorithm*, however, has to be applied in separated signals - in this study we had to isolate the cycles before aplying the cepstral procedure.

In conclusion, the comparison between the two algorithms showed that the *autoMeanwave* algorithm has a high efficiency level, giving better results and is more suitable for the type of data analyzed than a state of the art algorithm in this area.

4.3 Signal Alignment Techniques Evaluation

4.3.1 Overview

The evaluation of the alignment algorithm designed was made in the context of a study in collaboration with the Human Kinetics Faculty (*FMH, Portugal*). The study focused on the video collection's analysis of mice gait cycles after peripheral sciatic nerve injury. The rats were divided into 4 groups and in 4 weeks its gait cycles were acquired and analyzed, to see which group presented parameters closer to normal. Two different recovery strategies were used in two groups to understand its influence in gait signals. A *meanwave* was computed for each of these scenarios and a set of features was extracted from it. As each mouse took steps with different velocities, its waveshapes weren't coincident and therefore, to get better results from the *meanwave*, an alignment was made to all cycles. After that, some features were extracted from the aligned *meanwave*. We focused on the peak values of ankle angular velocity as parameter to assess changes in the pattern of coordination motion exhibited and verify stability of the pattern after the injury. An experimental study was performed to verify if different mechanical load would induce differences on gait recovery after sciatic nerve crush.

The methods for the signal acquisition and a contextualization on the signal processing and consequent results will be described below.

4.3.2 Methods

A total of 32 adult Sprague-Dawley male rats were randomly separated in the following groups: (1) sciatic crush plus treadmill-walking, (2) sciatic crush plus passive exercise, (3) sciatic crush and (4) sham-operated control. The follow-up period was 12 weeks - acquisitions were made 1, 4, 8 and 12 weeks after a sciatic nerve crush injury. The recovery strategy used for group 1 was regular treadmill-walking between week 2 and week 12 after injury. The treadmill training consists of 30 minutes of walking on a motorized treadmill at a speed of 10 m/min. For group 2, a passive stretch-shortening exercise was performed manually between week 2 and week 12 after injury. Passive exercise consisted on 6 minutes of passive range of motion at hip, knee and ankle joints simultaneously (flexion-extension) performed manually. Animals in the groups 3 and 4 remained in their cages during the 12-weeks recovery period with no other intervention. The sham-operated control group (group 4) was the only group in which the sciatic nerve wasn't injured. However, this group had also a sham operation, defined to differentiate the effects of surgery (pain, skin, muscle or bone damage) from the effects of the nerve injury itself. This works as a placebo operation protocol [63].

An optoelectronic system of six infrared cameras operating at a frame rate of 200Hz was used to record the motion of right hindlimb. Seven reflective markers were attached to the right hindlimb to access the tibiotarsal joint kinematics in time. The tibiotarsal joint is the joint between the tibia and the tarsus (figure 4.9).

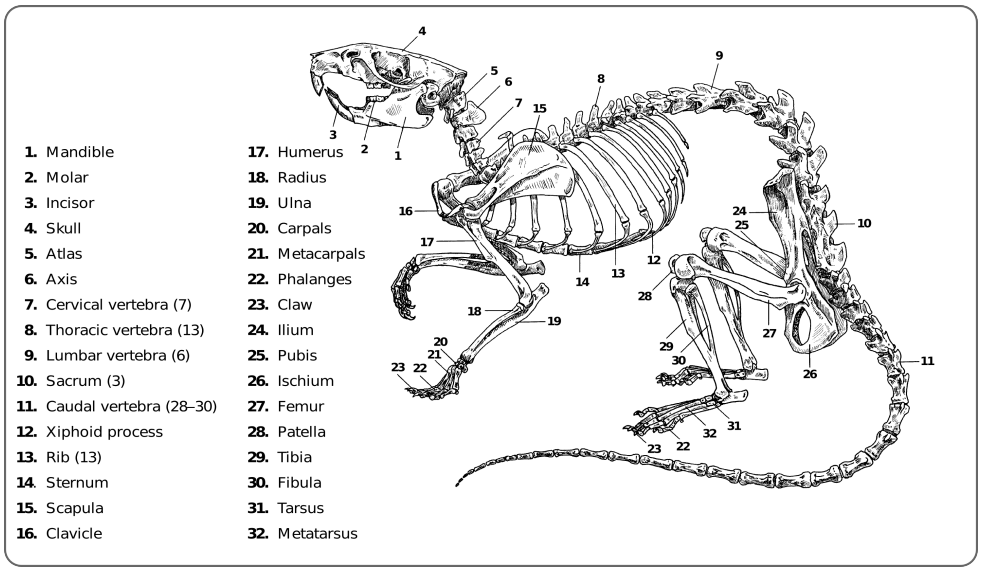

Figure 4.9: Illustration of the rat's skeleton. The Tibia (number 29) and Tarsus (number 31) are visible in this figure. From [67].

The reflective markers' trajectory was smoothed using a Butterworth low-pass filter (6 Hz cut-off) and the data was obtained by averaging six walking cycles. The signals were then normalized in time, so that the cycles varied from 0 to 100% [2].

4.3.3 Signal Processing

The set of waves acquired was submitted to our wave-alignment algorithm for a reduction of the cycles' variability and a better evaluation on the notable point of alignment. The scheme of this signal processing phase is represented in figure 4.10. The technique used for alignment and consequent features extracted will be defined below.

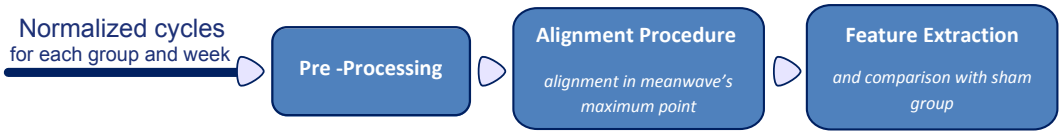

Figure 4.10: Alignment's evaluation processing scheme.

Alignment Technique

The data returned from the reflective markers were angular displacement cycles, and it was desired to work with cycles of angular velocity, so the first step in the pre-processing phase was to compute the derivative signal of each cycle. After that, the cycles were divided into the correspondent groups and weeks, and *meanwaves* were traced for each.

After a first look at the resulting *meanwaves*, a pattern in the waves was noticed: there was a peak located around 70% of the cycle which was growing in amplitude each week. It was decided to use that peak's position as the notable point of each wave in the wave-alignment algorithm. Each wave was then aligned with the maximum peak of the reference *meanwave* for each group and week.

For each reference *meanwave*, the deviation area (std_{normal}) was quantified by computing the difference between the upper and lower deviation waves for all samples and calculating its mean value. For the aligned *meanwaves* ($std_{aligned}$) the procedure was the same. To quantify the reduction of the deviation area with the alignment, equation 4.4 was applied for all 16 *meanwaves*, and its mean value computed.

[2]The procedures here described were developed by S. Amado, P. Armada-da-Silva and A. Veloso, researchers at FMH [2].

$$std_{total} = \frac{std_{normal} - std_{aligned}}{std_{normal}} \tag{4.4}$$

Features Extracted

After the alignment in the maximum point for all waves, it was possible to quantify the variability of the angular velocity peak's amplitude.

With that purpose, the peak's mean and standard deviation error values were withdrawn from the aligned *meanwave* and deviation wave, of each group and week. The values obtained were compared with week 12 of group 4, which was supposed to be the week where the rats were fully recovered (positive control).

It was also quantified the number of cycles that had its peaks between the interval $[mean \pm x \times std]$ of group four's final week (we used x=0.5, x=1 and x=2).

To evaluate the variability of a point's amplitude, the alignment of the waves in that point seems to be a good choice to reduce the error percentage.

4.3.4 Results

Figure 4.11 shows the resulting *meanwaves* before and after alignment. It's notorious that this alignment produces better results considering the variability of the the peak amplitude. With this alignment, the deviation area of all *meanwaves* has decreased, on average, by 32.8%.

The kinematic analysis distinguished recovery pattern between groups. With the features extracted from the aligned *meanwaves*, it was verified that the peak value of ankle angular velocity for the walking group (group 1) achieved the pattern of sham group (group 4) earlier and with lower variability. On the contrary, passive exercise revealed higher variability in this parameter. Table 4.4 confirms this statement.

Table 4.4: Mean and standard deviation error values for the peak amplitude of each week and group

$Mean \pm std$	W01	W04	W08	W12
G01	0.4 ± 0.9	2.6 ± 0.6	2.4 ± 0.6	2.7 ± 0.6
G02	0.1 ± 0.7	2.2 ± 0.7	2.4 ± 0.9	2.3 ± 0.6
G03	-0.2 ± 0.8	2.2 ± 0.7	2.3 ± 0.8	2.4 ± 0.6
G04	3.1 ± 0.9	2.2 ± 0.9	2.7 ± 0.8	2.9 ± 0.6

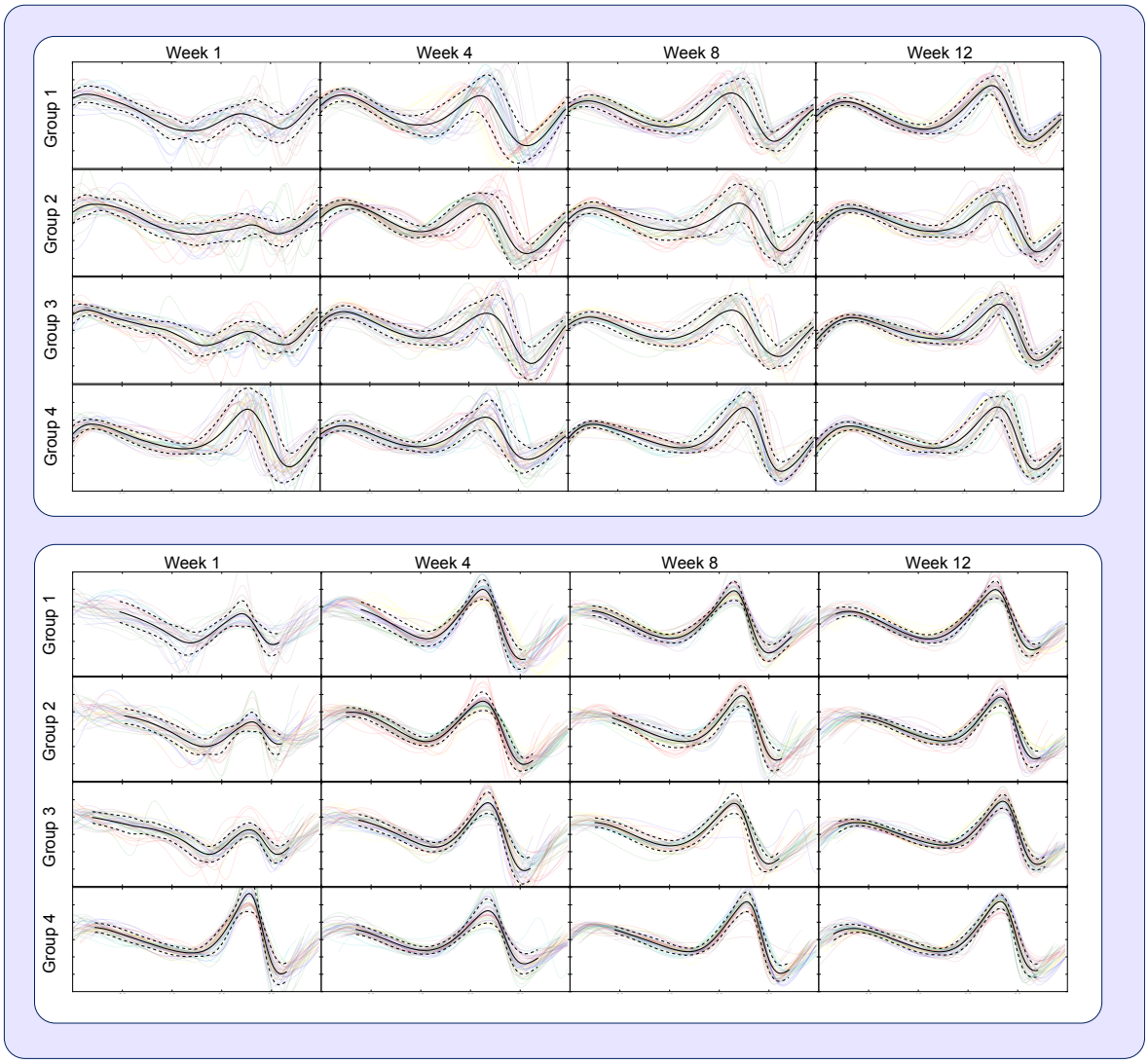

Figure 4.11: At the top: *Meanwaves* for each group and week before the alignment procedure. At the bottom: *Meanwaves* for each group and week after alignment.

Table 4.5: Percentage of waves inserted into the interval $[mean \pm std]$ of group 4, week 12.

[Mean - std ; Mean + std] %	W01	W04	W08	W12
G01	2.7%	58.3%	51.9%	61.7%
G02	0.0%	36.7%	40.9%	53.1%
G03	0.0%	40.4%	59.0%	56.0%
G04	58.7%	29.8%	62.5%	58.7%

The study of the rats' number inserted in the interval defined by $[mean \pm x \times std]$ of group sham's final week also supports that walking group is quicker to achieve the results of sham group. Table 4.5 shows the evolution over the weeks for x=1. Group 1 has a higher percentage of rats close to the sham's final week peak values, approaching those results earlier than other groups.

The peak value of ankle angular velocity was a key parameter for the detection of differences between groups. The results of this study suggest that mechanical load, specially active exercise, plays a role on functional recovery after peripheral nerve regeneration.

The variability of the gait cycles, indicated by the deviation area, decreased significantly with the alignment produced and the peak value of ankle angular velocity was accessed with more certainty. The wave-alignment techniques enabled a more reliable calculation of this parameter as well as a higher level estimation of the gait's *meanwave*.

Chapter 5

Applications

Parallel to the development of the algorithms described in this study, application projects were conceived which motivated the creation of our algorithms. In this chapter, these projects will be briefly described and it will be clarified how our tools are suitable answers for this type of work. We present these projects as applications of the algorithms produced; other possible applications will also be exposed in this chapter.

5.1 Case Study: Skiing Classification

5.1.1 Overview

This project was developed with researchers from the Norwegian School of Sport Sciences (*NIH - Norway*).

In this study, three male subjects volunteered to participate in a cross country (XC) skiing sprint. The XC-skiing is a winter sport in which participants use skis and poles to propel themselves across snow-covered terrain. It is a type of ski race which is divided into three parts: uphill, downhill and flat. Depending on the type of terrain, several different skiing techniques have to be used by the skiers [85]. The subjects were instrumented with five triaxial accelerometers (*xyzPLUX*). One was placed at the subject's lower back on the lumbar region, two were attached to each ski pole, below the handgrip, and the other two were attached to the heel of each ski-boot. A representation of this instrumentation appears in figure 5.1 [1].

[1] The signal acquisition procedure here described was fully done by H. Myklebust and J. Hallén, researchers at NIH [62].

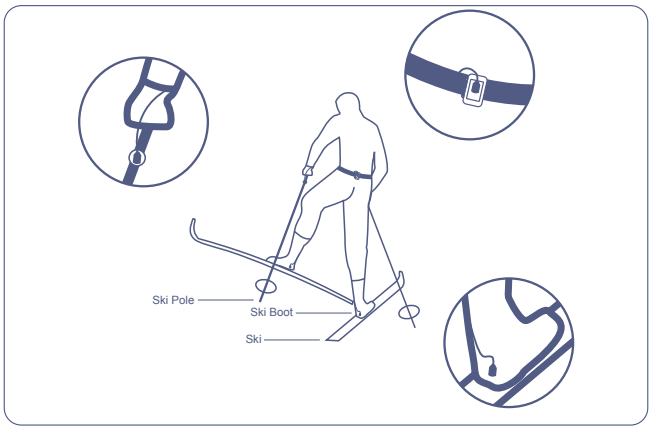

Figure 5.1: Representation of the subject skiing with the sensors and equipment.

The last four sensors were used as uniaxial accelerometers, as only the axis with inferior-superior orientation was connected to the acquiring system device (bioPLUX research). The aim of this study was to use accelerometers to extract relevant skiing information during XC-skiing, using video recordings for validation. Another goal was to develop a procedure to classify the different techniques used and detect the moments of technique transitions. This can help coaches and researchers to analyze the effect of different techniques in different tracks.

The techniques were divided into four groups:

- V_1 - One asymmetrical and asynchronous pole push for two leg strokes (one side);
- V_2 - Symmetrical and synchronous pole action for each leg stroke (both sides);
- V_3 - Symmetrical and synchronous pole push for two leg strokes;
- V_0 - Other techniques, including downhill, freeskate and turning techniques.

In the following sub-section, the procedure created to reach the defined goal will be briefly described and a comparison with an *autoMeanwave* procedure will also be made.

5.1.2 Threshold Method vs *autoMeanwave*

Figure 5.2 sets out the procedure done for both methods, exposing its similarities and its differences.

Both methods began with a pre-processing phase, in which the signals were low pass filtered and calibrated to g-units.

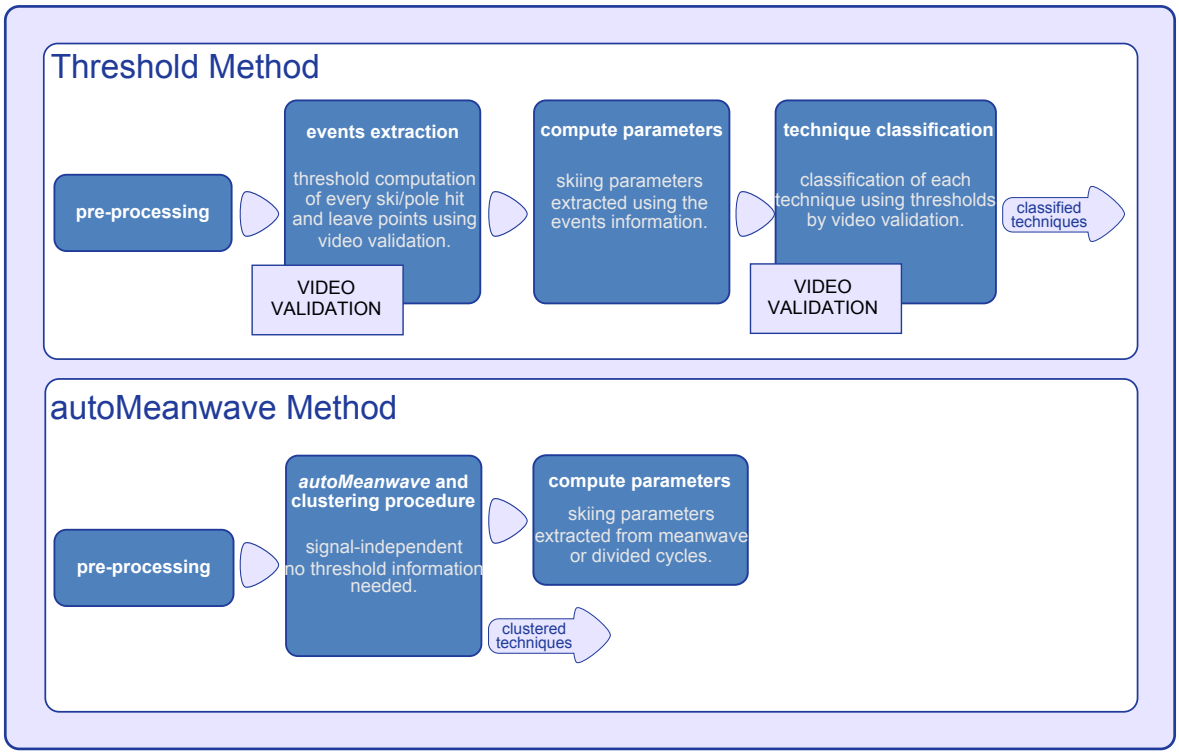

Figure 5.2: Schematics of both procedures for the skiing activity project.

For the threshold method, the next step was to extract the cycle events: the initiation (hit) and finalization (leave) ground contact points of each ski and pole. Algorithms were developed to compute the exact instant of these events. For that, an exhaustively analysis of the signal's behavior and also of its first and second derivative signals was necessary. By signal analysis and video examination, the optimal thresholds were found for the correct events extraction. With all instants in which the ski and the pole hit and leave the ground, it was possible to compute relevant skiing parameters - like the cycle time (CT), pushing time (PT), recovery time (RT), symmetry between each side (SYM) and overlap between right and left side (OLR).

- CT - Time spent in each cycle (the time between each hit point was considered);
- PT - Time spent with the ski or pole on the ground pushing the body forward (the time from a hit point to a leave point was considered);
- RT - Time spent to begin another cycle (CT minus PT was considered);
- SYM - Symmetry between left and right sides (left parameters minus right parameters);
- OLR - Time spent from each pole hit to the next left/right ski hit.

The OLR variable isn't an usually computed skiing parameter, but we notice that since each technique has different pole actions, the distance from each pole hit to the next right or left ski hit vary between techniques. That way, it was possible to detect which technique each pole hit represented and estimate the time points of the technique transitions. The OLR results were analyzed for all the subjects in detail and validated with video, to get the correct thresholds that separates and classifies the cycles correctly.

The expert-based classification procedure here described was previously published and presented at the BIOSTEC 2011 conference [62] [Appendix A].

Our *autoMeanwave* algorithm is a much suitable procedure, in terms of time consumption and signal independence. With the *autoMeanwave* procedure, the signal is only analyzed in the pre-processing phase, and then processed with no more information about the signal. Furthermore, the algorithm is signal-independent unlike the classifier made with the threshold method which may give other results for new subjects. The skiing parameters (CT, PT, RT and SYM) can also be computed directly from the resulting *meanwave* (if mean values wanted) or the resulting divided cycles (if concrete values wanted for each cycle).

One of the skiing signals here discussed was actually used to validate our *autoMeanwave* algorithm (section 4.1). A period in which the subject changed between V1 and V2 techniques (only two modes) was used and the clustering results for this period showed that our approach was efficient.

5.2 Case Study: Swimming Analysis

5.2.1 Overview

This project was performed in association with researchers from the Sport Sciences Higher School of Rio Maior (*ESDRM - Portugal*).

In this study, five male subjects volunteered to swim two sets of 25m with a breaststroke technique. In the first set the swimmers used a snorkel and for the second set they swam without snorkel. A snorkel is a tube used for breathing when the wearer's mouth and nose are submerged. It is often used for physiological and biomechanical analysis - some studies were made to analyze its reliability and possible mechanical

constraints [24]. The subjects were instrumented with two EMG sensors (*emgPLUX*) to collect the muscle activity of the *Biceps Brachii* (BB) and *Triceps Brachii* (TB) - see figure 5.3. The acquisition was made with a *bioPLUX research* [2].

The purpose of this study was to compare the average pattern of muscle activation in two situations: with and without the use of a snorkel in the breaststroke technique.

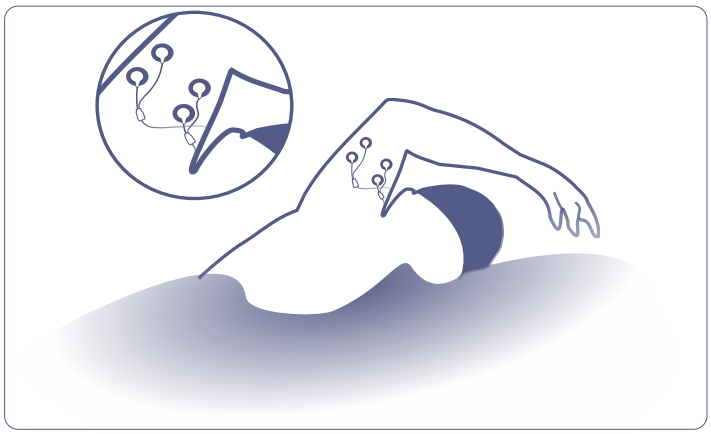

Figure 5.3: Representation of the sensors on the swimmer.

5.2.2 Threshold Method vs *autoMeanwave* Method

Figure 5.4 sets out the procedure done for both methods and shows its relations.

In this project, both basic threshold and automatic methods have the purpose to compute the *meanwaves* of two different situations and analyze its differences.

The first step for both procedures was the pre-processing of the EMG signals. All signals were centered at y-axis zero, rectified and smoothed with a low pass filter, to get the signal's envelope [83]. The *meanwave*'s manual computation was done using a threshold method. By signal's analysis, we concluded that the cycles should be separated by each cycle's minimum point. Those points were computed and the signal's cycles were divided and normalized in time, so that each cycle had exactly 100 samples. That way it was possible to directly compute the *meanwave*. Figure 5.5 shows one of the signals used, its cycles and events, and the resulting *meanwave*.

The *autoMeanwave* algorithm uses a procedure to compute the *meanwave*, so the estimation of the events, the division of each cycle and its normalization isn't necessary. Also to note that the threshold method might not work properly if applied to new data.

[2]The signal acquisition procedure here described was fully done by A. Conceição, D. Marinho, A. Costa, A. Silva and H. Louro, researchers at ESDRM [22].

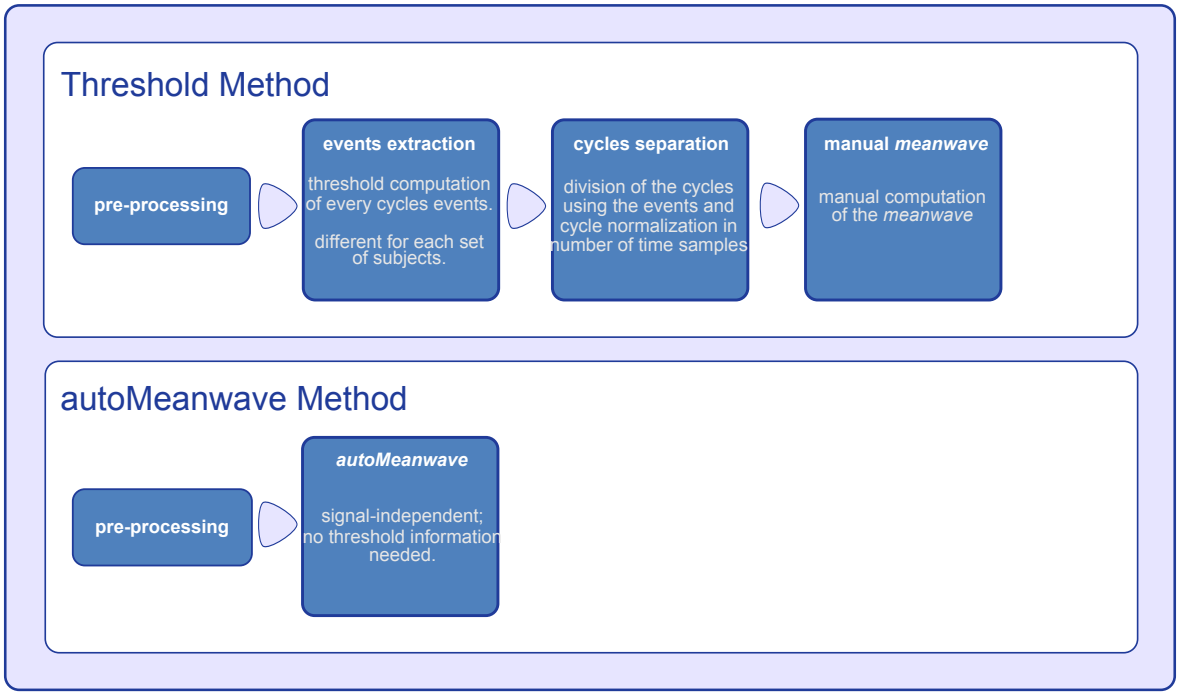

Figure 5.4: Schematics of both procedures for the swimming activity project.

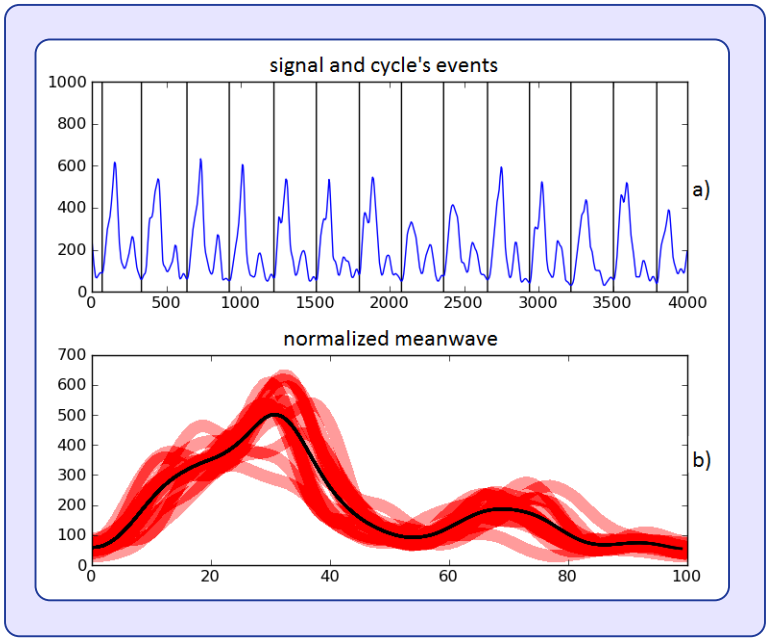

Figure 5.5: a) Example of a subject's *biceps brachii* EMG signal, with the events positions traced and b) resulting normalized *meanwave* by the threshold method.

After tracing the *meanwaves*, a comparison was made between each muscle (BB and TB) and each situation (with and without snorkel). The results demonstrated that the mean and the maximum activation (EMG) for the BB and TB muscles are higher with the use of snorkel. The activation time was also higher for the swimming situation with snorkel. Comparing between muscles, BB showed higher maximum and minimum activation values.

The resultant *meanwaves* and the information extracted were the same for both the basic threshold method and our signal-independent and less time consuming proposed algorithm.

This study, which at the time used the threshold method to compute the *meanwave*, was previously published and presented at the BMS 2010 conference [22] [Appendix A].

5.3 Case Study: Elderly Motion Analysis

5.3.1 Overview

This project was developed together with the research group "Neuromechanics of Human Movement" from the Faculty of Human Kinetics (*FMH - Portugal*).

This group is currently working in a research project called "Biomechanics of Locomotion in Elderly People. Relevant Variables for Risk of Fracture Reduction". This research aims to contribute to the study of fracture risk in elderly population, through the development of experimental tools to identify elders with propensity to fall, establishing prevention strategies for this population. One of the specific goals of this research work is the characterization of the elderly population with regard to the physical and functional fitness level. To study this parameter, some context tasks were defined and applied to the elderly population.

Two of the context tasks defined are described below:

- **Stand up and Go.** The subjects begin this task comfortably sitting on a chair, with their backs straight and both feet on the ground. When asked, they get up and walk to a mark at 2,44m, go around that mark and back to the chair, sitting down. The subjects are asked to perform this task as quickly as possible, without running.

- **Sit and Get Up.** The subjects begin this task comfortably sitting on a chair, with their backs straight and both feet on the ground. When asked, the subjects get up to a full straight position, and sit down back on the chair. The subjects are asked to perform this task many times as possible in 30s.

The researcher acts as an observer of the tasks, defining the start instant and timing the performance until the moment the subject sits on the chair, for the "Stand up and Go task", or until 30s if in the "Sit and Get Up" task. In this last task the researcher also counts the number of times the subject was able to sit and get up [3].

We were asked to include the use of accelerometers in this research, to validate the information given by the observer and to withdraw more relevant parameters from the data. Two triaxial (*xyzPLUX*) accelerometers were used: one located at the center of the chest, fixed by a band, and the other at the right hip, sticked to the *bioPLUX acquisition system* - figure 5.6.

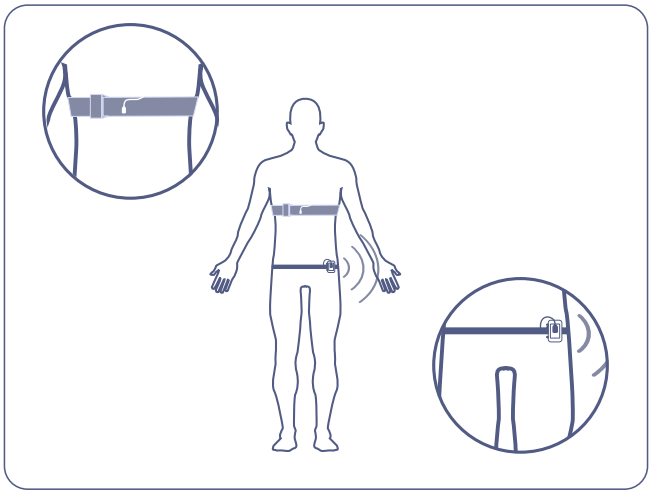

Figure 5.6: Representation of the sensors equipped on the subject.

5.3.2 Manual Method vs *autoMeanwave* Method

Figure 5.7 illustrates the procedure done with both manual and automatic methods in the analysis of the acquired signals.

Again, the first step is the pre-processing of the acquired signals. In this phase, we applied a calibration to g-units and a low pass filter to the signals.

[3] These methods were defined by F. Carnide, M. Machado, H. André, V. Moniz-Pereira and A. Veloso, researchers at FMH [56].

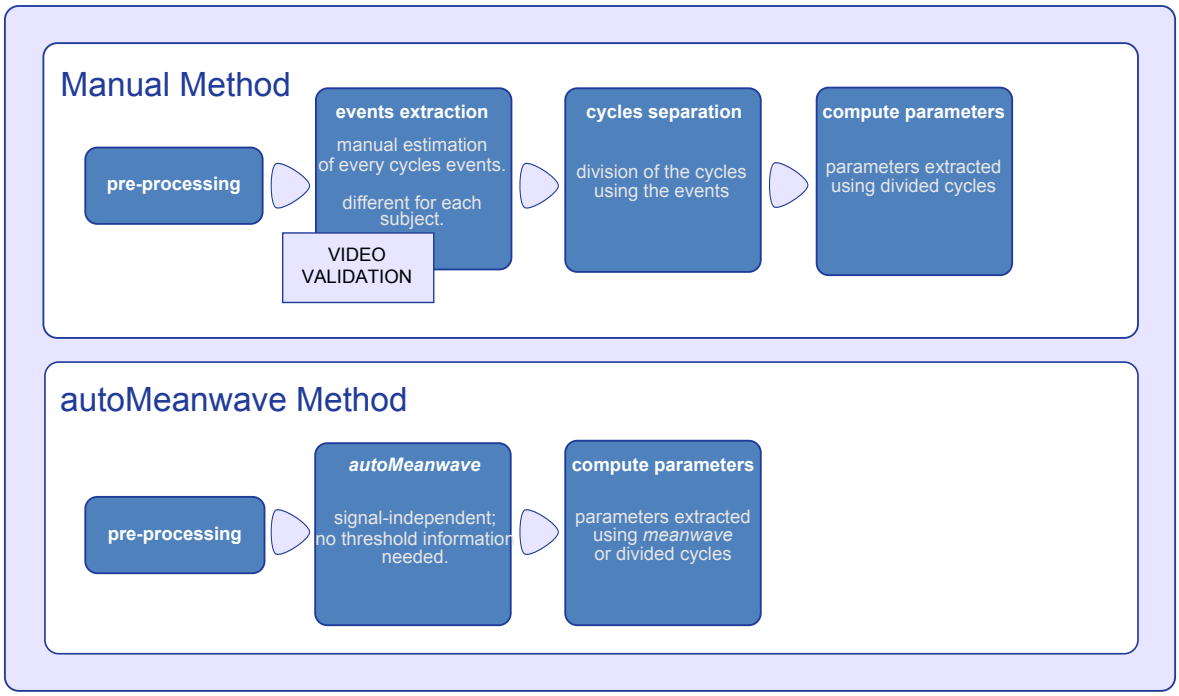

Figure 5.7: Schematics of both procedures for the elderly motion project.

We used the manual method described in figure 5.7 just for a few acquired signals, because as we will state next, it is an exhausting method for the user. In this method, the first step is to extract all the events for both tasks, and validate with video. For the first task, the events were defined as the beginning of a step. The instants when the subject sits on the chair were the defined events for the second task. After defining the events, the cycles can be divided and some useful parameters can be computed. As we were working with hundreds of signals, we used the *autoMeanwave* algorithm to separate each cycle automatically. For this amount of data, an automatic procedure is the best choice given that removes the necessity of time consuming data labeling.

After this, the parameters to compute can be extracted from the cycles (manual or automatic method) or from the *meanwave*. For the first task, the duration of each step was computed and also the cycle's mean duration and the total time spent in the task. For the second task, it was computed the number of times the subject sit and get up, the duration of each cycle and the cycles' mean duration. We also computed a trendline for the cycles' duration, to understand if the subjects could be divided into clusters: for example, if the subject tend to keep the same speed during the task, if it increases or decreases and if he is stable during the task or with many oscillations.

This information is relevant in the current research for a future correlation with the subjects' information about level of physical activity and propensity to fall.

5.4 Other Applications

Besides the applications referenced before, and the others already described in the previous chapter that were used to evaluate the algorithms (section 4.1 and 4.3), there are numerous more in which our work is adaptable and relevant.

Signal averaging algorithms are usually performed in ECG and EEG signals to improve the signal-to-noise ratio (SNR) and recover the low-amplitude potentials of the biological signals [16]. Concerning the ECG signals, a precise synchronization of heartbeats is essential and therefore several alignment methods have been proposed to optimize those averaging algorithms [47] [53]. In signal processing of EEG evoked potentials (EP), algorithms which incorporate alignment of the EP's in the averaging procedures have also been studied [94] [37].

This could be a general application of our algorithms in ECG and EEG signals. A brief study of the EEG evoked potentials which uses the *meanwave* concept to average the EEG stimuli response was developed and submitted in the IEEE Sensors 2001 conference [6]. The integration of the wave-alignment techniques in this study is also a future goal.

Another possible application of our *autoMeanwave* algorithm is its implementation at real time, by computing the *meanwave* of the first s seconds of the signal, and tracing the distance of each subsequent cycle to the initial *meanwave*. Considering the initial *meanwave* as the "normal" state, it will be possible to see unnatural changes over time (for example, the gait pattern of a subject that is off balance and beginning to fall or physiological abnormalities in signals such as ECG, BVP, respiration, etc.).

A similar approach can be used for data processing offline. We are now beginning a study in which the purpose is to find the muscle fatigue point in EMG signals of a subject performing intense exercise. The muscle fatigue is considered the incapacity to maintain the desirable level of force when performing a specific task [30]. By computing the *meanwave* for the first and last 30s of the signal and then the distance between the signal's cycles and each *meanwave*, it will be possible to see a progressive deviation of the waves from the initial *meanwave* and consequent approximation to the final. This analysis will possibly allow a good estimation of the fatigue point of the subject in that exercise.

Chapter 6

Conclusions

In this final chapter, the summary of the work developed and the general results and accomplishments of this thesis will be presented. This chapter outlines the contributions of our work, and discusses future goals and implications.

6.1 General Results

In the first chapter of this thesis we divided our study into three blocks which independently were tested, generating individual contributions. In the conclusion of this thesis we enumerate those contributions (see figure 6.1).

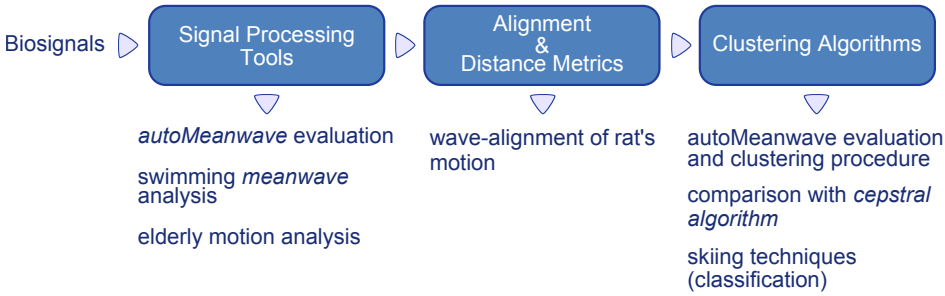

Figure 6.1: Independent tools developed for the thesis and its contributions.

In this dissertation, signal processing algorithms with applications in time series clustering were developed. We presented the concept of *meanwave* and described the procedure for its computation in cyclic signals acquired continuously. For an enhancement of the *meanwave*'s accuracy, wave-alignment techniques were also designed.

An evaluation and validation of our algorithms was included in this study. Biosignals were acquired and processed with the *autoMeanwave* algorithm to test its efficiency

as a time series clustering procedure. Our algorithm proves to be an effective method in the detection of changes in the signal's morphology, achieving 99.3% of clustering accuracy. We compared our clustering procedure with another method referenced in literature, the *cepstral algorithm*, which presented the best results to the date for time series data. We obtained better results using the same dataset of acquired data. The *autoMeanwave* procedure is also much more appropriate for the analysis of continuous signals, as it automatically separates the signals' cycles and doesn't need different inputs for different signal's modes, unlike the *cepstral algorithm*. A separate evaluation for the designed wave-alignment techniques was also made. In the context data in which the alignment was applied, the deviation area was reduced by 32.8%. This wave-alignment technique also has a substantial importance in the extraction of relevant information from a notable point of the cycles.

In this dissertation, we also presented a set of applications suitable to our algorithms. We showed that our contribution reduces the need of expert intervention on the signal processing and classification, also producing better results comparatively to other approaches.

The algorithms can be applied to a continuous cyclic time series, with no more than a small pre-processing phase, capturing the signal's behavior. The fact that this approach doesn't require any prior information and its good performance in different situations makes it a powerful tool for biosignals analysis and classification.

During the thesis development, a few number of papers were published, three of which are direct applications of the research presented in this dissertation and are exposed in Appendix B.

The work produced in this thesis was developed at *PLUX - Wireless Biosignals*, integrated at its Research and Development (R&D) department. Some work was developed in collaboration with other institutions, so a few visits to those installations were necessary to discuss important aspects of the projects. The good conditions and surrounding environment led to a healthy work ethic and an extremely enriching experience.

6.2 Future Work

The research here presented leaves a few aspects opened, which we intend to explore more in the future. Some ideas for the optimization of our algorithms will be described next.

Additional validation: Our algorithms were tested in a reduced set of subjects and scenarios. It is our will to extend our tests to a wider range of biosignals and acquisition scenarios and also use more subjects for each scenario. This would enable the study of the algorithms's performance for larger populations.

More than two modes: A major advance in this study will be to test the response of our algorithm in signals that have more than two modes. A preliminary test with a few signals with three and four modes has already been done, and the results achieved were highly satisfactory. It's also our intention to automatically perceive the number of clusters present in the signals and input that estimation into the k-means algorithm.

Local f_0 detection: The local detection of the fundamental frequency is a future goal for the optimization of the *autoMeanwave* algorithm. We intend to perceive when there's a major variation of fundamental frequency and make our algorithm adapt its behavior according to that variation.

Noise immunity test: We intend to perform a noise immunity test, by adding different types of noise and seeing the differences in the responses of our algorithms to each test. That will guarantee the efficiency of our work in signals with different noise influences.

Distance to *meanwave*: One of the outputs of our *autoMeanwave* algorithm is a signal composed with the distance of each signal's cycle to the resulting *meanwave*. This parameter wasn't used in our research, and we intend to study more its utility, for example, test the use of this distance metric to cluster the signals.

Rejection class: We want to introduce an automatically perception of cycles which are too distant from its cluster and assign those cycles to a new "rejection class". This will reduce the number of errors due to odd cycles, in particular the mode's transition cycles.

Multimodal algorithm: We have the intention to create a multimodal algorithm, which can receive and process more than one signal at the same time and with the same treatment. This could be useful, for example, if we want to use the three axis of

a triaxial accelerometer, a BVP with an ECG signal, or to conciliate the information of synchronous acquired signals.

New applications: It is our aim to use our algorithms in different application projects, to continuously test its efficiency and its limits and also increase the number of tools in our work by implementing new solutions to answer the needs of each project.

The continuously need to obtain more information, with more efficiency, more quickly and with less intervention from an expert has led to a growing application of signal processing techniques applied to biomedical data. The biosignal analysis and processing is a promising area with huge potential in medicine, sports and research. Being able to contribute with new and interesting techniques for the advance of this field was an extremely enriching and challenging experience.

Bibliography

[1] M. Akay. *Wiley Enyiclopedia of Biomedical Engineering*. John Wiley and Sons, 2006.

[2] S. Amado, J. Rodrigues, A. Luís, P. A. da Silva, M. Vieira, A. Gartner, M. Simões, A. Veloso, M. Fornaro, S. Raimondo, A. Varejao, S. Geuna, and A. Maurício. Effects of collagen membranes enriched with in vitro-differentiated n1e-115 cells on rat sciatic nerve regeneration after end-to-end repair. *Journal of Neuroengineering and Rehabilitation*, 2010.

[3] C. Amma, D. Gehrig, and T. Schultz. Airwriting recognition using wearable motion sensors. In *Proceedings of the 1st Augmented Human International Conference*, 2010.

[4] E. Andersson, M. Supej, O. Sandbakk, B. Sperlich, T. Stoggl, and H. Holmberg. Analysis of sprint cross-country skiing using a differential global navigation satellite system. *European Journal on Applied Physiology*, 2010.

[5] G. Antoniol, V. Rollo, and G. Venturi. Linear predictive coding and cepstrum coefficients for mining time variant information from software repositories. *MSR 2005: International Workshop on Mining Software Repositories*, 2005.

[6] T. Araújo, N. Nunes, C. Quintão, and H. Gamboa. Development of a localized electroencephalography sensor: application in the detection of evoked potentials. In *submitted to the 5th International ICST Conference on Pervasive Computing Technologies for Healthcare*, Dublin, Ireland, 2011.

[7] J. Baek, G. Lee, W. Park, and B. Yun. Accelerometer signal processing for user activity detection. *Knowledge-Based Intelligent Information & Engineering Systems, Vol. 3215*, pages 610–617, 2004.

[8] A. Bagnall and G. Janacek. Clustering time series from arma models with clipped data. In *Procedings of KDD '04, the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, Seattle, USA, Aug. 2004.

[9] M. Basseville and I. Nikiforov. *Detection of Abrupt Changes: Theory and Applications*. Prentice-Hall Inc., 1993.

[10] J. Bemmel and M. Musen. *Handbook of medical informatics*. 2nd Edition, Springer, 1997.

[11] J. Ben-Arie, Z. Wang, P. Pandit, and S. Rajaram. Human activity recognition using multidimensional indexing. *IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 24, no. 8*, pages 1091–1104, 2002.

[12] J. Boets, K. Cock, M. Espinoza, and B. Moor. Clustering time series, subspace identification and cepstral distances. *Communications in Information and Systems, Vol. 5, No. 1*, pages 69–96, 2005.

[13] J. Bradbury. Linear predictive coding. Florida Institute of Technology - http://my.fit.edu/∼vkepuska/ece5525/lpc_paper.pdf, 2000.

[14] F. Buttussi and L. Chittaro. Smarter phones for healthier lifestyles: An adaptive fitness game. *IEEE Pervasive Computing, Vol 9, Issue 4*, pages 51–57, 2010.

[15] B. Carter, T. Beaty, G. Steinberg, B. Childs, and P. Walsh. Mendelian inheritance of familial prostate cancer. *Proceedings of the National Academy of Sciences of the United States of America, vol 89*, pages 3367–3371, 1992.

[16] F. Censi, G. Calcagnini, M. D'Alessandro, M. Triventi, and P. Bartolini. Comparison of alignment algorithms for p-wave coherent averaging. *IEEE Transactions on Biomedical Engineering, Vol. 38, No.6*, pages 571–579, 1991.

[17] S. Cerutti, G. Magenes, and P. Bonato. eds, special issue on "smart wearable devices for human health and protection". *IEEE Trans TIT-B,14(3):691-3*, 2010.

[18] H. Chang and J. Moura. Biomedical signal processing. *Biomedical Engineering and Design Handbook, 2nd Edition, Vol 1, McGraw Hill.*, pages 559–579, 2010.

[19] H. Chang, J. Moura, Y. Wu, and C. Ho. Automatic detection of regional heart rejection in uspio-enhanced mri. *IEEE Transactions on Medical Imaging, vol 27, Issue 8*, pages 1095–1106, 2008.

[20] E. Ciaccio, S. Dunn, and M. Akay. Biosignal pattern recognition and interpretation systems - part 1 of 4: Fundamental concepts. *IEEE Engineering in Medicine and Biology*, pages 89–97, 1993.

[21] E. Clancy, E. Morin, and R. Merletti. Sampling, noise-reduction and amplitude estimation issues in surface electromyography. *Journal of Electromyography and Kinesiology, vol. 12*, pages 1–16, 2002.

[22] A. Conceição, H. Gamboa, S. Palma, T. Araújo, N. Nunes, D. Marinho, A. Costa, A. Silva, and H. Louro. Comparison between the standard average muscle activation with the use of snorkel and without snorkel in breaskstroke techinique. In *XIth International Symposium for Biomechanics and Medicine in Swimming (BMS 2010)*, Oslo, June 2010.

[23] M. Corduas and D. Piccolo. Time series clustering and classification by the autoregressive metric. *Computational Statistics & Data Analysis, Vol 52*, pages 1860–1872, 2008.

[24] M. Costa, A. Reis, V. Reis, A. Silva, N. Garrido, H. L. D. Marinho, C. Baldari, and T. Barbosa. Constraint caused by mechanical valve aquatrainer associated with system oxymetry direct ($k4b^2$) in breakstroke kinematic. In *Procedings of the 3^{r}d National Congress of Biomechanics. M. A. Vaz et al. (Eds)*, Bragança, Portugal, 2009.

[25] D. Cournapeau. Scikits talkbox documentation. http://www.ar.media.kyoto-u.ac.jp/members/david/softwares/talkbox/talkbox_doc/index.html, 2008.

[26] S. Coyle, D. Morris, K.-T. Lau, N. Moyna, and D. Diamond. Textile-based wearable sensors for assisting sports performance. In *In Proceedings of BSN 2009 - Body sensor networks*, Berkeley, CA, USA, 2009.

[27] T. S. D. Heger, F. Putze. Online workload recognition from EEG data during cognitive tests and human-computer interaction. In *in Proceedings of the 33rd Annual German Conference on Artificial Intelligence 2010, KI 2010*, Karlsruhe, Germany, 2010.

[28] N. Davies, E. Mynatt, and I. Siio. Watchme: communication and awareness between members of a closely-knit group. *in In Ubicomp, Vol. 3205*, pages 214–231, 2004.

[29] S. Elavarasi, J. Akilandeswari, and B. Sathiyabhama. A survey on partition clustering algorithms. *International Journal of Enterprise Computing and Business Systems, Vol1, Issue 1*, 2011.

[30] I. Freitas. *Fatigue detection in EMG signals.* MSc. dissertation, Technical University of Lisbon, Nov. 2008.

[31] A. Fridlung, G. Schwartz, and S. Fowler. Pattern recognition of self-reported emotional state from multiple-site facial EMG activity during affective imagery. *Society for Psychophysiological Research, Vol. 21, No. 6*, 2007.

[32] S. Furui. *Digital Speech Processing, Synthesis, and Recognition.* Marcel Deckker, Inc., 1989.

[33] D. Gerhard. Pitch extraction and fundamental frequency: History and current techniques. *Technical Report TR-CS 2003-06*, 2003.

[34] Z. Ghahramani. Unsupervised learning. Gatsby Computational Neuroscience Unit, University College London, UK, 2004.

[35] H. Gholam-Hosseini, H. Nazeran, and K. Reynolds. ECG noise cancellation using digital filters. In *Proceedings of International Conference on Bioelectromagnetism*, pages 151–152, Melbourne, Australia, Feb. 1998.

[36] A. Gordon. *Classification.* Chapman-Hall, 1999.

[37] L. Gupta, D. Molfese, R. Tammana, and P. Simos. Nonlinear alignment and averaging for estimating the evoked potential. *IEEE Transactions on Biomedical Engineering, Vol. 43, No.4*, pages 1–8, 1996.

[38] F. Gustafsson. *Adaptive Filtering and Change Detection.* John Wiley & Sons Inc., 2000.

[39] L. Hanson. Using numerical python masked arrays. Bureau of Meteorology Research Centre, 2007.

[40] N. Hazarika, A. Tsoi, and A. Sergejew. Nonlinear considerations in EEG signal classification. *IEEE Transactions on Signal Processing, vol 45*, pages 829–836, 1997.

[41] I. Hlimonenko, K. Meigas, and R. Vahisalu. Waveform analysis of peripheral pulse wave detected in the fingertip with photoplethysmograph. *Measurement Science Review, Vol. 3, Section 2*, pages 49–52, 2003.

[42] Y. Hong, I. Kim, S. Ahn, and H.-G. Kim. Activity recognition on an accelerometer embedded mobile phone with varying positions and orientations. *Simulation Modelling Practive and Theory, Vol 18, Issue 4*, pages 446–455, 2010.

[43] J. Hunter and D. Dale. The matplotlib user′s guide. http://matplotlib.sourceforge.net/tutorial.html.

[44] D. Huynh. *Human Activity Recognition with Wearable Sensors*. PhD. dissertation, Technische Universitat Darmstadt, Aug. 2008.

[45] M. Ibrahimy. Biomedical signal processing and applications. In *Procedings of the International Conference on Industrial Engineering and Operations Management*, Dhaka, Bangladesh, Jan. 2010.

[46] A. Jain, M. Murty, and P. Flynn. Data clustering: A review. *ACM Computing Surveys, Vol. 31, No. 3*, 1999.

[47] R. Jané, H. Rix, P. Caminal, and P. Laguna. Alignment methods for averaging of high resolution cardiac signals: A comparative study of performance. *Computers in Cardiology, 2006*, pages 921–924, 2006.

[48] M. Janke, M. Wand, and T. Schultz. Impact of lack of acoustic feedback in EMG-based silent speech recognition. In *11th Annual Conference of the International Speech Communication Association (Interspeech)*, Makuhari, Japan, 2010.

[49] K. Kalpakis, D. Gada, and V. Puttagunta. Distance measures for effective clustering of arima time-series. In *Procedings of the 2001 IEEE International Conference on Data Mining*, pages 273–280, 2001.

[50] W. Kang, J. Shiu, C. Cheng, J. Lai, H. Tsao, and T.-S. Kuo. The application of cepstral coefficients and maximum likelihood method in EMG pattern recognition. *IEEE Transactions on Biomedical Engineering, Vol 42, No 8*, pages 777–785, 1995.

[51] R. Kashyap and A. Rao. *Dynamic stochastic models from empirical data*. Academic Press, Inc, 1976.

[52] E. Keogh, J. Lin, and W. Truppel. Clustering of time series subsequences is meaningless: Implications for past and future research. In *3rd IEEE International Conference on Data Mining*, Melbourne, FL, USA, 2003.

[53] E. Laciar, R. Jané, and D. Brooks. Improved alignment method for noisy high-resolution ECG and holter records using multiscale cross-correlation. *IEEE Transactions on Biomedical Engineering, Vol. 50, No.3*, pages 344–353, 2003.

[54] C. Levkov, G. Mihov, R. Ivanov, I. Daskalov, I. Christov, and I. Dotsinsky. Removal of power-line interference from the ECG: a review of the subtraction procedure. *BioMedical Engineering OnLine, vol. 4*, pages 1–8, 2005.

[55] W. Liao. Clustering of time series data - a survey. *Pattern Recognition, Vol 38*, pages 1857–1874, 2005.

[56] M. Machado, F. Carnide, V. Moniz-Pereira, H. André, and A. Veloso. The role of physical activity and functional fitness on perceived health in aging. *Medicine & Science in Sports & Exercise, Vol 42, Issue 5*, pages 49–50, 2010.

[57] D. Martins, M. Mattos, P. Simões, C. Cechinel, and J. Bettiol. Aplicação do algoritmo k-means em dados de prevalência da asma e rinite em escolares. In *Proocedings of XI Brasilian Congress on Health's Informatics*, Brasil, 2008.

[58] R. Martins, J. Medeiros, S. Palma, H. Gamboa, and M. Reis. Development of a blood volume pulse sensor to measure heart rate variability. In *Procedings of IBERSENSOR 2010*, Lisbon, Portugal, Nov.

[59] R. Merletti and P. Parker. *Electromyography: Physiology, Engineering, and Non-invasive Applications*. IEEE Press Series in Biomedical Engineering, 2004.

[60] J. Millman. *Microelectronics Digital and Analog Circuits and Systems*. McGraw-Hill Book Company, 1979.

[61] E. Miluzzo, N. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S. Eisenman, X. Zheng, and A. Campbell. Sensing meets mobile social networks: The design, implementation and evaluation of the cenceme application. In *in Proceedings of the International Conference on Embedded Networked Sensor Systems (SenSys)*, pages 337–350, 2008.

[62] H. Myklebust, N. Nunes, J. Hallén, and H. Gamboa. Morphological analysis of acceleration signals in cross-country skiing - information extraction and technique transitions detection. In *Procedings of the 4^{th} International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2011)*, Rome, Jan. 2011.

[63] C. Nichols, T. Myckatyn, S. Rickman, I. Fox, T. Hadlock, and S. Mackinnon. Choosing the correct functional assay: A comprehensive assessment of functional tests in the rat. *Behav. Brain Res., 163*, pages 143–158, 2005.

[64] N. Nunes, T. Araújo, and H. Gamboa. Two-modes cyclic biosignal clustering based on time series analysis. In *Procedings of the 4^{th} International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2011)*, Rome, Jan. 2011.

[65] T. Oliphant. *SciPy Tutorial*. http://www.tau.ac.il/∼kineret/amit/scipy_tutorial, 2004.

[66] T. Oliphant. *Guide to Numpy*. Tregol Publishing, 2006.

[67] Online. Rat skeleton. www.carolina.com/text/pdf/owls/ratskeleton.pdf.

[68] Online. A tutorial on clustering algorithms - clustering: An introduction. http://home.dei.polimi.it/matteucc/Clustering/tutorial_html/.

[69] Online. Using a digitizer for time-domain measurements. http://zone.ni.com/devzone/cda/tut/p/id/10669, 2009.

[70] Online. Telecomunication. http://boim-the-rockmonster.blogspot.com/2010/05/telecommunication_27.html, 2010.

[71] Online. Opensignals. http://www.opensignals.net/, 2011.

[72] Online. Physionet. http://www.physionet.org/physiobank/database/, 2011.

[73] Online. PLUX - wireless biosignals. http://www.plux.info/, 2011.

[74] A. Oppenheim, R. Schafer, and J. R. Buck. *Discrete-Time Signal Processing*. Prentice Hall; 2nd edition, 1999.

[75] D. O'Shaughnessy. *Speech Communication: Human and Machine*. Addison-Wesley Publishing Company, 1992.

[76] J. Pansiot, D. Stoyanov, D. McIlwraith, B. Lo, , and G.-Z. Yang. Ambient and wearable sensor fusion for activity recognition in healthcare monitoring systems. In *In IFMBE proceedings of the 4th International Workshop on Wearable and Implantable Body Sensor Networks 2007 (BSN)*, Aachen, Germany, 2007.

[77] T. H. Park. *Introduction to Digital Signal Processing: Computer Musically Speaking*. World Scientific Publishing Co. Pte. Ltd., 2010.

[78] T. Penzel, K. Kesper, and H. F. Becker. Biosignal monitoring and recording. *Information Technology Solutions for Healthcare*, pages 288–301, 2006.

[79] E. Peper, R.Harvey, M. Lin, H.Tylova, and D. Moss. Is there more to blood volume pulse than heart rate variability, respiratory sinus arrhythmia, and cardiorespiratory synchrony? *Biofeedback Special Topics, Vol. 35*, pages 54–61, 2007.

[80] R. Quiroga. Spike sorting. Scholarpedia, 2(12):3583, 2007.

[81] L. Rabiner and B. Juang. *Fundamentals of Speech Recognition*. Prentice Hall Signal Processing Series, 1993.

[82] H. Ramos, F. Coito, R. Silva, and M. Ortigueira. Análise de sinais em engenharia biomédica. FCT-UNL, 2009.

[83] G. Robertson. Electromyography: Processing. Biomechanics Laboratory, School of Human Kinetics, University of Ottawa, Canada, 2007.

[84] G. Robertson, G. Caldwell, J. Hamill, G. Kamen, and S. Whittlesey. *Research Methods in Biomechanics*. Human Kinetics, 1st edition, 2004.

[85] H. Rusko. *Cross Country Skiing*. Blackwell Science Ltd., 2003.

[86] A. Savvides, V. Promponas, and K. Fokianos. Clustering of biological time series by cepstral coefficients based distances. *Pattern Recognition, Vol 41*, pages 2398–2412, 2008.

[87] K. Schaaff and T. Schultz. Towards an EEG-based emotion recognizer for humanoid robots. In *18th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, Toyama, Japan, 2009.

[88] A. Sedra and K. Smith. *Microelectronic Circuits*. Oxford University Press, Inc., 2004.

[89] M. Singh, K. Kaur, and B. Singh. Cluster algorithm for genetic diversity. *World Academy of Science, Engineering and Technology 42*, pages 453–457, 2008.

[90] J. Smith, K. Fishkin, B. Jiang, A. Mamishev, M. Philipose, A. Rea, S. Roy, and K. Sundara-Rajan. Rfid-based techniques for human-activity detection. *Communications of the ACM, Vol. 48, No. 9*, pages 39–44, 2005.

[91] J. Sotelo. *Biosignal analysis for cardiac arrhythmia detection using non-supervised techniques*. PhD dissertation, Faculty of Engineering and Architecture of Universidad Nacional de Colombia, 2010.

[92] P. Souza. Statistical tests and distance measures for LPC coefficients. *IEEE Transactions on Acoustics, Speech and Signal Processing, Vol 25, Issue 6*, pages 554–559, 1977.

[93] F. Spitzer. *Principles of Random Walk*. Springer, 2nd edition, 2001.

[94] M. Stefanelli, F. D'Alvano, J. Regidor, and T. Pérez. Single stimulus evoked potential estimation using adaptive noise cancelling with reference alignment. *Rev. Tec. Ing. Univ. Zulia, Vol. 21, No.2*, pages 124–130, 1998.

[95] L. Sun, D. Zhang, B. Li, B. Guo, and S. Li. Mobile health monitoring system based on activity recognition using accelerometer. In *Proceedings of the 7th international conference on Ubiquitous intelligence and computing*, 2010.

[96] M. Tarvainen. *Estimation Methods for Nonstationary Biosignals*. PhD dissertation, Faculty of Natural and Environmental Sciences of the University of Kuopio, June 2004.

[97] F. Theis and A. Meyer-Base. *Biomedical Signal Analysis: Contemporary Methods and Applications*. The MIT Press, 2010.

[98] G. van Rossum. *The Python Language Reference Manual*. Network Theory Ltd., 2003.

[99] J. Webster. *The Measurement, Instrumentation and Sensors Handbook*. CRC Press, 1st edition, 1998.

[100] R. Wotiz, S. Chang, B. Cole, B. Toba, J. Lin, S. Nawab, and C. Luca. Muscles alive: Decomposition of highly unpredictable real-life EMG signals. http://www.bu.edu/iss/research-projects/muscles-alive, 2010.

[101] X. Wu, V. Kumar, J. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. McLachlan, A. Ng, B. Liu, P. Yu, Z. Zhou, M. Steinbach, D. Hand, and D. Steinberg. Top 10 algorithms in data mining. *Knowledge Information Systems, 14:1-37*, 2007.

[102] Y. Xiong and D. Yeung. Mixtures for arma models for model-based time series clustering. In *Procedings of the IEEE International Conference on Data Mining*, pages 717–720, 2002.

[103] R. Xu and D. Wunsch. Survey of clustering algorithms. *IEEE Transactions on Neural Networks, Vol. 16, No. 3*, 2005.

[104] R. Xu and D. Wunsch. *Clustering*. Wiley-IEEE Press, 2008.

[105] O. Zaiane. Chapter8: Data clustering. [Online Presentation] http://webdocs.cs.ualberta.ca/∼zaiane/courses/cmput690/slides/Chapter8/index.html, 1999.

Appendix A

Publications

During the development of this project, three articles were submitted and presented in international conferences. The first publication is entitled *'Two-Modes Cyclic Biosignal Clustering based on Time Series Analysis'* and was presented in BIOSIGNALS 2011 which is a co-located conference of the *'4th International Joint Conference on Biomedical Engineering Systems and Technologies'* (BIOSTEC 2011), held in Rome in January 2011. This paper was also selected to be included in the series "Communications in Computer and Information Science" (CCIS) published by Springer-Verlag. This book will include the updated and extended versions of only a short list of selected papers from BIOSTEC 2011. The second paper was also presented in the BIOSIGNALS 2011 conference, and is entitled *'Morphological Analysis of Acceleration Signals in Cross-Country Skiing'*. The third publication has the title *'Comparison Between the Standard Average Muscle Activation with the use of Snorkel and without Snorkel in Breakstroke Technique'* and was held in Oslo in June 2010, at the *'XI th International Symposium for Biomechanics and Medicine in Swimming'* (BMS2010).

A.1 *Biosignals 2011*

Two-Modes Cyclic Biosignal Clustering based on Time Series Analysis

TWO-MODES CYCLIC BIOSIGNAL CLUSTERING BASED ON TIME SERIES ANALYSIS

Neuza Nunes, Tiago Araújo
Physics Department, FCT-UNL, Lisbon, Portugal
neuzanunes@gmail.com, tiago_sergio87@hotmail.com

Hugo Gamboa
Physics Department, FCT-UNL, Lisbon, Portugal
PLUX – Wireless Biosignals, Lisbon, Portugal
h.gamboa@fct.unl.pt, hgamboa@plux.info

Keywords: Biosignals, waves, unsupervised learning, clustering, data mining, signal-processing.

Abstract: In this paper we introduce an unsupervised learning algorithm which distinguishes two different modes in a cyclic signal. We also present the concept of "*mean wave"* which averages all signal waves aligned in a notable point (n^{th} zero derivative). With that information the signal's morphology is captured. The clustering mechanism is based on the information collected with the *mean wave* approach using a k-means algorithm. The algorithm produced is signal-independent, and therefore can be applied to any type of signal providing it is a cyclic signal that has no major changes in the fundamental frequency. To test the effectiveness of the proposed method, we acquired several biosignals (accelerometry, electromyography and blood volume pressure signals) in context tasks performed by the subjects with two distinct modes in each. The algorithm successfully separates the two modes with 99.3% of efficiency. The fact that this approach doesn't require any prior information and the preliminary good classification performance makes this algorithm a powerful tool for biosignals analysis and classification.

1 INTRODUCTION

Human-activity tracking techniques focus on direct observation of people and their behavior. This could be done, as an example, with cameras (Jezekiel Ben-Arie, 2002), accelerometers to track human motion (Jonghun Baek et al., 2004), or contact switches to compute facial expressions with the electromyography patterns (Joshua R. Smith, 2005) (Alan J. Fridlund, 2007).

In this work, we acquired several cyclic biosignals – such as accelerometry (ACC), electromyography (EMG) and blood volume pressure (BVP) signals – from subjects performing some context tasks. An unsupervised learning algorithm, which is capable to distinguish two different modes in the same acquired signal, was also developed.

The developed algorithm follows an unsupervised learning approach, as it doesn't require any prior information (Zoubin Ghahramani, 2004). We use the k-means cluster algorithm due to its efficiency and effectiveness (Xindong Wu, 2007).

As a clustering method, our algorithm is signal-independent as it doesn't use specific information about the signals. Although our algorithm is signal-independent, the signals used must be cyclic, with only two distinctive modes and a small variation of fundamental frequency between those modes.

Warren Liao (2005) presents a survey on time series data clustering, exposing past researches on the subject. He organizes the works in three groups: whether they work directly with the raw data, indirectly with features extracted or indirectly with models built from the raw data. We created a different algorithm as we intended to work with single signals with different modes or activities in it, and the previous studies uses various signals each one distinct with only one mode or activity.

A more resemble approach, as the clustering is based on the similarity of wave shapes presented in a single time series data, is the work of Dr. Rodrigo Quiroga (2007) with spike sorting. However, as the

neuron activity is not periodic, the spikes are detected with a threshold and the clustering procedure uses features extracted from those parts of the signal.

We present the concept of "*mean wave"* which averages all signal waves aligned in a notable point, that we call triggering point, such as maximum, minimum, zero or inflexion point. Our algorithm automatically separates each signal's cycle and computes the *mean wave*, with center on the triggering point of each cycle. With that information the signal's morphology is captured. Our clustering algorithm uses a distance metric, gathered from the information of the signal's cycles, to separate the two modes of the entire signal.

As our *mean wave* approach effectively captures the morphology of a signal, it can be useful in several areas – as a clustering basis or just for signal analysis.

In the following section the signal acquisition methodology is presented. In section 3 we expose the signal processing detailing all algorithms steps. Finally in sections 4 and 5 we detail and discuss our results and algorithm performance, concluding the work.

2 METHODS

2.1 Acquisition system and sensors

To acquire the biosignals necessary to this study we used a surface electromyography (EMG) sensor, *emgPLUX*, a triaxial accelerometer (ACC), *xyzPLUX*, and a finger blood volume pressure (BVP) sensor, *bvpPLUX*.

For the signal's analog to digital conversion and bluetooth transmission to the computer we used a wireless signal acquisition system, bioPLUX research, which has 12 bit ADC and a sampling frequency of 1000 Hz. In the acquisitions with accelerometers only the axis with inferior-superior direction was connected to the bioPLUX (bioPLUX Research Manual, 2010).

2.2 Data acquisition and data format

Several tasks were designed and executed in order to acquire signals with two distinct modes.

We conceived a synthetic digital signal and collected signals from four different activities scenarios with the accelerometer sensor, and one for each EMG and BVP sensors.

2.2.1 Synthetic signal

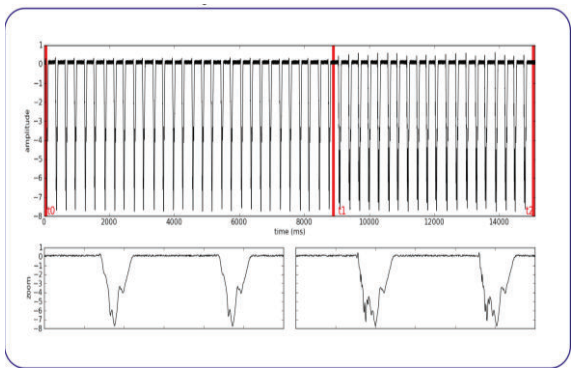

Figure 1 a): Synthetic signal with identical waves from t0 to t1 and from t1 to t2; b): corresponding zoomed waves.

To test our algorithm, a synthetic cycle was created using a low-pass filtered random walk (of 100 samples), with a moving average smoothing window of 10% of signal's length, and multiplying it by a hanning window. That cycle was repeated 30 times for the first mode, so all the cycles were identical. After a small break on the signal, the cycle was repeated 20 more times, but with an identical small change of 40 samples in all waves, creating a second mode. These two modes created the synthetic wave represented in Figure 1.

2.2.2 Walking and running (ACC)

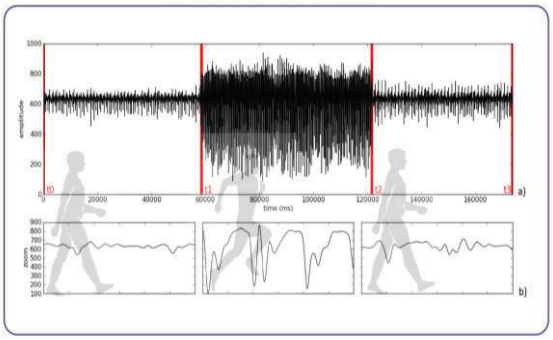

Figure 2 a): Acceleration signal of walking (t0 to t1 and t2 to t3) and running (t1 to t2); b): corresponding zoomed waves.

With an accelerometer located at the right hip and oriented so the y axis of the accelerometer (the only connected to the bioPLUX) was pointing upward,

the subjects performed a task of walking and running non-stop (on a large circle drawn on the floor).

The subjects walked for about 1 minute at a slow speed, then spent 1 minute running, and ended with 1 minute walking again. The signal acquired is represented in figure 2.

2.2.3 Running and jumping (ACC)

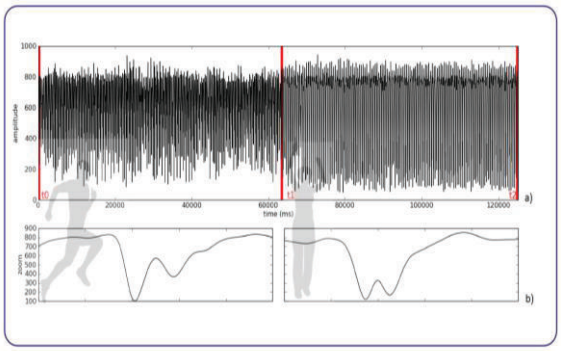

Figure 3 a): Acceleration signal of running (t0 to t1) and jumping (t1 to t2); b): corresponding zoomed waves.

With an accelerometer located at the right hip and oriented so the y axis of the accelerometer was pointing upward, the subjects performed a task of running non-stop, on a large circle drawn on the floor, and jumping also continuously but at the same place.

The subjects spent 1 minute running, followed by 1 minute jumping. The signal acquired is represented in figure 3.

2.2.4 Jumping with and without impulsion (ACC)

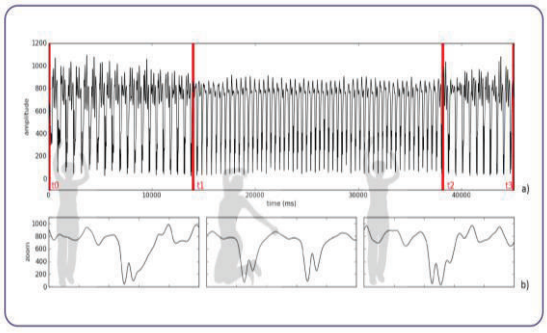

Figure 4 a): Acceleration signal of normal jumps (t0 to t1 and t2 to t3) and jumps with boost (t1 to t2); b): corresponding zoomed waves.

In this task, the following procedure was executed: 14 seconds of "normal" jumping (small jumps without a big impulsion), 24 seconds of jumping with some boost and 7 seconds of normal jumping again.

The subjects used an accelerometer located at the right hip and oriented so the y axis of the accelerometer was pointing upward. The signal acquired is represented in figure 4.

2.2.5 Skiing (ACC)

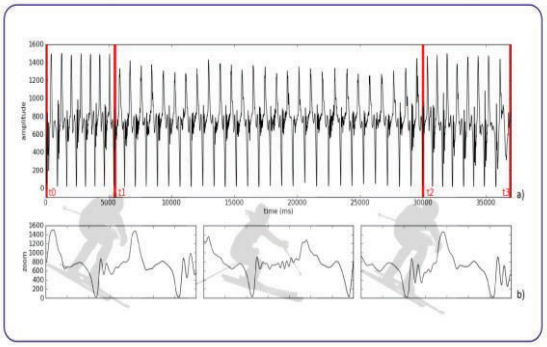

Figure 5 a): Acceleration signal of skiing with V2 technique (t0 to t1 and t2 to t3) and skiing with V1 technique (t1 to t2); b): corresponding zoomed waves.

This acquisition was made during a cross-country skiing study, in which the subject had an accelerometer attached to the ski pole, bellow the handgrip. Figure 5 illustrates the acceleration signal of that accelerometer.

In the 37 seconds of the signal the subject performed two different techniques, called V1 and V2. V1 skate is an asymmetrical uphill technique involving one poling action over every second leg stroke. V2 skate is used for moderate uphill slopes and on level terrain, involving one poling action for each leg stroke. (Erik Andersson, 2010)

The first 7 cycles of the signal (about 5 seconds) were produced through a V2 technique, the next 27 cycles (about 25 seconds) used a V1 technique and the final 8 cycles the technique was V2 again.

2.2.6 Elevation and squat of the legs (EMG)

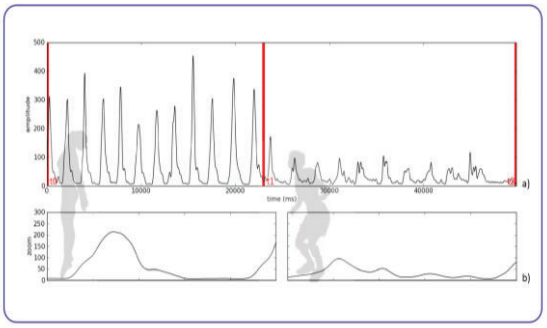

Figure 6 a): EMG signal of the gastrocnemius muscle's contraction through the elevation (t0 to t1) and squat (t1 to t2) of the inferior members; b): corresponding zoomed waves.

In this task, the subject was standing straight with both feet completely on the ground and was asked to performed 12 elevations of the legs - getting on the tiptoes and back with both feet completely on the ground - followed by 11 squats - bending the knees and back standing straight - (Figure 6). The EMG data were collected using bipolar electrodes at the gastrocnemius muscles of the right leg.

2.2.7 Normal (at rest) and high beat (after exercise) signal (BVP)

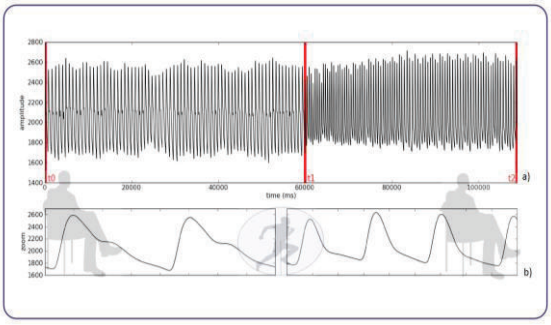

Figure 7 a): BVP signal with the subject at rest (t0 to t1) and after exercise (t1 to t2); b): corresponding zoomed waves.

The subjects were instrumented with a BVP sensor on the fourth finger of the left hand and were sitting with the left forearm resting on a platform.

One acquisition was made with the subjects at rest and then the subjects were asked to perform intensive exercise which was not collected to avoid undesirable movement artifacts. Immediately after the exercise, another BVP signal was acquired with the subject sitting again but tired. For the purpose of this study both signals (at rest and after exercise) were used in the same file, cutting a part of each signal and concatenating them offline. The resulting signal is represented in figure 7.

All the signals referenced above are available at OpenSignals (Opensignals.net website, 2010).

3 SIGNAL PROCESSING

The collected data was processed offline using Python with the numpy (T. Oliphant, 2006) and scipy (T. Oliphant, 2007) packages.

Signal processing algorithms were developed for an automatic detection of a *mean wave* representative of the signal's behavior and the k-means algorithm was used to cluster the signals. The main idea of this algorithm is to define a loop with k centroids far away from each other, take each point belonging to a given data set and associate it to the nearest centroid. Repeating the loop, the centroids position will change because they are re-calculated as barycenters of the clusters result, and after several iterations the position will stabilize and we achieved the final clusters (Dênis Martins, 2008).

Figure 8 describes the method used to process the signals. All biosignals were submitted to a signal-specific pre-processing phase and then to a generic signal-independent phase (composed with a *mean wave* and clustering procedure) which was applied to all the signals of this study.

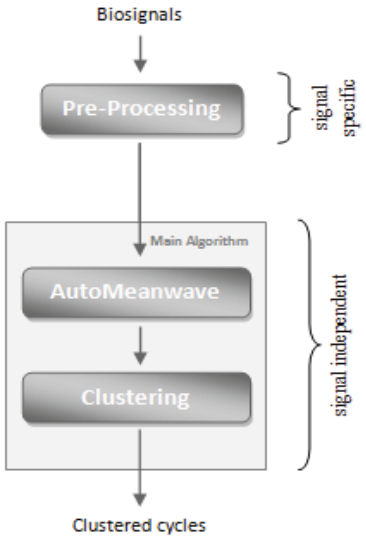

Figure 8: Signal's processing procedure schematics.

For the pre-processing phase, the acceleration signals were low-pass filtered using a smoothing filter with a moving average window of 50 points. The BVP signal was also low-passed filtered with the same moving average window as the acceleration signals. Random noise with 1/5 of original amplitude was added to the synthetic signal. The EMG signal was centered at y axis zero, by subtracting its mean value, and then rectified. Then we applied the smoothing filter with a moving average window of 300 points, to get the EMG envelope signal.

In the next sections the generic processing procedure will be described.

3.1 *Mean wave*

Algorithm 1	`autoMeanwave`
Input:	Signal, sampling frequency, trigger mode.
Output:	Fundamental frequency, window size, events.

The `autoMeanwave` algorithm is the base function to identify the individual waves. After running this algorithm we will have the *mean wave* computed with the individual wave's information.

This algorithm receives, by input, a signal, its sampling frequency and a trigger mode (this one can be omitted and the algorithm will use the maximum point as default).

As we're working with cyclic signals, the first step of the automatic *mean wave* algorithm is the detection of the signal's fundamental frequency. For that we use the `fundamentalFrequency` algorithm. With the result we compute the window size value and randomly selected a part of the signal with the same number of samples as the window size. With that signal's part and the original signal itself we run `sumvolve` algorithm to get the signal events (series of points that we consider the center of each cycle).

After this we have all the information necessary to compute the *mean wave*, which we do in the `computeMeanwave` algorithm.

Next we will describe minutely the sub-algorithms referenced above.

Algorithm 2	`fundamentalFrequency`
Input:	Signal, sampling frequency.
Output:	Fundamental frequency.

There are many ways of computing the fundamental frequency (f_0) and there isn't any ultimate method to estimate it as a procedure that returns good results for one type of signal can perform poorly for others (David Gerhard, 2003).

For the purpose of this study, the estimation of the f_0 was based on the extraction of the first signal's harmonic. So, the first step of this function was to smooth the result of the signal's fast fourier transform with a moving average window of 5% of the signal's length. We assumed the frequency value of the first big peak located at the smoothed FFT signal as the f_0 of the original signal.

With the fundamental frequency value we could compute the sampling size of a signal's cycle. We call that value "window size", with a 20% margin:

$$winsize = (f_S / f_0) * 1.2 \quad (1)$$

Being f_S the sampling frequency and f_0 the fundamental frequency. We open the window 20% to use some more samples than a cycle.

Although there are more robust methods to determine the fundamental frequency of a signal, this approach is adequate for our work as the purpose was to have a close idea of the size of a cycle. We actually use more than one exact cycle as we use a margin of 20%, opening the window calculated with the fundamental frequency. Notice that further on we use a correlation function to detect meaningful events on a cycle, so the fundamental frequency is just used as a preliminary estimation to support other algorithms.

Algorithm 3	`sumvolve`
Input:	Two signals
Output:	Distance values.

This algorithm works as a correlation function. Sliding the smaller window part of the signal (given by argument) through the original signal, one sample at a time, this algorithm compares the distance of the two overlapped waves. The expression used to compute the distance value for each sample is exposed in equation (2).

$$distance_i = \frac{\sum_{j=1}^{N} \left| sig_{[i:i+N]j} - window_j \right|}{N} \quad (2)$$

With *i* ranging from 1 to number of signal's samples minus the window size. The result of this algorithm is a signal composed with distance values. That distance signal shows the difference between each overlapped cycle and the window selected at the first place.

After that, all the minimum peaks of the resulted correlation signal were found. Those peaks were assumed as the cycle's events.

Algorithm 4	`computeMeanwave`
Input:	Signal, events, window size.
Output:	*Mean wave* and *standard deviation error wave.*

With the events and the window size, the signals can be separated into periods that are assumed as the signal's cycles:

$$\text{cycle} = \text{signal}[\text{event} - \text{winsize}/2 : \text{event} + \text{winsize}/2] \quad (3)$$

This way, based on all cycles, we could compute the mean value to each cycle sample, and compose a *mean wave*. The *standard deviation error wave* is computed with the same principle, calculating the standard deviation error instead of the mean value. For a better visualization of the results, we computed an error area with the *standard deviation error wave* obtained. For that, we added and subtracted one *standard deviation error wave* to the *mean wave*, getting a superior and inferior wave, to graphically present the error area (66% of the error). This is shown in the results section.

After the procedure described above, a final adjustment was made: the rearrangement of the cycle's events positions. A trigger position is computed to rearrange the cycle's events. The trigger position is a notable point of the previously traced *mean wave*. The chosen trigger mode for this study was the *mean wave*'s minimum point – other possibilities were designed, we could use the maximum (of the signal or the derivative signal), or the zero crossings, for example.

With this trigger point we recalculate the peak events, or cutting points, used in the `computeMeanwave` algorithm:

$$\text{events} = \text{events} + \text{trigger} - \text{winsize}/2 \quad (4)$$

With the events variable recalculated, we used the `computeMeanWave` function again, so the cycles had the center on a notable point visually more recognizable in the *mean wave*.

3.2 Clustering

Algorithm 5	`distanceMatrix`
Input:	Signal, Cutting events, window size.
Output:	Matrix with wave-to-wave distances.

For the clustering procedure we developed a function that receives the signal to cluster, the window size and cutting events produced with the `autoMeanwave` algorithm.

We go through all the cutting events and for each we select a part of the signal with center at that event and a number of samples to both sizes equal to the window size. Then we compare that cut with each of the others (with the center in the others cutting events and the same window size), using the mean square error distance formula:

$$distance = \sqrt{\sum_{i=1}^{N}(cycle1_i - cycle2_i)^2} \quad (5)$$

With $\mathbf{cycle_1}$ and $\mathbf{cycle_2}$ being the parts of the signal selected before.

With all of the distance values for each wave, we built a matrix of distances.

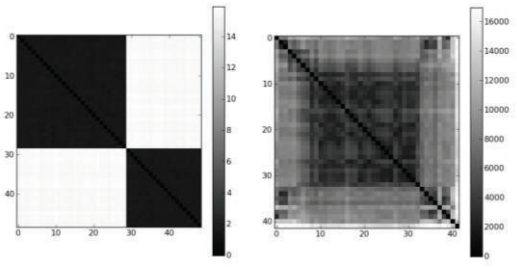

Figure 9: Matrix of distances produced for the synthetic waves (a)) and the skiing task (b)).

Figure 9 presents two matrices of distances, obtained with the *imshow* command. Figure 9 a) shows the matrix of the synthetic waves distances and figure 9 b) the matrix for the skiing task distances. As we can see, the synthetic matrix is almost ideal, as all the waves are equal – the distance values are only minimums or maximums. In the skiing matrix the matrix assumes a greater variation of distances, as the cycles are not exactly the same. However, it's visible the similarity between the cycles of the same technique (7 cycles V2, 27 cycles V1 and 8 cycles V2).

To cluster the signal we used the k-means algorithm. This algorithm received the matrices created with the `distanceMatrix` algorithm and the number of clusters expected in the data, returning the clusters and distances to the clusters.

Table 1: Clustering results

Task	*Number of Cycles*	*Cycles correctly clustered*	*Errors*	*Misses*
Synthetic	50	49	0	1
Walk and run	343	342	1	0
Run and jump	296	295	1	0
Jumps	85	84	1	0
Skiing	42	41	0	1
Elevation and squat	23	23	0	0
BVP rest and after exercise	165	159	4	2
All	1004	993	7	4

4 RESULTS AND DISCUSSION

Figure 10 shows the graphics of the resulting *mean wave*s (line) and deviation error area (filling) after running the algorithms referenced above. At the left the graphics represent the initial *mean wave*s traced, before the clustering procedure. At the right we have the *mean waves* representative of the signal parts that were divided according to the resultant clustering codebook.

It's visible that the *mean wave*s at the left gather information about the signal's behavior, even if there are some changes in its shape or frequency. After the clustering procedure there are some predictable variations in the resultant *mean wave*s. We notice an overall reduction of the deviation error after the clustering procedure and also a reshaping of the *mean wave*.

After running the clustering procedure we gather the results for each task performed. These results are exposed in table 1.

It is important to note that some cycles weren't classified, and that occurred because sometimes the borders of the signal didn't have a full cycle - the `distanceMatrix` algorithm (algorithm 5) cannot be used to compare a short cycle with the regular ones. Therefore, those cycles have been rejected for lack of pattern quality, and won't be taken into account.

In the "walk and run" activity there were some extra classification points. The cycles were correctly clustered (with only 1 error encountered), but in the "walking" mode there were some extra points between those cycles that were also counted. That occurs due to a relatively large variation in the fundamental frequency from the walking to the running activity - despite one activity has all cycles well defined by the window size variable, the other has less than one cycle per window size. This condition shows a limitation of our algorithm: doesn't allow big changes in the frequency domain for the different modes presented on the signal.

Only 7 errors resulted from this algorithm, and 2 of those errors were in transition periods – where the cycle wave is still reshaping to form the other activity and the distance value to the *mean wave* or to the clusters mean values is bigger than anywhere else on the signal. This occurred in the jumps and in the walk and run activities.

Given the results we can affirm that our clustering algorithm based on the *mean wave* information only returned 7 errors out of 999 cycles with pattern quality, and therefore we achieved 99.3% of efficiency.

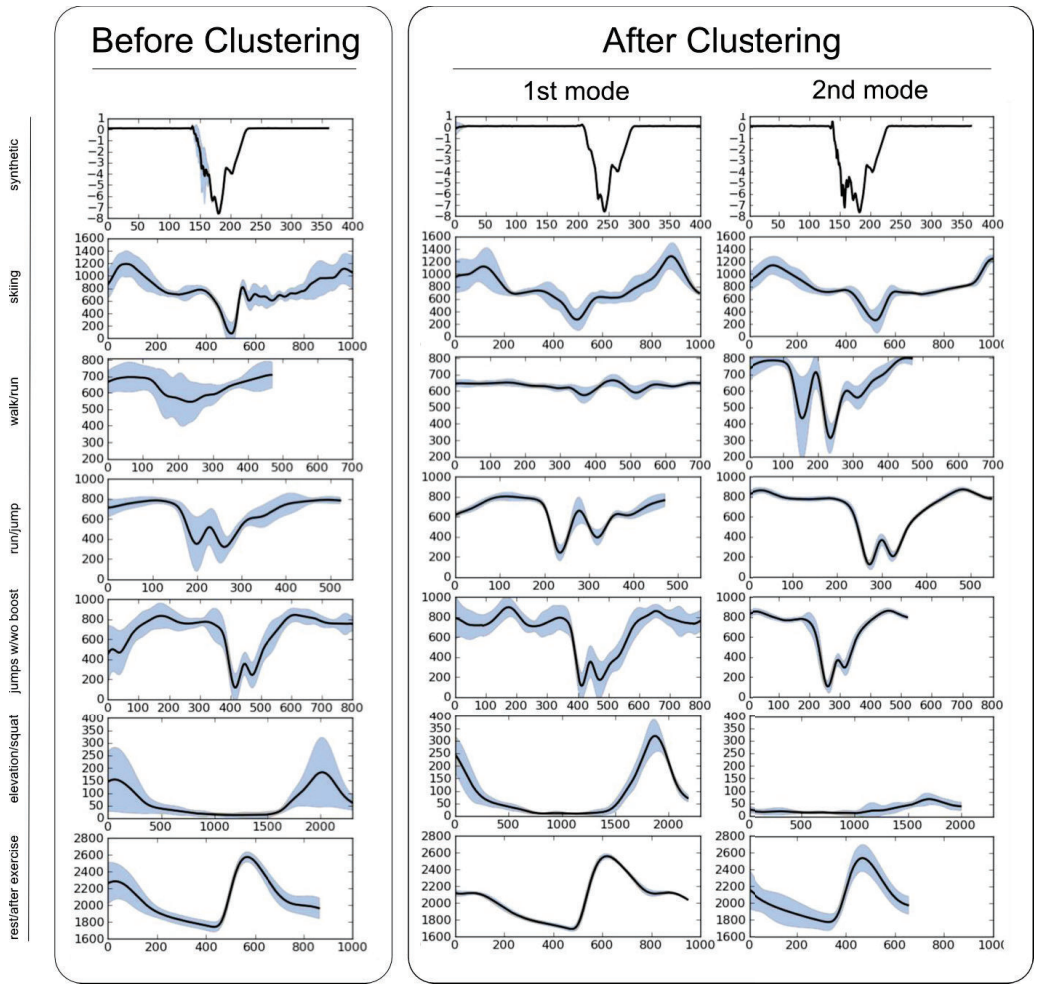

Figure 10: *Mean waves* of all the tasks before and after the clustering procedure.

5 CONCLUSIONS

The proposed algorithm represents an advance in the biosignals processing field, as it has an effective detection of signal variations, tracing different patterns for distinct clusters, whether it's an activity, synthetic or physiological signal.

FUTURE WORK

In future works we intend to repeat this procedure with a wider range of subjects performing the same task. We also intend to perform a noise immunity test and run the algorithm using a signal with more than two modes.

We intend to introduce an automatically perception of the cycles which are too distance from the cluster and assign those cycles to a new "rejection class". This will reduce the number of errors due to a strange cycle, in particular the mode's transition cycles.

The local detection of the fundamental frequency is also a future goal, as we intend to realize when there's a major variation of fundamental frequency and make our algorithms adapt its behavior according to that variation.

Finally, we have the intention of creating a multimodal algorithm, which can receive more than one signal, and process those at the same time and with the same treatment. This could be useful if we want to use the 3 axis of an accelerometer, or

conciliate the information of a BVP with an electrocardiography (ECG) signal.

ACKNOWLEDGEMENTS

The authors would like to thank PLUX – Wireless Biosignals for providing the acquisition system and sensors necessary to this investigation. We also like to thank NIH, the Norwegian School of Sports and Science, Håvard Myklebust and Jostein Hallén for acquiring and allowing us to work with the Skiing signal used in this study. We acknowledge Rui Martins and José Medeiros for their help and advices on the BVP acquisition procedure.

REFERENCES

Andersson, E., Supej, M., Sandbakk, Ø., Sperlich, B., Stöggl, T., Holmberg, H. (2010) Analysis of sprint cross-country skiing using a differential global navigation satellite system. Eur J Appl Physiol. DOI 10.1007/s00421-010-1535-2

Baek, J., Lee G., Park, W., Yun, B. (2004). Accelerometer Signal Processing for User Activity Detection, *Knowledge-Based Intelligent Information & Engineering Systems,* Vol. 3215/2004, 610-617. DOI 10.1007/978-3-540-30134-9_82

Ben-Arie, J., Wang, Z., Pandit, P., Rajaram, S. (2002). Human Activity Recognition Using Multidimensional Indexing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 8, 1091-1104.

Fridlund, A., Schwartz, G., Fowler, S. (2007) Pattern Recognition of Self-Reported Emotional State from Multiple-Site Facial EMG Activity During Affective Imagery. *Society for Psychophysiological Research.*, vol. 21, no. 6. DOI 10.1111/j.1469-8986.1984.tb00249.x

Gerhard, D. (2003) Pitch extraction and fundamental frequency: History and current techniques). *Technical Report TR-CS 2003-06.*

Liao, T. Warren. (2005) Clustering of time series data – a survey. Pattern Recognition 38 (2005) 1857 – 1874.

Martins, D., Mattos, M., Simões, P., Cechinel, C., Bettiol, J.; Barbosa, A. (2008) Aplicação do Algoritmo K-Means em Dados de Prevalência da Asma e Rinite em Escolares. In: *XI Congresso Brasileiro de Informática em Saúde (CBIS'2008),* 2008.

PLUX – Wireless Biosignals, bioPLUX Research Manual, PLUX's internal report, 2010.

Quiroga, Rodrigo Q. (2007) Spike sorting. Scholarpedia, 2(12):3583

Smith, J., Fishkin, K., Jiang, B., Mamishev, A., Philipose, M., Rea, A., Roy, S., Sundara-Rajan, K. (2005). RFID-Based Techniques for Human-Activity Detection. *Communications of the ACM*, vol. 48, no. 9, 39-44.

T. Oliphant. Guide to Numpy. Tregol Publishing, 2006.

T. Oliphant. SciPy Tutorial. SciPy, http://www.scipy.org/SciPy Tutorial, 2007.

Wu, X., Kumar, V., Quinlan J., Ghosh J., Yang, Q., Motoda, H., McLachlan, G., Ng, A., Liu, B., Yu, P., Zhou, Z., Steinbach, M., Hand, D., Steinberg, D. (2007). Top 10 algorithms in data mining. *Knowledge Information Systems*, 14:1–37. DOI 10.1007/s10115-007-0114-2

www.opensignals.net, last accessed on 15/07/2010

A.2 *Biosignals 2011*

Morphological Analysis of Acceleration Signals in Cross-Country Skiing

MORPHOLOGICAL ANALYSIS OF ACCELERATION SIGNALS IN CROSS-COUNTRY SKIING

Information extraction and technique transitions detection

Håvard Myklebust
Research centre for training and performance,
Norwegian school of sports sciences, Oslo, Norway
havard.myklebust@nih.no

Neuza Nunes
Physics Department, FCT-UNL, Lisbon, Portugal
neuzanunes@gmail.com

Jostein Hallén
Research centre for training and performance,
Norwegian school of sports sciences, Oslo, Norway
jostein.hallen@nih.no

Hugo Gamboa
Physics Department, FCT-UNL, Lisbon, Portugal
PLUX – Wireless Biosignals, Lisbon, Portugal
h.gamboa@fct.unl.pt, hgamboa@plux.info

Keywords: Cross-country skiing, accelerometers, expert-based classification, biosignals, signal-processing.

Abstract: **Aims**: Experience morphology of acceleration signals, extract useful information and classify time periods into defined techniques during cross-country skiing. **Method**: Three Norwegian cross-country skiers ski skated one lap in the 2011 world championship sprint track as fast as possible with 5 accelerometers attached to their body and equipment. Algorithms for detecting ski/pole hits and leaves and computing specific ski parameters like cycle times (CT), poling/pushing times (PT), recovery times (RT), symmetry between left and right side and technique transition times were developed based on thresholds and validated against video. **Results:** In stable and repeated techniques, pole hits/leaves and ski leaves were detected 99% correctly, while ski hits were more difficult to detect (77%). From these hit and leave values CT, PT, RT, symmetry and technique transitions were successfully calculated. **Conclusion:** Accelerometers can definitely contribute to biomechanical research in cross-country skiing and studies combining force, position and accelerometer data will probably be seen more frequently in the future.

1 INTRODUCTION

The increased numbers and decreased sizes of electronic devices is a major cause to the development of biomechanical research in real sports situations the last 15 years. In cross-country (XC) skiing research, different research groups have mounted small strain gauges into the poles and used commercial insoles for measuring forces from arms and legs of the skiers in addition to video recordings for quite some years (Millet et al. 1998, Holmberg et al. 2005, Stöggl et al. 2010). In addition to forces they often present parameters like cycle time (CT), poling/pushing time (PT), recovery time (RT) and

figures showing timing of arms and legs (Lindinger et al. 2009). The strain gauges used, still have some limitations though. The weight and size of the equipment and the fact that skiers can not use their own poles makes data collection from competitions more difficult.

Skiers change between different types of techniques many times during a XC-skiing competition. It can be speculated if one technique is better than another in special types of terrain. We know there have been some coaches and researchers systematically looking at video and split times in different terrains, trying to understand what techniques are most efficient. Recently Anderson et al. (2010) presented a work in XC-skiing where a GPS were synchronised to video to get position and speed when the skiers changed technique.

In alpine skiing, Supej (2010) validated a system combining a suit with inertial sensors (accelerometers) and GPS for detecting body trajectory and segment movements. To our knowledge, accelerometers have not been used in XC-skiing.

The aims of this study were therefore to use accelerometers to extract cycle time (CT), poling/pushing time (PT), recovery time (RT) and symmetry between right and left side during XC-skiing using video recordings for validation. We also intended to develop an expert-based classification system which classifies the techniques used and detects the moments of technique transitions. This can help coaches and researchers in analysing the effect of different techniques in different tracks more effectively.

The following sections will describe our study and expose the results achieved. In section 2 we describe the acquisition scenario, the participants and apparatus used. In section 3 we expose the data analysis and processing, and how we acquire the necessary information from the accelerometers. Section 4 describes the procedure used to classify the cycles into techniques. Section 5, 6 and 7 presents the results, discussion and conclusion of our work, respectively.

2 MATERIALS AND METHODS

2.1 Overall study procedure

In this study, three XC-skiers finished the World Championship 2011 sprint event track (1480m) as fast as possible. They had accelerometers attached to their body and equipment, while two hand held cameras videotaped most of the track for validation.

The acquired data were analysed for the initiation (hit) and finalization (leave) events of skis and poles ground contact. The exact times when these events occurred were computed and validated against the video.

With these time points we were able to calculate CT, PT, RT and symmetry between right and left ski/pole. We also developed an expert-based system which classified the cycles of the accelerometer signals into defined skiing techniques, by fitting in the thresholds defined after signal analysis.

Because the World Cup was held this day, the snow conditions were optimal and we could get top level athletes to participate, but we could not standardize the start and end point of the track 100%.

2.2 Subjects

Three Norwegian male XC skiers, two 17 year old juniors and a 21 year old senior volunteered to participate in this study. The juniors (FP2 and FP3) are among the best in their age in Norway and the senior (FP1) were participating in the World Cup the day of testing. He volunteered to take a run with the accelerometers about one hour after he failed to qualify for the finals.

2.3 Techniques

The track used is designed in accordance to international regulations and made the skiers change between all normal skating techniques. We choose to name the techniques V1, V2, V3 and V0.

V1 is generally considered as an uphill technique and uses an asymmetrical and asynchronous pole push on one leg (strong side) but not on the other leg (weak side). This technique is also called “paddling”, “offset”, “gear 2” and other names in the literature. If the strong side is simultaneously with the right ski push we call the technique V1r and if the strong side is simultaneously with the left ski push we call the technique V1l.

V2 is usually viewed as a high speed technique used on flat terrain or moderate uphill. With this technique propulsive forces are symmetrically and synchronously applied during the ground contact of the poles for each skating push (both sides). Other names are “double dance”, “one skate” and gear 3.

V3 is used at even higher speeds on flat terrain. The technique is similar to V2 but the poles are only

used on one side. Other names are “single dance” and gear 4.

V0 is here used for all other techniques including downhill, freeskate (just legs working) and turning techniques.

2.4 Apparatus and experimental design

To collect the acceleration data necessary for this study, five triaxial accelerometers, xyzPLUX (bioPLUX Research Manual, 2010), were used.

One accelerometer (ACG) was placed at the subject’s lower back on the lumbar region, near the centre of gravity. The default x axis of the accelerometer was orientated with positive values from left to the right, the default y axis were on the vertical direction, being positive from inferior-superior direction and the default z axis had positive values from posterior to anterior orientation. One accelerometer was attached to each pole just below the handgrip, and one accelerometer was attached at the heel of each ski-boot. The last four accelerometers were used as uniaxial accelerometers, as only one axis of the accelerometers (the one pointing upward in a neutral position) was connected to the acquiring system device.

To acquire and convert acceleration signals to digital data, a wireless acquisition system, bioPLUX research, was used. The system has a 12bit ADC with a sampling frequency of 1000Hz and the information is transmitted by Bluetooth at real-time. In this particular test a HTC mobile phone with Windows Mobile 6.1 received and stored the collected data for post processing, using an application, loggerPLUX, created for that purpose. (bioPLUX Research Manual, 2010)

3 DATA ANALYSIS

The data collected with the accelerometers was processed offline using Python with the numpy (T. Oliphant, 2006) and scipy (T. Oliphant, 2007) packages. Algorithms were developed to automatically perceive the initiation (hit) and finalization (leave) time of each ski and pole ground contact. For checking these time points against the video, Dartfish Connect 4.5.2.0 (Dartfish.com website, 2010) was used. Also, with this information, it was possible to compute CT, PT, RT, symmetry between right and left side, technique used and time points for technique transitions.

Figure 1 summarizes the data analysis procedure that is minutely described foremost in this section.

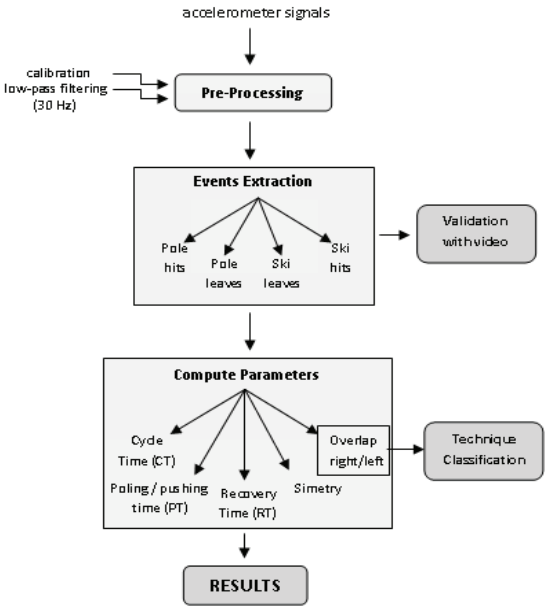

Figure 1: Schematics of the procedure.

3.1 Preliminary Processing

The primary procedure was to apply a low-pass filter with a cutting frequency of 30Hz to all signals.

We then converted the accelerometer data to G-units using calibration constants from each accelerometer. To get the calibration constants we acquired the rotation signal of the sensors through the 3 axes, so that acceleration on each axis ranged from -1g to +1g. The calibration constants are the maximum and minimum values on each axis. We get the mean value of these constants and with that information we can finally convert our acceleration data to G-units, applying the following formula:

$$s\_cal = (s - mean\_cal) / (max\_cal - mean\_cal) \quad (1)$$

with **s** being our acceleration signal, **max_cal** the maximum calibration constant, **mean_cal** the mean of the two calibration constants and **s_cal** our signal after the conversion.

For ACG we calculated the total acceleration from the following formula:

$$a\_total = sqrt((a\_x)^2+(a\_y)^2+(a\_z)^2) \quad (2)$$

where a_x, a_y and a_z is the acceleration in the three directions.

3.2 Poles

The first data analysed were the signals from the right and left poles accelerometers. In order to get the moments when the pole hits and leaves the ground, we needed to exhaustively analyse the signal's behaviour and also its jerk and span signals (1^{st} and 2^{nd} derivative), so we could get the optimal thresholds for all the subjects.

In the next sections we will describe the procedure to differentiate the pole hits from the pole leaves.

3.2.1 Pole hits

By video and signal analysis we concluded that the pole hits happens near an inflexion point just after a minimum peak of the signal.

We took all the maximums of the jerk signal that were bigger than 0.035G/s and all the maximums of the span signal that were bigger than $0.0025G/s^2$ (optimal values we estimated after some analysis) and the pole hits were considered to be the samples giving the maximum jerk values that were close to (less than 50 samples apart) the maximum span signal. To eliminate some undesirable points, the events should correspond to a low signal value (less than -0.38G).

After this procedure there were still some extra poling hits mistakenly calculated, so we eliminate all the events that were less than 300 ms apart from each other. We also knew that left and right pole hits should be almost at the same time and eliminated the ones with a distance value bigger than 75ms.

3.2.2 Pole leaves

Analysing the video data synchronized with our signal, we concluded that the pole leaves happens near an inflexion point just before a maximum peak of the signal.

We therefore defined the pole leaves as the points were the maximums of the jerk signal were bigger than 0.04G/s, if that corresponded to a high signal value (more than 0.29G).

To eliminate some extra poling leaves mistakenly calculated, we eliminate all the events that were less than 300 ms apart from each other. We also knew that the left and right pole hits should be almost at the same time so we erased the ones with a distance value bigger than 100ms.

3.3 Skis

As the skis acceleration signals were very distinct from the poles acceleration signals, the processing used with the skis was somewhat different to the one used with the poles. For this part of the processing it was also necessary to analyse the signals with detail to get the optimal thresholds.

The procedure to get the ski hits and leaves will be described below.

3.2.1 Ski leaves

We began this part of the processing finding the maximum points of the ski signal that had a value bigger than 2.0G. However, with this approach some ski hit points were mistakenly confused as leave points. We then low pass filtered the acceleration signal with a smoothing average window of 500 samples and found the maximum peaks again but with a threshold of 1.323G. With this big smoothing window not all the peaks computed before met the required threshold value.

After that we compared the two peak results and we eliminated all the events that were more than 100ms apart, in other words, we erased some of the peaks encountered with the 2.0G threshold because they don't reach the 1.323G with a smoothing factor applied.

To eliminate some extra ski leave points, we eliminate all the events that were less than 200ms apart from each other.

3.3.2 Ski hits

For the ski hit events we only used the span signal of the left and right skis. We detect the minimum peaks that had a value lower than $-0.0045G/s^2$, and eliminate all the peaks that were less than 200 ms apart. To erase the downhill parts (undesirable because the skis don't leave the ground) we compared the skis leaves computed before with the skis hits and erased all the events that were more than 1300ms apart. We still had too many hit values compared with the leave ones, so we erased all the hits that were too close of the next leave (less than 250ms).

3.4 Skiing parameters

3.4.1 Cycle time, poling/pushing time and recovery time

From the hits/leaves for poles/skis we could calculate CT, PT and RT using these definitions:

(3) The cycle time (CT) is the time spent in each cycle. We consider that the beginning and ending of the cycle is a hit point. So, to compute the cycle times we get the distance values between all the hit events.

$$CT_i = hit_{i+1} - hit_i \quad (3)$$

Remark that calculating CT in V2 technique using pole hits require to use time between every other pole hit.

(4) Poling/pushing time (PT) is defined as the time spent with the ski or pole on the ground, the time between a hit and a leave. To compute these values, we subtract the hits events to the corresponding leaves points.

$$PT_i = leaves_i - hits_i \quad (4)$$

(5) The recovery time (RT) is the time which the subject spends takes to begin another cycle, after getting the ski or pole off the ground. That way this value can be defined as the cycle time minus the pulling time.

$$RT_i = CT_i - PT_i \quad (5)$$

3.4.2 Symmetry between right and left side

Another interesting variable is the symmetry between right and left side and if pole hits/leaves are synchronic or not. This was checked by subtracting hit, leave, CT, PT and RT calculated from right pole from the values calculated from the left pole. For example, for the poling/pushing times we did:

$$Sync\_PTpoles_i = PTleft\_pole_i - PTright\_pole_i \quad (6)$$

4 DATA CLASSIFICATION

The information gathered about the hitting and leaving timepoints from the ski and pole accelerometers were used also to classify the data into techniques.

For each pole hit we calculated two variables, one giving the time distance to the closest right ski leave ("overlap_right") and one giving the time distance to the closest left ski leave ("overlap_left"). Since this distances vary between techniques we could detect which technique each pole hit represented and from this also calculate the time points of the technique transitions.

Again, we had to analyse the overlap results for all the subjects in detail, to get the correct thresholds that separates and classifies our cycles correctly. The optimal thresholds were:

V1 right technique

$$250 < overlap\_right < 500$$
and
$$-50 < overlap\_left < 200$$

V1 left technique

$$-150 < overlap\_right < 130$$
and
$$290 < overlap\_left < 575$$

For V1 and V3 (see later) techniques it's also necessary that the previous or next cycle presents the same values for overlap_right and overlap_left.

V2 technique

As the V2 technique has a poling action for each ski push, there are two classifications possible with different thresholds.

Either:

$$300 < overlap\_right < 600$$
and
$$-570 < overlap\_left < -250.$$

And the previous or next cycle must be:

$$-530 < overlap\_right < -250$$
and
$$300 < overlap\_left < 655.$$

Or (switched around):

$$-530 < overlap\_right < -250$$
and
$$300 < overlap\_left < 655$$

and the previous or next cycle must be:
300 < overlap_right < 600
and
-570 < overlap_left < -250.

V3 right technique
-530 < overlap_right < -250
and
300 < overlap_left < 655

V3 left technique
300 < overlap_right < 600
and
-570 < overlap_left < -250

As for V1 technique, it is necessary that the previous or next cycle presents the same values for overlap_right and overlap_left.

Other techniques
All the other values that don't fit on any of the situations referenced above were classified as "other techniques" (V0).

5 RESULTS

5.1 Quality of our subjects

The junior skiers skied at a speed corresponding to 88% and 91% of the senior skier (FP1), respectively. When the senior skier skied for us he held a speed corresponding to 98% of the pace he used during the world cup event, which again corresponds to 97% of speed required to qualify for the finals in the world cup (less than 3 minutes).

5.2 Validity of hits and leaves

Our algorithm detected 99% of the pole hits and leaves correctly. For the ski leaves, 95-99% were detected correctly (Table 1), depending on if you look at all leaves in the track or only at parts of the track with stable technique (ST) over some time (only V1 or V2 in this samples).

For ski hits our code detected 77% correctly for ST. The problems of detecting hits were clearly greater in the V2 than in the V1 technique (Table 2).

5.3 Skiing parameters

5.3.1 Technique changes and % of time

Out of totally 67 technique transitions, our code made 8 mistakes, in other words 88% correct detection. The mistakes were 6 false transitions, 1 transition with wrong technique and 1 transition missing. Figure 2 shows the % of time in each technique based on the calculated technique time changes.

Table 1: Number of hits and leaves from poles and skis detected from video and % of correct detection from our algorithm. ST meaning stable techniques held over several cycles where in this case is only V1 and V2 techniques.

FP	Pole hits		Pole leaves		Ski leaves		
	N video	Correct (%)	N video	Correct (%)	N video	Correct (%)	Correct ST (%)
FPH	224	100.0%	224	100.0%	154	97.4%	99.4%
FPL	300	100.0%	296	100.0%	250	93.2%	98.8%
FPS	316	99.7%	282	99.6%	242	95.0%	99.2%
Total	840	99.9%	802	99.9%	646	95.2%	99.1%

Table 2: Number of ski hits analyzed from video (n) and correct detection from our code (%) subdivided into "all" (all techniques), "ST" ("stable techniques" held over several cycles, where in this case only V1 and V2 techniques), V1 and V2. Two hits per cycle were sometimes found in V2. The table shows how many of this 2.hit we found and how many % of correct detection our code gets if we assume that the 2.hit is wrong or correct.

FP	Ski hits						
	Correct				Correct V2		
	N video	All (%)	ST (%)	V1 (%)	N video	2 hit = wrong	2 hit = correct
FPH	172	67.0%	77%	97.0%	27	48.0%	85.0%
FPL	264	74.0%	86%	96.0%	14	71.0%	88.0%
FPS	251	59.0%	69%	95.0%	33	16.0%	63.0%
Total	687	67.0%	77%	96.0%	74	47.0%	57.0%

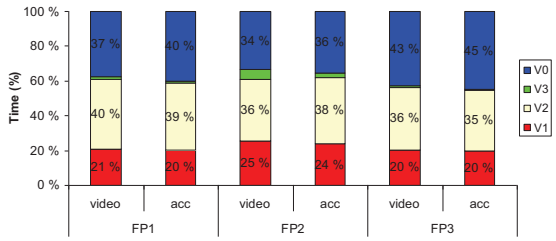

Figure 2: Relative time in each technique for each FP based on video analysis and accelerometer data (acc).

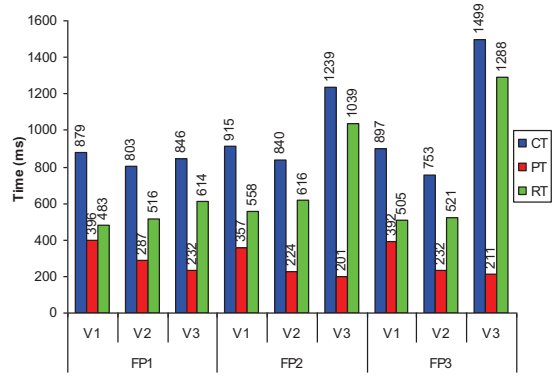

Figure 3: Mean CT, PT and RT for each technique and each FP based on right pole. Remark that CT, PT and RT for V2 will be twice as big for a complete cycle.

5.3.2 Cycle time, poling/pushing time, recovery time and timing of events

Differences between techniques were seen for CT, PT and RT (Figure 3). Figure 4 shows differences in timing of events between skiers in V1 technique and this can also be seen as differences between when poles and skis hits/leaves ground compared to centre of gravity total acceleration in the different skiers (Figure 5).

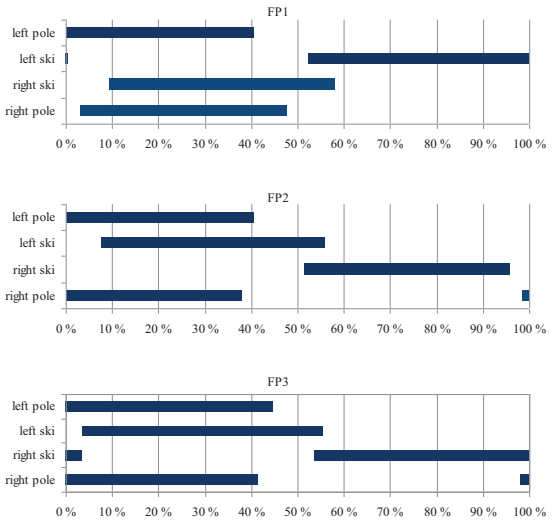

Figure 4: Cycle phase structure in V1 for the different subjects.

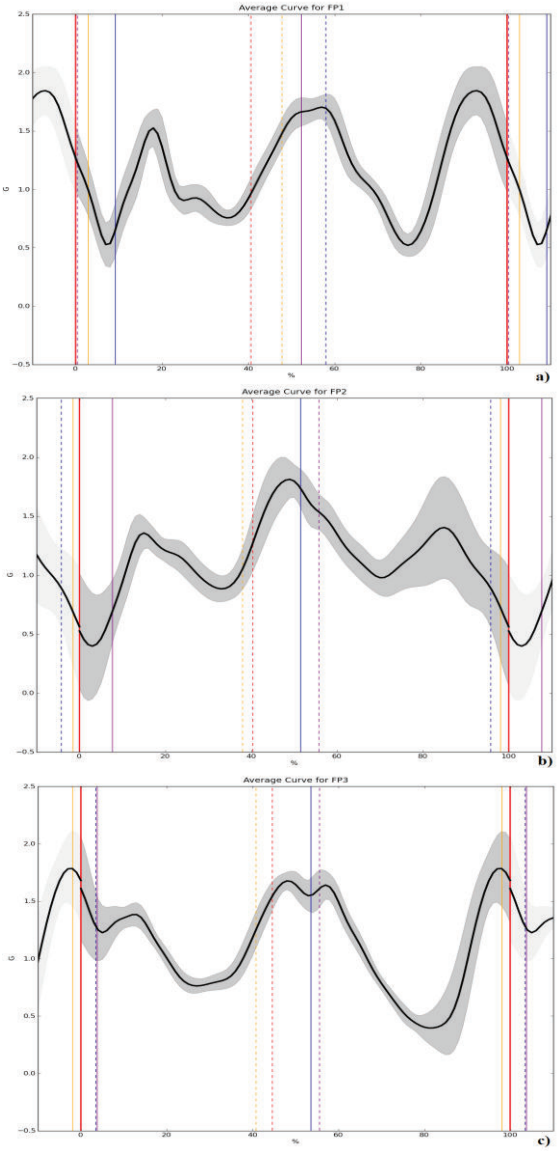

Figure 5: Average total acceleration from ACG during V1 technique. Time points for hits (solid lines) and leaves (dashed lines) of poles (orange = right, red = left) and skis (blue = right, purple = left), for FP1, FP2 and FP3 subjects (Figure 5 a), b) and c) respectively).

5.3.3 Symmetry between right and left side

FP1 had clear differences in symmetry between left and right pole in V1 compared to V2. This was not found in the other subjects, at least not in FP2 (Figure 6). Remark that FP1 used V1r (pole push simultaneous with right ski push) while FP2 and FP3 used V1l (pole push simultaneous with left ski push). FP1 did not ski the end of the track where

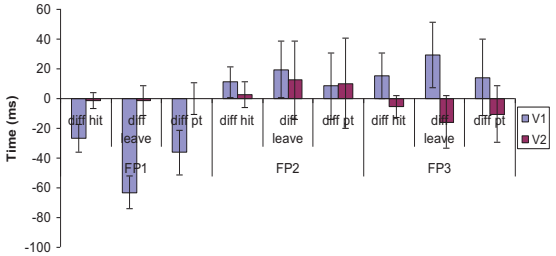

Figure 6: Time differences (Mean (SD)) in pole hit, pole leave and PT between left and right poles. Negative values for FP1 V1 mean that left pole hits the ground first, leaves the ground first and right pole has most time in the ground.

there was a typically V1 uphill and the uphill he (and the others) skied was in a slightly right curve. Even though this might influence the data a bit, we also see that FP1 has less variation (smaller standard deviation) than the others indicating a more stable technique (Figure 6).

6 DISCUSSION

Our approach gave good results in the detection of pole hits/leaves and ski leaves. In addition to calculate CT, PT and RT previously only calculated when measuring forces (Stöggl 2010, Lindinger 2009), we were able to detect technique transitions.

Ski hits were more difficult to detect, especially in V2 because two hits sometimes showed up. This second hit results from a re-direction of the ski before the push off. Some skiers clearly use this newly developed "double-push" technique described by Stöggl (2008), and others (like our subjects) change technique over time using something in between of "double-push" and traditional V2. As the signals sometimes shows the second hit and other times doesn't, and we are unsure if and when the second hit should be there and not, the worst results we get from the ski hits could be understood. This was also the reason why we did not present CT, PT and RT from the skis. We clearly have to either find a better approach or use strain gauges or pressure sensors for detecting ski hits. One approach might be to create a separate algorithm when the technique is classified as V2.

In addition to forces, strain gauges and force sensors can give the same timing parameters of hits and leaves as we have found with accelerometers, but we will point that the weight of equipment used for measuring forces are 3-5 times as high as our accelerometer equipment (1,5 kg vs. 300-500g. Stöggl 2010). We also think our equipment is easier

to put on the skiers and the skiers can use their own poles. Even though we used accelerometers with cables into the wireless acquisition system in this study, there will shortly be devices available without need of cables. This will make the preparation even easier.

Combining different technologies like Supej (2010) have done in alpine skiing will probably be the future of biomechanical research. Accelerometer data from the area around centre of gravity or different limbs of the body in addition to force and positioning data will probably be useful during XC-skiing research.

7 CONCLUSIONS

Accelerometers were shown to be useful tools in XC skiing research. Accelerometers will probably be used more frequently in the future, in combination with force and positioning systems. Working with accelerometers can give insight in biological movement patterns and can give both solutions and ideas for more advanced biomechanical questions in the future.

FUTURE WORK

The thresholds used were fitted for these subjects and situation. Shortly, we will test the procedure on more data and different situations. We will try to improve our methods by finding the thresholds automatically and we will also check what information we can get from fewer accelerometers. The problems of finding ski hits obviously need more effort and we will continuously give feedback to the producers for developing even better equipment.

ACKNOWLEDGEMENTS

Thanks to the organising committee of the Holmenkollen 2010 World Cup for allowing our testing between the arrangement, and the subjects for participating on such short notice!

REFERENCES

Andersson, E., Supej, M., Sandbakk, Ø., Sperlich, B., Stöggl, T., Holmberg, HC., 2010. Analysis of sprint cross-country skiing using a differential global navigation satellite system. *Eur J Appl Physiol.* 2010 Jun 23 (Epub ahead of print)

Holmberg, HC., Lindinger, S., Stöggl, T., Eitzlmair, E., Müller, E., 2005. Biomechanical analysis of double poling in elite cross-country skiers. *MedSci in Spors & Exercise*, 37(5), 807-18.

Lindinger, S.J., Göpfert, C., Stöggl, T., Müller, E., Holmberg, HC., 2009. Biomechanical pole and leg characteristics during uphill diagonal roller skiing. *Sports Biomechanics,* 8(4), pp 318-333.

Millet, G.Y., Hoffman, M.D., Vandau, R.B., Clifford, P.S., 1998. Poling forces during roller skiing: effects of technique and speed. *Med Sci in Sports & Exercise,*30(11) pp 1645-1653.

PLUX – Wireless Biosignals, bioPLUX Research Manual, PLUX's internal report, 2010.

Stöggl, T., Müller, E., Lindinger, S., 2008. Biomechanical comparison of the double-push technique and the conventional skate skiing technique in cross-country sprint skiing. *J Sports Sci.* 26(11), 1225-1233

Stöggl, T., Müller, E., Ainegren, M., Holmberg, HC., 2010. General strength and kinetics: fundamental to sprinting faster in cross country skiing? *Scand J Med Sci Sports*, 2010, 1-13.

Supej, M. 2010. 3D measurements of alpine skiing with an inerial sensor motion capture suit and GNSS RTK system. *J Sports Sci.* 28(7), 759-69.

T. Oliphant. Guide to Numpy. Tregol Publishing, 2006.

T. Oliphant. SciPy Tutorial. SciPy, http://www.scipy.org/SciPy Tutorial, 2007.

www.dartfish.com, 2010, last accessed on 19/07/2010

A.3 *BMS2010*

Comparison Between the Standard Average Muscle Activation with the use of Snorkel and without Snorkel in Breakstroke Technique

COMPARISON BETWEEN THE STANDARD AVERAGE MUSCLE ACTIVATION WITH THE USE OF SNORKEL AND WITHOUT SNORKEL IN BREAKSTROKE TECHNIQUE

Conceição, A[1,2].; Gamboa, H. [3]; Palma, S. [3]; Araújo, T. [3]; Nunes, N. [3]; Marinho, D. [4,2]; Costa, A. [4,2]; Silva, A. [5,2]; Louro, H. [1,2]

[1] Sports Sciences School of Rio Maior, Polytechnic Institute of Santarém, Portugal

[2] Research Center for Sport, Health and Human Development (CIDESD), UTAD, Vila Real, Portugal

[3] PLUX- Biosignal Acquisition and Processing, Lisboa, Portugal

[4] Departament of C. of Sport, University of Beira Interior, Covilhã, Portugal

[5] Department of C. of Sport, Exercise and Health of University of Trás-os-Montes and Alto Douro; Vila Real, Portugal.

Introduction

In swimming, the snorkel (K4b^2, Italy, Rome), which consists of a valve train Aquatrainer (Cosmed, Rome, Italy), is often used for analysis of various physiological and biomechanical aspects [1,2]. Researchers analyzed its feability and reliability, and the mechanical constraints caused by this system [3]. Electromyography (EMG) is used to evaluate the neuromuscular activity, by plotting the electrical activity of the muscles, using the pattern of muscle activation as benchmark [4,5]. The purpose of this study is to compare the average pattern of muscle activation in two situations: using a snorkel and without the use of snorkeling in the breakstroke swimming technique.

Methods

5 male subjects (Mean ± SD: age 19 ± 3,67 years; weight 76.1 ± 6.58 kg; height 178 ± 0.05 cm; fat mass percentage 14,68 ± 1.96; IMC 24 ± 1,66), were subjected to a test consisting of a protocol of 2 x 25m breakstroke swimming. In the first part of the test the swimmers used a snorkel; in the second part they swam without snorkel, making each part to 95% of transit time for 200m crawl. Using a wireless signal acquisition system (bioPLUX research, Portugal) and EMG sensors (emgPLUX, Portugal), the muscle activity of Biceps Brachii (BB) and Triceps Brachii (TB) of the right arm was recorded throughout the test and synchronized with the video images. The raw EMG was processed offline using Python (version 2.4) routines to compare morphology of the pattern of EMG signal recorded form BB and TB during both test conditions. The signals were sub-sampled to a frequency of 200Hz, low-pass filtered with a smoothing window of 50 samples and rectified. We selected the(middle-700_middle+2300) samples of the raw signal on all identical pathways(15m). For each subject, muscle and test condition, the mean, standart deviation, maximum and minimum values fo EMG were determined. In order to compare the pattern EMG wave of the swimming movement with and without snorkel, the mean EMG wave was computed for each subject, muscle and test condition.

Results

The results demonstrated that the mean (EMG) of the BB and TB are higher with the use of snorkel, thus showing greater activation during the action cycles in this implementation. Looking at the maximum value of EMG activation, it was possible to see that BB muscle has higher values than TB and both muscles presents higher maximum values with the use of a snorkel. The minimum values are also higher in the BB in both situations.

Discussion

We can observe that both muscles, BB and TB, present higher values with the use of snorkel. The BB muscle is the one which shows higher values both with and without snorkel, meaning higher activation in both situations. The time of muscle activation is also bigger with the use of snorkel.
The curve of the EMG signal pattern of the cycles for each muscle group is different from subject to subject, and was different between each situation.

References

1.Fernandes, R., Cardoso, C.,Silva, J.,Vilar, S., Colaço, P.,Barbosa, T.,Keskinen,K., Vilas-Boas, JP.(2006).*Assessment of the time limita t lowest speed corresponding to maximal oxygen consumption in the four competitive swimming strokes*.In: Biomechanics and Medicine in Swimming X.Portuguese Journal of Sport Sciences, Porto.
2.Rodriguez, F. A., K. L. Keskinen, Et Al. (2008).*"Validity of a swimming snorkel for metabolic testing.*" Int J Sports Med 29(2): 120-8.
3.Costa, M., Reis, A., Reis, Vm., Silva, Aj.,Garrido, N.,Louro,H.,Marinho,Da.,Baldari, C., Barbosa, Tm.(2009). *Constraint caused by mechanical valve Aquatrainer associated with system oxymetry direct (K4B^2) in breakstroke kinematic.*3º Nacional Congress of Biomechanic, M.A.Vaz *et al* (Eds), Bragança.
4.Clarys, J. (1983). *A review of EMG in swimming: explanation of facts and/or feedback information*. In: A. Hollander, P. Huijing, G. Groot (eds), Biomechanics and medicine in swimming, pp. 123-135. Champaign, Illinois, USA.
5.Rouard, A., Billat, R., Deschodt, V.; Clarys, J.(1993). *Muscular activation in sweep movements of the upper limb in freestyle swimming.* In: H., Riehle, M., Vieten (Eds), *XIX I.S.B. Congress*, pp 781-782.