University of Bergamo

School of Doctoral Studies
Doctoral Degree in Engineering and Applied Sciences
XXXI Cycle
SSD: ING-INF/01 Electronics

Doctoral Thesis

Monitoring Techniques based on Wearable Inertial Platforms for Patients Affected by Central Nervous System Diseases

Supervisor:
Chiar.mo Prof. Valerio Re

Co-supervisor:
Prof. Gianluca Traversi

Author:
Patrick LOCATELLI
Student ID: 1007755

Academic Year 2017/2018

To all the people who believed in me...

Ringraziamenti

Il percorso che ha portato alla stesura di questa tesi di dottorato non sarebbe stato lo stesso senza il contributo di molte persone.

In primo luogo, ringrazio il mio tutor, il professor Valerio Re, per avermi dato l'opportunità di far parte del suo gruppo di lavoro. La libertà concessami nel definire il percorso di formazione che meglio si adattasse ai miei interessi, nonché la fiducia espressa nell'affidarmi responsabilità didattiche e progettuali, hanno favorito la crescita della mia persona. Oltre al mio relatore, vorrei ringraziare il professor Gianluca Traversi per il suo contributo alla revisione della tesi e per l'incoraggiamento, così come il professor Massimo Manghisoni e il dottor Luigi Gaioni per essere stati sempre disponibili al confronto di idee su temi spesso lontani dai loro ambiti di ricerca.

In secondo luogo, ringrazio tutti i colleghi del MicroLab, ex e attuali, con i quali ho convissuto e collaborato negli ultimi 6 anni. Ringrazio pertanto Michele, Daniele, Michael, Francesco e Benedetta, per avermi accompagnato dalla tesi di laurea magistrale fino ai primi mesi di dottorato. E ancora, ringrazio Andrea, Francesco, Mauro e Matteo, sia per il sostegno lavorativo e gli utili consigli (di carattere hardware/firmware, software, sportivo/alimentare e videoludico, rispettivamente) all'interno del laboratorio, sia per i momenti di svago e divertimento all'esterno.

Un ringraziamento va inoltre al dottor Dario Alimonti, ideatore di due dei progetti qui descritti. Il suo impegno ha contribuito al reclutamento dei primi soggetti e alla raccolta di dati preliminari per lo sviluppo della prima versione degli algoritmi.

Ringrazio di cuore i miei genitori, per non avermi mai fatto mancare nulla, per avermi dato l'opportunità di arrivare a questo traguardo, ma soprattutto per aver sempre creduto in me ed aver supportato le mie scelte. Un grazie anche a mio fratello e a tutti gli amici che in questi anni hanno fatto il tifo per me.

"Last but not least", voglio ringraziare la mia fidanzata Elisa, che oltre ad avermi incoraggiato ad intraprendere questa esperienza, mi ha accompagnato e sostenuto nei momenti di difficoltà, rendendo meno faticoso il mio cammino.

Contents

List of Figures

List of Tables

Introduction

This thesis presents the results of the development of wireless systems based on advanced sensor platforms and algorithms, for the remote monitoring of patients affected by central nervous system (CNS) diseases.

Central nervous system diseases are a broad category of neurological disorders affecting the structure or the functionality of the brain or spinal cord. Such conditions may be an inherited metabolic disorder; the result of damage from an infection, a degenerative condition, stroke, a brain tumor or other problems; or arise from unknown or multiple factors. Several types of CNS diseases exist, but they all share the loss of sufficient, intact nervous system circuits that orchestrate diverse functions such as memory formation (as in the case of Alzheimer's disease) or voluntary motion (such as in movement disorders). Multiple sclerosis (MS), Parkinson's disease (PD), and essential tremor are only some of the most common disorders causing the reduction of motor capabilities, yet their incidence and prevalence on the population are remarkable: more than 10 million people across the world are currently living with Parkinson's, and other 2.5 million people are affected by MS.

Nowadays, CNS diseases represent one of the major public health challenges: in the US alone, the direct costs of Alzheimer's disease was estimated to be 260 billion dollar in 2017, whereas the combined direct and indirect cost of Parkinson's was estimated to be nearly 25 billion dollar per year. To help with their management, it is important for doctors and people with these long-term conditions to understand how their disease changes over time: in fact, while most conditions in this group cannot be completely cured, symptoms of central nervous system diseases can often be managed through a range of therapies, from medical to surgical treatment. To this purpose, several assessment tests are used by clinicians to diagnose the disorders and, more importantly, to monitor their progress: however, although such tests are typically administered and performed in supervised environments (such as hospitals), most of the symptoms and episodes happen outside of the healthcare structures.

Thanks to advances in semiconductor and microelectronics technologies, the last

decade has seen an increasing diffusion of wireless inertial platforms, and with them an explosion in the capability of monitoring individuals via sensors integrated in smartphones and wearable devices. This kind of systems, in fact, can be used not only as consumer devices, but they can also be adopted for medical and rehabilitation purposes in the healthcare field. A huge amount of studies have been carried out in recent years to support and promote the use of such technologies as quantitative and objective tools to diagnose some specific diseases such as movement disorders, to monitor the disease progress, as well as the beneficial effects of pharmacological therapies and rehabilitation exercises, and to help clinicians in prescribing the most appropriate therapeutic programs.

Wearable technologies are typically used in two contexts, each coming with different considerations:

- In a supervised context (such as a clinic or an hospital), subjects are followed by trained personnel while performing the required tests, therefore there is a certain "guarantee" that the task was performed according to the planned protocol. Thus, by knowing *a priori* the test conditions, as well as the environment surrounding the test area, it is possible to exploit the wearable devices to extract many useful and disease-specific information.

- On the contrary, in an unsupervised context (such as home), subjects must be monitored while performing their everyday activities: this increases the probability of bumping into a scenario not replicable in clinic, for which the disease clearly shows itself. In some cases, the subject may be also instructed to perform some rehabilitation exercises at home. The main drawbacks of such context are the lack of knowledge, either about the conformity to the protocols, or about the environment surrounding the user while performing the prescribed tests. In this cases, thus, only general information about the subject activity can be actually inferred (e.g. total amount of activity, number of walked steps per day), and few or none features related to the specific disease can be extracted: nevertheless, such general information can still be very useful, depending on the goal of the study.

For these reasons, accurate methods for retrieving information from data need to be designed on the basis of the specific monitored disease and usage context. In addition, other than the definition of the analysis algorithms, the wearable-based systems need to meet specific requirements in order for them to be used as long-term, research-grade monitoring tools: among these requirements, the low power consumption and the high memory capacity are the most important ones.

The studies presented in this thesis involve the use of inertial platforms for the monitoring of patients affected by CNS diseases. Several analysis algorithms as well as a dedicated wearable platform were specifically designed to cover the entire range of the above mentioned use cases, in order to provide medical staff with objective and simple-to-use monitoring tools.

Basic concepts about the central nervous system are introduced in chapter 1. After an overview of its structure and functionality, the chapter presents the most common CNS diseases, with particular attention to multiple sclerosis and Parkinson's disease.

Chapter 2 reports the development of a reliable tremor classification system based on a research-grade wearable device. The chapter runs from the characterization of the wireless inertial platform as a vibration monitoring tool, to the design and test of methods to analyze data collected from patients affected by two of the most common types of tremors: parkinsonian and essential. Finally, the steps and the results of the classifier building process are presented.

In chapter 3, the inertial sensors integrated in a wearable device were exploited to assess the functional motor skills of subjects. Specifically, a body sensor network (BSN) composed of a variable number of nodes was used to monitor the subjects while performing some standard clinical motor tests, typically administered to patients affected by Parkinson's disease: the Extended Timed Up and Go test, the Six-Minute Walk Test and a set of stability tests. The goal of the study was to design efficient analysis procedures able to deal with data from both supervised and unsupervised environments.

On the other hand, by focusing only on the home monitoring, chapter 4 describes the RADAR-CNS project, a European research program aiming to remotely assess the disease progress by exploiting commercial wearable devices. The chapter illustrates the analysis plan designed to monitor the motor disabilities of patients affected by multiple sclerosis, as well as the author's personal contribution to the project in terms of developed algorithms and preliminary results obtained when tested on healthy subjects.

Lastly, chapter 5 presents a new wearable device named Winter, specifically designed to overcome the limitations of the platforms used in previous chapters. The chapter includes the design strategies and the description of the device components. Finally, results of preliminary performance tests are reported, in terms of power consumption and data compatibility.

Chapter 1

Central Nervous System Disorders

The nervous system is a complex collection of nerves and specialized cells known as neurons that transmit signals between different parts of the body: as such, it helps all parts of the body communicate effectively. Vertebrates have central (CNS) and peripheral (PNS) nervous systems: in particular, the former represents the integration and command center of the body, consisting of the brain, spinal cord and retina. Because the CNS controls and coordinates the major functions of the body, any lesion caused in the brain and spinal cord can lead to some dangerous and deadly diseases. Such disorders may be the results of an injury from trauma, but may also depend on underlying genetic conditions, metabolic derangement, or infection.

After a wide introduction to the nervous system's structure and functioning, the chapter focuses on the description of the central nervous system's role and parts. In addition, most common CNS diseases including multiple sclerosis (MS), Parkinson's disease and other disorders, are presented in terms of nature, diffusion and symptoms.

1.1 The Nervous System

The nervous system is the part of an animal that coordinates its actions by transmitting signals to and from different parts of its body. In vertebrates it consists of two main parts, the central nervous system and the peripheral nervous system: the former consists of the brain and spinal cord, whereas the latter consists mainly of nerves that connect the CNS to every other part of the body. Nerves that transmit signals from the brain are called motor or *efferent* nerves, while those nerves that transmit information from the body to the CNS are called sensory or *afferent*. Spinal nerves serve both functions and are called *mixed* nerves.

1.1.1 Functions

The primary function of the nervous system is to control the body. It does this by extracting information from the environment using sensory receptors, sending signals that encode this information into the central nervous system, processing the information to determine an appropriate response, and sending output signals to muscles or glands to activate the response.

At the most basic level, the function of the nervous system is to send signals from one cell to others, or from one part of the body to others. There are multiple ways of sending signals from a cell to others. One is by releasing chemicals called hormones into the internal circulation, so that they can diffuse to distant sites. In contrast to this "broadcast" mode of signaling, the nervous system provides "point-to-point" signals, making connections with specific target cells. Thus, neural signaling is capable of a much higher level of specificity than hormonal signaling. It is also much faster: the fastest nerve signals travel at speeds that exceed 100 m/s.

1.1.2 Structure

At the cellular level, the nervous system contains two main types of cells: neurons and glial cells.

Neurons

The nervous system is defined by the presence of a special type of cell, called *neuron*, also known as a "nerve cell" [1]. Neurons are highly specialized for the processing and transmission of cellular signals. A typical neuron (see Figure 1.1) is divided into three parts: the *soma* or cell body, *dendrites*, and the *axon*. The soma is the body of the neuron and contains its nucleus. It gives rise to numerous dendrites, filaments that extrude from it, branching profusely and extending few hundred micrometers from the soma. On the contrary, the axon can extend for great distances from the soma, giving rise to hundreds of branches. Axons typically extend throughout the body in bundles called *nerves*. Neurons communicate with other cells via *synapses*, which are membrane-to-membrane junctions containing molecular machinery that allows rapid transmission of signals [1]: synaptic signals from other neurons are received by the soma and dendrites; signals to other neurons are transmitted by the axon. Hundreds of different types of neurons exist, with a wide variety of morphologies and functions [2]: sensory neurons, for instance, transmute physical stimuli such as light and sound into neural signals, while motor neurons transmute neural signals into activation of muscles or glands. However the great majority

of neurons participate in the formation of centralized structures (the brain and ganglia) and they receive all of their input from other neurons and send their output to other neurons [1]. The human brain has a huge number of synapses. Each of the 10^{11} (one hundred billion) neurons has on average 7000 synaptic connections to other neurons. It has been estimated that the brain of a three-year-old child has about 10^{15} synapses (1 quadrillion). This number declines with age, stabilizing by adulthood. Estimates vary for an adult, ranging from 10^{14}–$5 \cdot 10^{14}$ synapses (100 to 500 trillion) [3].

Glial cells

Along with neurons, the nervous system contains other non-neuronal specialized cells called *glial cells* (named from the Greek for "glue"). In the human brain, it is estimated that the total number of glia roughly equals the number of neurons, although the ratio varies from one part of the brain to another [4]. Among the most important functions of glial cells are:

- To support neurons and hold them in place;
- To supply nutrients to neurons;
- To insulate neurons electrically;
- To destroy pathogens and remove dead neurons;
- To provide guidance cues directing the axons of neurons to their targets.

A very important type of glial cell (oligodendrocytes in the CNS, Schwann cells in the PNS) generates layers of a fatty substance called *myelin* that wraps around axons and provides electrical insulation, allowing them to transmit signals much more rapidly and efficiently.

1.1.3 Signals Propagation

To support the general function of the nervous system, neurons have evolved unique capabilities for intracellular signaling (communication within the cell) and intercellular signaling (communication between cells). Neurons have evolved special abilities for sending electrical signals (action potentials) along axons, allowing them to achieve long distances and perform rapid communication. This mechanism, called *conduction*, is how the cell body of a neuron communicates with its own terminals via the axon. Communication between neurons is achieved at synapses by the process of *neurotransmission*.

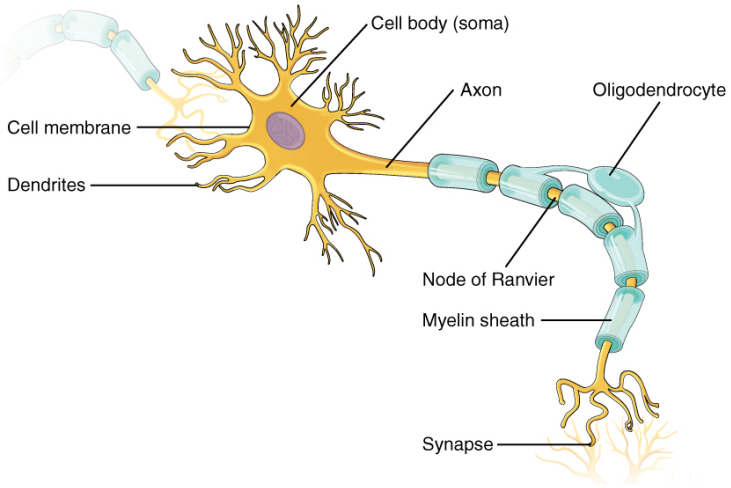


Figure 1.1 – Structure of a typical neuron. The electrical signal travels down the axon from the nucleus. The myelin sheath speeds up the neural impulse. Once it reaches the axon terminals, the synaptic junctions transfer the signal to other cells.
Source: "*Neurons and glial cells*", Figure 2. From OpenStax College, Biology [CC BY 3.0]

Conduction

To begin conduction, an *action potential* is generated near the soma, where the axon leaves the cell body. An action potential is an electrical signal very much like the electrical signals in electronic devices. But whereas an electrical signal in an electronic device occurs because electrons move along a wire, an electrical signal in a neuron occurs because ions move across the neuronal membrane.

All cells in animal body tissues are electrically polarized — in other words, they maintain a voltage difference across the cell's plasma membrane, known as the *membrane potential*. This electrical polarization is due to the fact that there is an excess of negative ions inside the cell membrane and an excess of positive ions outside. In addition, the concentration of ions differs from the inside to the outside of the cell, leading to an electrochemical gradient across the membrane: for instance, the concentration of potassium ions (K^+) is 30 times greater in the fluid inside the cell than outside, whereas the concentration of sodium ions (Na^+) is nearly 10 times greater in the fluid outside the cell than the inside (see Figure 1.2). The unequal distribution of Na^+ and K^+ is regulated by two energy-dependent protein structures embedded in the membrane: the voltage-gated *ion channels*, enabling the passive diffusion of specific ions across it; and the *ion pumps*, which actively pumps ions into and out of the cells against the concentration gradient (this process requires energy in form of ATP, adenosine triphosphate).

When the nerve cell is in its resting state, the voltage-gated Na^+ channels are

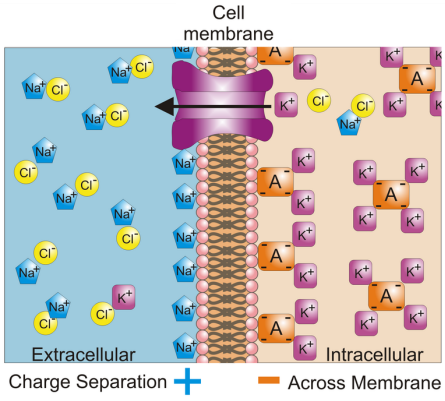


Figure 1.2 – Differences in the concentrations of ions on opposite sides of a cellular membrane lead to the *membrane potential*.
Source: *Wikimedia Commons*, by Synaptidude [CC BY 3.0]

closed, maintaining the unequal distribution. Due to both the excess of negative charge inside the cell membrane and the different ions concentration, the potential difference across the membrane is about −70 mV: this is the nerve cell's *resting potential*. If a physical or chemical stimulus is strong enough to cause depolarization from the resting potential of −70 mV to around −55 mV, the voltage-gated Na^+ channels open. Favored by both concentration and electric gradients, Na^+ ions flow into the cell, creating an electric current. The influx of Na^+ causes a local reversal of electric polarity of the membrane, changing the electric potential to about 30 mV. The Na^+ concentration gradient (outside greater than the inside) becomes balanced by the electric gradient (due to the membrane potential now having become positive on the inside), and the depolarization is complete at the site of the original stimulus. The Na^+ channels then close again. The K^+ channels respond to the changes in polarity, sending K^+ flowing out of the cell. The movement of K^+ ions and slower action of the Na^+-K^+ pump soon restore concentration differentials and electric gradient to the resting state. The transient change in the electric potential across the membrane is the action potential. After depolarization, for a brief period (milliseconds), the Na^+ channels cannot be stimulated: this is called the *refractory period*. Figure 1.3 depicts the entire action potential generation process.

The local depolarization at the site of the original stimulus causes a passive diffusion of ions into areas adjacent to the site of the stimulus. The potential in the adjacent area reaching the threshold level of −50 mV, increases the permeability of Na^+ and propagates the action potential of 40 mV in a wavelike manner along the length of the nerve cell. Because of the refractory period, the nerve impulse can only be propagated in one direction, away from the nerve cell body towards the terminal branches.

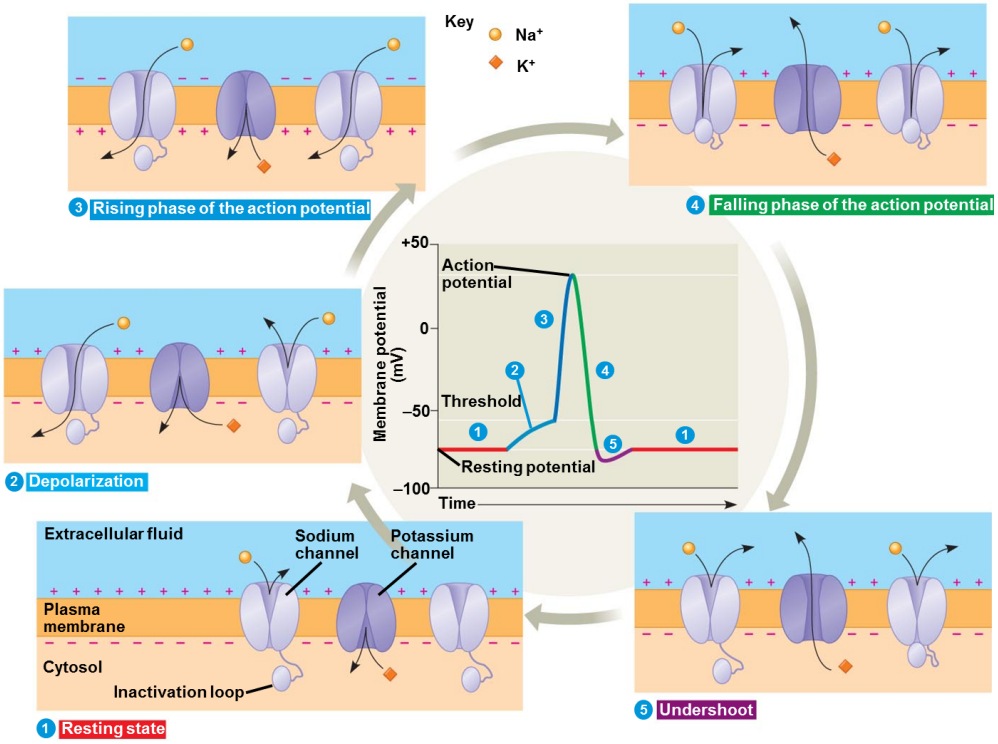


Figure 1.3 – Generation of an action potential. From the resting state (1), an external stimulus increases the membrane potential up to the threshold (2). The cell depolarizes and the Na^+ channels open, generating an action potential (3). The K^+ channels then open, lowering the concentration of K^+ ions in the cytosol and thus the membrane potential. Finally, the Na^+-K^+ pump restores the normal ions concentration.
Source: *Biology in Focus*, "*Chapter 37: Neurons, Synapses, and Signaling*", Figure 37.11. © 2014 Pearson Education, Inc.

Myelin sheath Propagation of the action potential along the axon is dependent on how fast the membrane is able to charge, which is determined by the membrane capacitance and the axial resistance (how resistant the interior axonal medium is to electrical current): the lower these parameters, the higher the conduction velocity. The reduction in the internal axial resistance may be achieved by increasing the diameter of the axon, which reduces the resistance that Na^+ ions must face as they passively spread through the axon [5]. While increasing axonal diameter is a viable solution for rapid axonal conduction, it also comes at a price in terms of energy consumption and space constraints [6]. In many neurons, axons are thus surrounded by a discontinuous multilayered covering, called *myelin sheath*, providing both insulation of electric current and metabolic support for the axon. The myelin sheath is formed by glial cells and normally runs along the axon in segments about 1 mm long, punctuated by unsheathed nodes of Ranvier about 1 µm wide (see Figure 1.1).

During the myelination process the voltage-gated Na^+ channels cluster to regions adjacent to the myelinating segment, resulting in a high concentration at the nodes of Ranvier and a lack of Na^+ channels along the myelinated internodes. Such migration reduces the leakage of Na^+ into the extracellular fluid (ECF), maintaining a strong separation of charge between the intracellular fluid (ICF) and the ECF at myelinated segments (i.e., reduction in axial resistance). This increases sodium's ability to travel along the axon. However, the sodium diffuses in a rapid but decremental way, meaning that it cannot trigger the opening of other voltage-gated sodium channels as it becomes weaker. The large amounts of Na^+ channels contained in the nodes of Ranvier, located about every 1 mm, allow enough sodium into the axon to regenerate the action potential to its original value. For this reason in myelinated fibers, impulses "hop" or propagate by *saltatory conduction* (see Figure 1.4). Whereas unmyelinated axon conduction velocities range from about 0.5 to 10 m/s, myelinated axons can conduct at velocities up to 150 m/s.

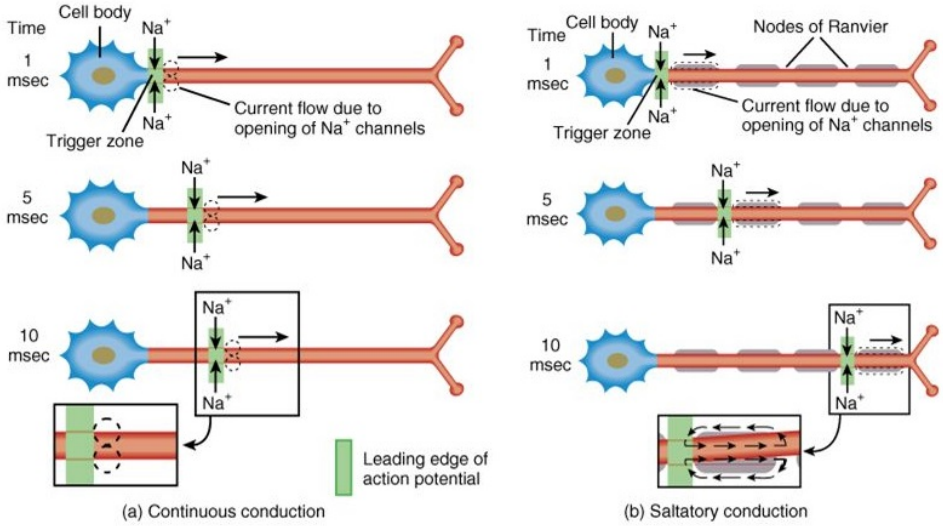


Figure 1.4 – Action potential conduction in (a) unmyelinated axon and in (b) myelinated axon.
Source: Lecture notes on *BIOLOGY 251 Human Anatomy & Physiology*, "*Chapter 12: The Nervous System*". © 2015 John Wiley & Sons, Inc.

Neurotransmission

The action potential is propagated along the axon until it reaches its terminals: here conduction ends and neurotransmission begins. Neurotransmission (or synaptic transmission) is the communication process between neurons as accomplished by the movement of chemicals or electrical signals. This process takes place at synapses,

close point contacts between neurons and other neurons or other types of excitable efferent cells (e.g. muscles or gland cells). The vast majority of synapses in the nervous system are axo-dendritic synapses (i.e., axon synapsing upon a dendrite), however, a variety of other arrangements exist. A *presynaptic* neuron is the one that fires the signal as a result of an action potential entering its axon terminal. In contrast, a *postsynaptic* neuron is a neuron that receives the signal after it has crossed the synapse: it may experience an action potential if the stimulus is strong enough.

When positive ions depolarize the postsynaptic membrane, an excitatory postsynaptic potential (EPSP) occurs. Postsynaptic neurons work through *spatial summation* and *temporal summation* (see Figure 1.5). The former means that when many subthreshold EPSPs, originated from different presynaptic neurons, all stimulate the neuron simultaneously at the trigger zone, they will sum up to create a suprathreshold stimulus and an action potential will be generated. On the other hand, the latter means that multiple EPSPs from a single presynaptic neuron will sum up as well to create a suprathreshold potential, if they reach the trigger zone close enough together in time.

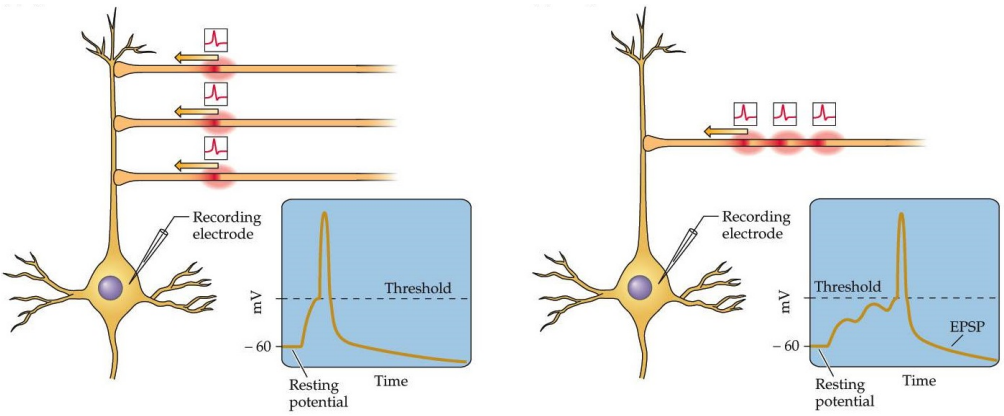


Figure 1.5 – Spatial summation (left) and temporal summation (right) of excitatory postsynaptic potentials (EPSPs).
Source: *The Mind's Machine* second edition, Figure 3.10 (edited). © 2016 Sinauer Associates, Inc.

Electrical synapses At electrical synapses, the two synaptic partner cells are tightly joined together, so tight that specialized proteins in their membranes can dock to each other and form a pore from cell to cell: this is comparable to the docking devices of rockets attaching to the International Space Station, forming a gate through which astronauts and materials can pass. In electrical synapses, these docking devices between pre- and postsynaptic cells are called *gap junctions*. The

pore of a gap junction channel is wide enough to allow ions to flow from one cell to the next, thereby connecting the two cell's cytoplasm. Thus when the membrane potential of one cell changes, ions may rapidly move through from one cell to the next, carrying positive charge with them and depolarizing the postsynaptic cell. The response in the postsynaptic neuron is in general smaller in amplitude than the source: the amount of attenuation of the signal is due to the membrane resistance of the presynaptic and postsynaptic neurons.

Chemical synapses Chemical neurotransmission occurs at chemical synapses. In this process, the presynaptic neuron and the postsynaptic neuron are separated by a small gap — the synaptic cleft, filled with extracellular fluid. Although very small, typically on the order of a few nanometers, the synaptic cleft creates a physical barrier for the electrical signal carried by one neuron to be transferred to another neuron. Chemical synapses contain little membrane vesicles filled with a chemical substance called *neurotransmitter*: acting like a chemical messenger, it links the action potential of one neuron with a synaptic potential in another. An action potential causes vesicles to migrate to the axon terminal, where they "dock" to the membrane. When the nerve impulse reaches the synapse, the membranes of the docked vesicles fuse with the presynaptic membrane, exposing their interior to the outside space. In this way, the neurotransmitter is released into the synaptic cleft and can now diffuse towards the postsynaptic membrane where some specialized ion channels called transmitter receptor proteins are harbored. When transmitter binds to them, this triggers their ion channels to open and let sodium ions in, depolarizing the postsynaptic membrane. If this depolarization is strong enough to open voltage-gated ion channels, a nerve impulse is triggered in the postsynaptic cells. Finally, neurotransmitter is removed from the synaptic cleft and re-absorbed by presynaptic membrane transporters, allowing its recycle.

Transmission at chemical synapses is far more complicated and slow than in electrical ones, because they translate the nerve impulse into a chemical message, to then revert it into a nerve impulse. Nonetheless, the majority of synapses in the nervous system is chemical rather than electrical. The reason lies in the enormous ability of the nervous system to process information, learn and adapt.

- Firstly, chemical synapses do not simply pass the message on, but they can change its meaning. For instance, an action potential reaching an excitatory synapse triggers an action potential in the postsynaptic cell. In contrast, an action potential reaching an inhibitory synapse prevents postsynaptic cells

from firing action potentials. Inhibitory synapses are therefore a means to tone down activity in neuronal networks.

- Secondly, chemical synapses can change the quality of information transfer over time by modifying different steps in the nerve impulse to chemical signal conversion. For example, when being used a lot, synapses can adapt by desensitization (e.g. to selectively filter background noise during a conversation) or by enhancing synaptic transmission (referred to as "increasing synaptic strength", for example during learning processes). Synaptic strength can be increased through lowering the threshold for transmitter release, increasing the volume of transmitter being released per event, raising the number or the opening probability of transmitter receptors: any of these changes makes transmission at these synapses far more effective and likely, therefore changing the way in which an incoming action potential is passed on to other cells.

1.2 The Central Nervous System

The central nervous system (CNS) is contained within the dorsal body cavity and consists of two major structures: the brain and spinal cord. The CNS is so named because it integrates information it receives from, and coordinates and influences the activity of, all parts of the bodies of bilaterally symmetric animals and it contains the majority of the nervous system. There are a number of differences between the central and the peripheral nervous systems. One difference is the size of the cells: the nerve axons of the CNS are much shorter. PNS nerve axons can be up to 1 meter long (for instance, the nerve that activates the big toe) whereas, within the CNS, they are rarely longer than a few millimeters. Another major difference between the CNS and PNS involves regeneration (regrowth of cells). Much of the PNS has the ability to regenerate: if a nerve in a finger is severed, it can regrow. The CNS, however, does not have this ability.

In vertebrates, the brain is encased in the skull, and protected by the cranium. The spinal cord is continuous with the brain and lies caudally to the brain, and is protected by the vertebrae. The spinal cord reaches from the base of the skull, continues through or starting below the *foramen magnum* (a large oval opening in the occipital bone of the skull), and terminates roughly level with the first or second lumbar vertebra, occupying the upper sections of the vertebral canal. Cerebrospinal fluid surrounds the brain and the spinal cord and also circulates within the cavities of the CNS. Both the brain and the spinal cord are enveloped by three protecting membranes, called *meninges*. The cerebrospinal fluid also circulates between the

first two meningeal layers, called the *pia matter* and the *arachnoid*. The outer, thicker layer serves the role of a protective shield and is called the *dura matter*.

Areas of the CNS that are mainly made up of myelinated axons are called "white matter", because of the white coloration of the myelin. It affects brain functions by modulating the distribution of action potentials and coordinating communication between its different regions. In contrast, the "gray matter" refers to areas that mainly contain numerous cell bodies and relatively few myelinated axons. It includes regions of the brain involved in muscle control and sensory perception.

1.2.1 Brain

The brain is the most complex organ in the human body, controlling all functions of the body and interpreting information from the outside world. Intelligence, creativity, emotion and memory are a few of the many things governed by the brain. In total, around 100 billion neurons and 1000 billion glial cells make up the human brain, using around 20 % of the body's total energy. The brain is the central control module of the body and coordinates activity: from physical motion to the secretion of hormones, the creation of memories, and the sensation of emotion. To carry out these functions, some sections of the brain have dedicated roles. However, many higher functions — reasoning, problem-solving, creativity — involve different areas working together in networks. Protected within the skull, the brain is composed of 3 main structural divisions:

Cerebrum The largest part of the brain, composed of right and left hemispheres. It performs high functions like interpreting touch, vision and hearing, as well as speech, reasoning, emotions, learning, and fine movement control.

Cerebellum Located under the cerebrum, its function is to coordinate muscle movements, maintain posture, and balance.

Brainstem It acts as a relay center connecting the cerebrum and cerebellum to the spinal cord. It performs many automatic functions such as breathing, heart rate, body temperature, wake and sleep cycles, digestion and coughing.

The cerebrum contains the cerebral cortex, as well as several subcortical structures, including the *hippocampus* and the *basal ganglia*. The right and left hemispheres are joined by a broad band of nerve fibers called *corpus callosum* that transmits messages from one side to the other. Each hemisphere controls the opposite side of the body: if a stroke occurs on the right side of the brain, the left arm or leg may be weak or paralyzed. Not all functions of the hemispheres are shared. In general,

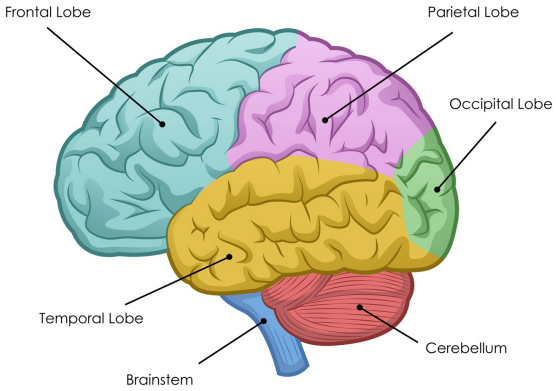


Figure 1.6 – Left hemisphere of a human brain. The four lobes composing the cerebrum have been highlighted with different colors.
Source: www.timvandevall.com. © Dutch Renaissance Press LLC

the left hemisphere controls speech, comprehension, arithmetic and writing. The right hemisphere controls creativity, spatial ability, artistic, and musical skills. The left hemisphere is dominant in hand use and language in about 92 % of people.

The cerebral hemispheres have distinct fissures, dividing the brain into lobes (see Figure 1.6). Each hemisphere has four lobes, and each of them may be divided once again into areas that serve very specific functions. It is important to underline that each lobe does not function alone: there are very complex relationships between the lobes of the brain and between the right and left hemispheres.

Frontal lobe It is located in the front of the brain and contains the majority of dopamine-sensitive neurons. It is responsible for voluntary movement and it is involved in attention, short-term memory, motivation, and planning.

Parietal lobe It is located behind the frontal lobe and above the occipital lobe. It processes sensory information deriving from the skin, such as temperature, pain and touch. The processing also includes information about spatial sense and navigation (proprioception).

Temporal lobe It is positioned on both sides of the brain. It is involved in processing sensory input into derived meanings for the appropriate retention of visual memories, language comprehension and emotion association.

Occipital lobe It is located at the back of the brain. Housing the visual cortex, it is the visual processing center of the brain.

1.2.2 Spinal cord

The spinal cord is a long, thin, tubular bundle of nervous tissue extending from the brain. Running almost the full length of the back, the spinal cord is surrounded by the vertebral bodies that form the spinal column and carries information between the brain and the peripheral nervous system.

The central structures of the spinal cord are made up of gray matter (nerve cell bodies), and the external or surrounding tissues are made up of white matter. From the brainstem, where the spinal cord meets the brain, the human spinal cord is made of 31 segments: 8 cervical segments, 12 thoracic segments, 5 lumbar segments, 5 sacral segments and 1 coccygeal segment. One pair of sensory nerve roots and one pair of motor nerve roots branches from the back and the front of each nerve segment (see Figure 1.7). The roots then merge into bilaterally symmetrical pairs of spinal nerves, which carry motor, sensory, and autonomic signals between the spinal cord and the PNS. Motor commands from the brain travel from the spine to the muscles, whereas sensory information travels from the sensory tissues — such as the skin — toward the spinal cord and finally up to the brain.

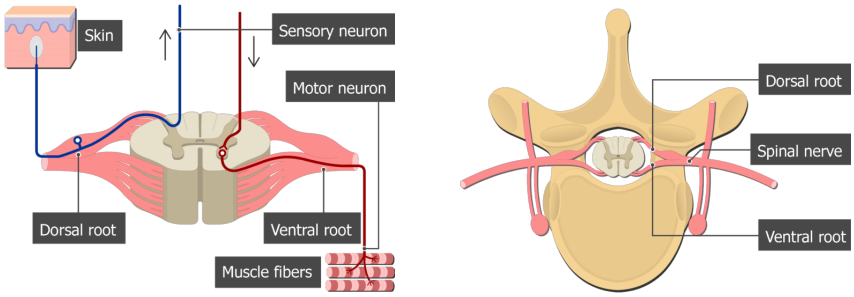


Figure 1.7 – On the left, signal propagation through the spinal nerve. Ventral roots (front) propagate the motor command, whereas the sensory information travels back through dorsal roots (back). On the right, the position of the spinal nerve with respect to a vertebra.
Source: www.getbodysmart.com. © GetBodySmart 2018

The spinal cord contains circuits that control certain reflexive responses, such as the involuntary movement your arm might make if your finger was to touch a flame. The circuits within the spine can also generate more complex movements such as walking. Even without input from the brain, the spinal nerves can coordinate all of the muscles necessary to walk. For instance, if the brain of a cat is separated from its spine so that its brain has no contact with its body, it will start spontaneously walking when placed on a treadmill. The brain is only required to stop and start the process, or make changes if, for instance, an object appears in your path.

1.3 CNS Diseases

Central nervous system diseases (or CNS disorders) are a group of neurological disorders that affect the structure or function of the brain or spinal cord [7]. There are several causes associated with CNS diseases. Any type of traumatic brain injury or injury done to the spinal cord, for instance, can result in a wide spectrum of disabilities in a person. Infections and tumors may affect the brain and the spinal cord directly. Degenerative spinal disorders (involving a loss of function in the spine) and brain degeneration also causes CNS diseases. Every disease comes with different signs and symptoms, such as persistent headache, memory loss, spasticity and seizures. In this section, the following diseases will be described in more detail:

- Multiple sclerosis;
- Parkinson's disease;
- Alzheimer's disease;
- Stroke.

1.3.1 Multiple Sclerosis

Multiple sclerosis (MS) is a chronic demyelinating disease of the central nervous system, first described in 1868 by Jean-Martin Charcot. In this disease, the insulating covers of nerve cells in the brain and spinal cord are damaged, disrupting the ability of parts of the nervous system to communicate. The course of MS may be considered as the expression of two clinical phenomena: relapses of acute neurological symptoms, which end with a partial or complete remission; and progression, which refers to the steady and irreversible worsening of symptoms and signs over time that produces disability accumulation. Life expectancy is on average 5 to 10 years lower than that of an unaffected population [8].

Epidemiology

Multiple sclerosis is the most common autoimmune disorder of the central nervous system, affecting more than 2.3 million people worldwide. The prevalence of MS is not uniform throughout the world: it is very common in European and North American Caucasian populations (with a prevalence of 80 and 8.3 per 100 000 per year, respectively), and it is very rare among Asian and African ones, with less than 3 cases per 100 000 per year [9]. Several studies have shown a direct proportionality between disease risk and latitude: the prevalence is in fact higher

in North America and Northern Europe, whereas it is lower in subtropical and tropical regions [8]. According to recent studies, Italy is in a high-risk area for multiple sclerosis, with a total of 109 000 patients with multiple sclerosis [10]. The worldwide incidence is three times greater in females than in men, probably due to the increased susceptibility to immunological and inflammatory factors. The disease usually appears in adults in their late twenties or early thirties but it can rarely start in childhood and after 50 years of age [9].

Pathophysiology

MS involves the loss of oligodendrocytes, the cells responsible for creating and maintaining the myelin sheath which helps the neurons carry electrical signals (see section 1.1.3) [8]. Normally, a person's nervous system would be inaccessible to the white blood cells due to a specific mechanism called *blood–brain barrier*. This highly selective semipermeable membrane separates the circulating blood from the brain and extracellular fluid in the CNS. Magnetic resonance imaging has shown that when a person is undergoing an MS "attack", this barrier has broken down in a section of the brain or spinal cord, allowing T lymphocytes to enter into the brain. Once they get there, the T cells recognize myelin as foreign and attack it: this results in a thinning or complete loss of myelin and, as the disease advances, the breakdown of the axons of neurons. The attack of myelin also starts inflammatory processes, which triggers other immune cells and the release of soluble factors like antibodies and other destructive proteins. Up to date, the autoimmune mechanism that leads T cells to attack the myelin is still unknown.

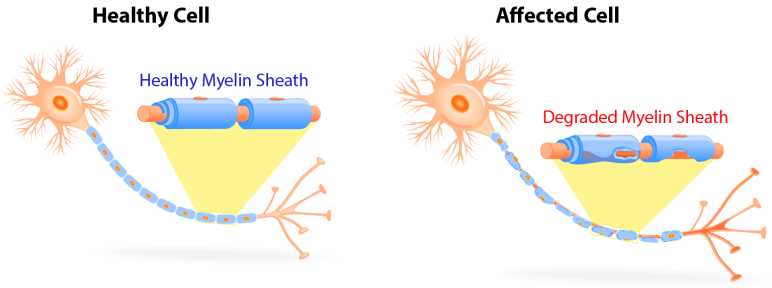


Figure 1.8 – Demyelination process. Normally, T cells protect the body by attacking invaders such as viruses. In MS, some T cells attack and damage the myelin sheath of neurons.
Source: www.pemf-tech.com. © PEMF-TECH 2018

When the myelin is lost, a neuron can no longer effectively conduct electrical signals. A repair process, called remyelination, takes place in early phases of the disease, but the oligodendrocytes are unable to completely rebuild the cell's myelin

sheath [11]. Repeated attacks lead to successively less effective remyelinations, until a scar-like plaque is built up around the damaged axons. These multiple scars (*sclerae*, better known as plaques or lesions) give name to the disease and are the origin of the symptoms: they prevent the saltatory conduction of the action potentials along the axons, leading to slowing down or block of the neuron conductivity. The lesions most commonly affect the white matter in the optic nerve, brain stem, basal ganglia, and spinal cord.

Clinical features

The progress of the MS considerably varies from person to person. Several phenotypes, or pattern of progression, have been described: these types use the past course of the disease in an attempt to predict the future course. Currently, the Multiple Sclerosis International Federation describes four types of MS [12].

Clinically isolated syndrome (CIS) In CIS, a person has an attack suggestive of demyelination, but does not fulfill the criteria for multiple sclerosis. 30 to 70 % of people experiencing CIS later develop MS.

Relapsing-remitting MS (RRMS) 85 % of persons affected by MS begin with this type. It is characterized by unpredictable relapses followed by periods of months to years of relative quiet (remission) with no new signs of disease activity. Deficits that occur during attacks may either resolve (*benign* MS) or leave problems, the latter in about 40 % of attacks and being more common the longer a person has been affected by the disease. It usually begins with a CIS.

Primary progressive MS (PPMS) It is characterized by progression of disability from onset, with no, or only occasional and minor, remissions and improvements. The usual age of onset for the primary progressive subtype is later than of the relapsing-remitting subtype (around 40 years of age).

Secondary progressive MS (SPMS) It occurs in around 65 % of those with RRMS, who eventually show a progressive neurologic decline between acute attacks with no periods of remission. The typical length of time between disease onset and conversion from RRMS to SPMS is 19 years.

From a disability perspective, after 10 years from the begin of the disease around 30 % of patients shows a moderate-serious disability. After the disease turns into its progressive form, after 5 years around 75 % of the patients is still able to walk,

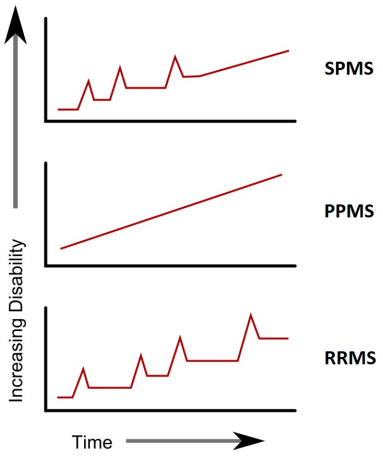


Figure 1.9 – Phenotypes of multiple sclerosis. From bottom to top, the relapsing-remitting, the primary progressive and the secondary progressive MS clinical course.
Source: [12]

but the percentage drop below 40 % after 15 years. The most common MS signs and symptoms are the following.

Visual functions Visual system abnormalities are frequent and often represent the only symptom at the beginning. The most frequent symptoms are a sensation of loss of the visual acuity that develops within a few days due to an involvement of the optic nerve.

Sensory functions The main sensory symptoms are paresthesia, tingling and numbness. A characteristic symptom is the *Lhermitte's sign*, a sensation of electric shock that spreads along the vertebral column and the upper limbs triggered by a bending of the cervical spine.

Motor functions The most typical manifestation is hyposthenia (lack of strength), which can manifest itself as a monoparesis, hemiparesis or paraparesis and evolve into paralysis. Another common symptom is spasticity, linked to the increase in muscle tone due to lesions of the cortico-spinal pathways, which can cause pain, retractions, spasms and considerably interfere in the daily life activities.

Fatigue It is one of the most common and disabling symptoms, defined as an overwhelming lack of energy and a sense of exhaustion. It affects about 85 % of those with MS.

Cerebellar functions The most frequent symptoms are dysmetria of the limbs, tremors, and ataxia, i.e., the lack of voluntary coordination of muscle movements, which has an important influence on the gait.

Rating scale

MS is characterized by a highly variegated symptoms spectrum: it is easy to understand how, being a disease of the CNS white matter, it can affect any functional system and show signs and symptoms corresponding to the different areas affected. Because of this extreme variability, a very precise evaluation and classification system is necessary to evaluate or monitor the possible flare-up or progression of the disease. The gold standard for disability assessment in MS is the Expanded Disability Status Scale (EDSS). It is a quantitative rating scale, with scores ranging from 0 (no signs and symptoms) to 10 (death due to MS), with the possibility of half a point between a finite value and the other. The total score is determined by the ability to walk and the level of autonomy for each functional system [13].

EDSS scores from 4.0 to 7.5 are primarily based on gait dysfunction. Initially, this is based on the distance a patient can reasonably walk without having to stop. Beginning with EDSS score of 6.0, the scale makes use of both distance and the aids necessary for ambulation. At EDSS 7.5, the ability to walk is essentially lost. The EDSS has several limitations, including the inadequate assessment of relevant domains as fatigue and cognition, its ordinal nature, and relatively poor sensitivity to change.

1.3.2 Parkinson's Disease

Parkinson's is a neurodegenerative disease, with a slow but progressive evolution, which mainly involves some functions such as movement control and balance. The disease is part of a group of diseases known as *movement disorders*, and among them it is the most frequent. The name is related to James Parkinson, a nineteenth-century London pharmacist surgeon who first described large part of the symptoms of the disease in a famous booklet, *An Essay on the Shaking Palsy*.

Epidemiology

Parkinson's disease (PD) is one of the most common neurodegenerative diseases in humans; the second for frequency after Alzheimer's dementia and its variants.

A 2015 Global Burden of Disease Study estimated the prevalence of Parkinson's to be approximately 6.2 million people worldwide, including all races and cultures [14]. Nevertheless, the figure may be considerably higher since many people go undiagnosed. The age of onset is usually above 60 years and the incidence of the disease increases with age, with about 3 % of the population over 80 years old; however, it is estimated that one in 10 people is diagnosed with Parkinson's disease before the age of 50; men are slightly more affected than women. In industrialized countries, the prevalence in the general population is estimated to be 120–180 cases per 100 000 individuals, while around 20 new cases per 100 000 people per year are estimated [15]. For demographic and social reasons, it is expected that in the next few decades the share of the older population of Western countries will be destined to increase [16, 17]. Since neurodegenerative diseases are directly bound to the age of the people [18], it follows that in the same period of time the world will face a significant increase in these pathologies.

Pathophysiology

The pathophysiology of Parkinson's disease is the death of dopaminergic neurons as a result of changes in biological activity in the brain with respect to Parkinson's disease. There are several proposed mechanisms for neuronal death in PD, including changes in cell metabolism and neuroinflammation. The basal ganglia are the brain areas more affected in Parkinson's disease: they are a group of subcortical nuclei situated at the base of the cerebrum, strongly interconnected with the cerebral cortex, the thalamus and the brainstem. Housing three out of the four major *dopaminergic pathways* in the brain, the basal ganglia participate in the control of voluntary movements, besides associative learning and cognition.

When the cerebral cortex, generally involved in decision making and planning, determines that a particular movement will be executed, it sends activating signals to motor cortices: in turn, they send signals to the basal ganglia to refine the choice of muscles that will participate in the movement and to amplify the activities that will drive the muscle contractions. One of the main components of the basal ganglia, the *substantia nigra*, is characterized by a high concentration of dopaminergic neurons — i.e., neurons that synthesize or contain a neurotransmitter called dopamine. Electrical stimulation of the substantia nigra activates the supply of dopamine to another basal ganglia component, the *striatum*, through the so-called nigrostriatal pathway. Its activation influences two pathways (see Figure 1.10):

Direct pathway: it facilitates the initiation and execution of voluntary movements. In normal condition, dopamine neurons excite this pathway, whose

net effect is to allow the activation of the thalamus which, in turn, sends activating signals to the motor cortices. These events amplify motor cortical activity that will eventually drive muscle contractions.

Indirect pathway: it helps to prevent unwanted muscle contractions from competing with voluntary movement. In normal condition, dopamine neurons inhibit this pathway, whose net effect is to send more inhibitory signals to the thalamus: in this way, the development of significant activity in the motor cortices is inhibited, and thus the activation of motor cortical areas that would compete with the voluntary movement is prevented.

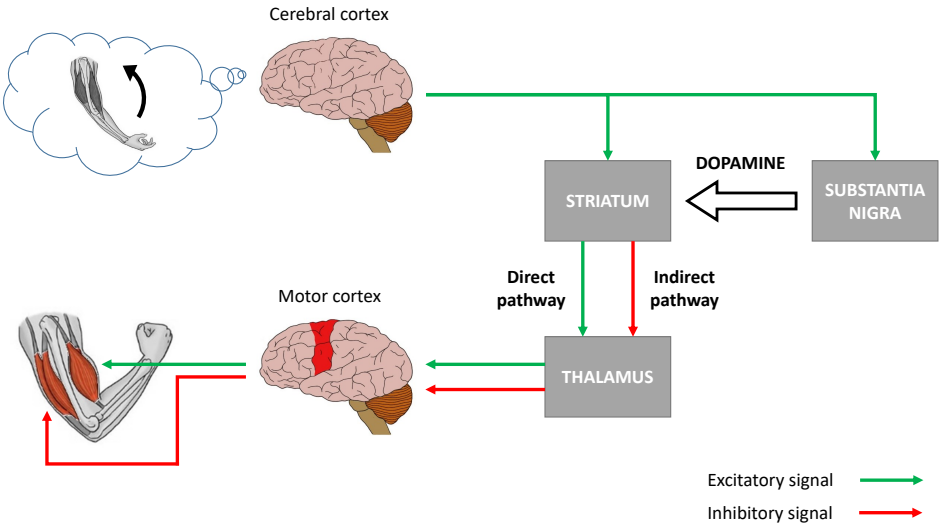


Figure 1.10 – Initiation of a voluntary movement. The cerebral cortex activates the striatum and the substantia nigra, which supply the former with dopamine. Both the direct pathway and the indirect pathway are influenced, determining the contraction of the biceps (agonist muscle) and the relaxation of the triceps (antagonist muscle) in order to perform an arm flexion.

The first major proposed cause of neuronal death in Parkinson's disease is the bundling of proteins. The alpha-synuclein protein has increased presence in the brains of PD patients and, since α-synuclein is insoluble, it aggregates to form the so-called Lewy bodies in neurons. Lewy bodies are widely recognized as a pathological marker of Parkinson's disease. Recent studies suggest that Lewy bodies are not the main cause of cell death in Parkinson's disease, as traditionally were thought; instead, they lead to other effects that cause cell death [19]. As the disease progresses, Lewy bodies mainly develop in the substantia nigra. Since α-synuclein is toxic, its accumulation impairs the cellular functions and interferes with microtubular transport, causing the death of the dopamine-producing neurons

of the substantia nigra. The consequent drop of dopamine in the striatum results in both a decreased activity in the direct pathway, and an increased activation of the indirect one: these effects lead to the simultaneous contraction of both agonist and antagonist muscles, resulting in the typical motor deficits of Parkinson's disease including tremors, rigidity, and postural imbalance.

Clinical features

There are five major pathways in the brain connecting other brain areas with the basal ganglia. These are known as the motor, oculo-motor, associative, limbic and orbitofrontal circuits, with names indicating the main projection area of each circuit. All of them are affected in PD, and their disruption explains many of the symptoms of the disease, since these circuits are involved in a wide variety of functions, including movement, attention and learning [20]. The classical motor symptoms of Parkinson's disease include bradykinesia, muscular rigidity, rest tremor, and postural and gait impairment. Empirical clinical observations suggest two major Parkinson's disease subtypes: tremor-dominant PD (with a relative absence of other motor symptoms) and non-tremor-dominant PD. Course and prognosis of disease differ between the subtypes; tremor-dominant Parkinson's disease is often associated with a slower rate of progression and less functional disability than non-tremor-dominant Parkinson's disease [21].

Non-motor features include olfactory dysfunction, cognitive impairment, psychiatric symptoms, sleep disorders, pain, and fatigue. These symptoms are common in early PD and are associated with reduced health-related quality of life. Non-motor features are also frequently present in Parkinson's disease before the onset of the classical motor symptoms. This premotor phase of the disease can be characterized by impaired olfaction, depression, excessive daytime sleepiness, and rapid eye movement sleep behavior disorder (RBD). The premotor phase can be prolonged; for example, the average latency between onset of RBD and occurrence of parkinsonian motor symptoms is 12–14 years. The process that causes Parkinson's disease is presumed to be underway during the premotor phase; thus, this period provides a potential temporal window during which disease-modifying therapy, once it becomes available, could be administered to prevent or delay the development and progression of disease.

Progression of Parkinson's disease is characterized by worsening of motor features, which initially can be managed with symptomatic therapies. However, as the disease advances, there is an emergence of complications related to long-term symptomatic treatment, including motor and non-motor fluctuations, dyskinesia, and psychosis.

In late-stage Parkinson's disease, treatment-resistant motor and non-motor features are prominent and include axial motor symptoms such as postural instability, freezing of gait, falls, and speech dysfunction. After about 17 years of disease, up to 80 % of patients with PD have freezing of gait and falls, and up to 50 % of patients report choking. Dementia is particularly prevalent, occurring in 83 % of patients with PD who have had 20 years disease duration. These levodopa-resistant symptoms of late-stage Parkinson's disease contribute substantially to disability and are strong predictors of a need for admission to an institution [21].

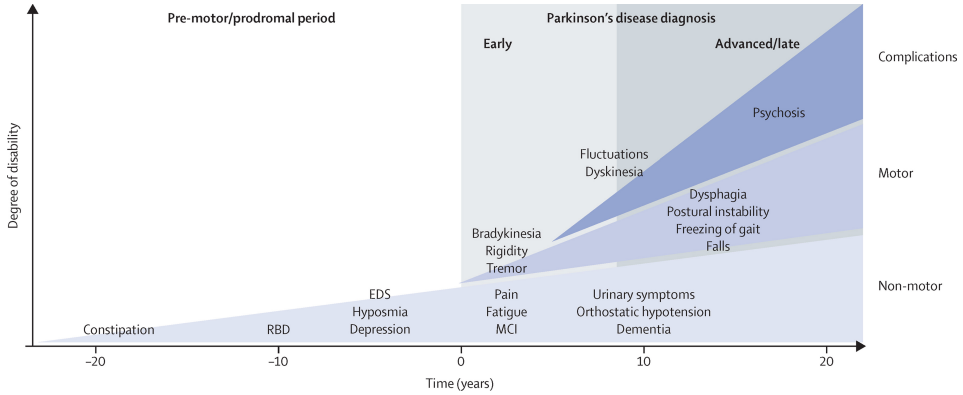


Figure 1.11 – Clinical symptoms and time course of Parkinson's disease progression.
Source: [21], Figure 1.

Diagnosis and rating scales

Nowadays, no tests can clearly identify the Parkinson's disease: the doctor can diagnose its presence starting from the patient's medical history and a neurological examination. Exams such as computerized tomography (TC) and magnetic resonance imaging (MRI) are often useful to rule out other diseases that can give rise to similar symptoms. In fact, some "parkinsonism", i.e., disorders produced by Parkinson's disease, can be caused by other diseases such as cerebral stroke, Alzheimer's disease, vascular tumors and pathologies of the basal ganglia. The difference between the syndromes lies in the fact that those deriving from Parkinson's disease present a temporal evolution, therefore by means of a differential diagnosis it is possible to establish the nature of the disorder.

The Hoehn and Yahr scale (H&Y), published in 1967 by Melvin Yahr and Margaret Hoehn, is commonly used to assess the progress of Parkinson's disease. The original version included five stages of the disease course; later two intermediate stages were introduced. Table 4.1 shows the description of the staircase stages currently in use. In 1987 a new system was introduced, combining the elements

of different scales of Parkinson's disease (including the H&Y scale), becoming an efficient and flexible tool to monitor the impact of the disease and the level of caused disability: this system is known as Unified Parkinson's Desease Rating Scale (UPDRS), and today it is the most widely used scale for the assessment of the disease. The UPDRS system consists of 5 sections:

Part I Assessment of cognitive abilities, behavior and tone mood.

Part II Self-assessment of daily activities (ADL, Activity of Daily Life) including handwriting, dressing, walking, hygiene, falls, swallowing, salivation, cutting food.

Part III Clinical evaluation of motor skills.

Part IV Assessment of any complications related to drug therapy.

Part V Assessment of the severity of the disease according to the Hoehn and Yahr scale.

This scale, however, attributed a relatively low importance to the non-motor characteristics of the disease. For this reason, in 2007 the Movement Disorder Society (MDS) published a revision of the UPDRS system known as MDS-UPDRS. This scale has not yet replaced the previous one in clinical practice, but has begun to be used in clinical trials.

1.3.3 Other CNS diseases

Movement disorders

Movement disorders are clinical syndromes with either an excess of movement or a lack of voluntary and involuntary movements, unrelated to weakness or spasticity. Recurring causes of movement disorders are neurodegenerative diseases, drug induction, ischemia and CNS infections. In primary movement disorders, the abnormal movement is the primary manifestation of the disorder. In secondary movement disorders, the abnormal movement is a manifestation of another systemic or neurological disorder.

Movement disorders are conventionally divided into two major categories: *hyperkinetic* and *hypokinetic*. Hyperkinetic movement disorders refer to dyskinesia, or excessive, often repetitive, involuntary movements that intrude upon the normal flow of motor activity. Cramps, tremors, dystonia and cerebral palsy are typical examples of hyperkinetic movement disorders. On the opposite, hypokinetic movement disorders refer to akinesia (lack of movement), hypokinesia (reduced

amplitude of movements), bradykinesia (slow movement) and rigidity. Tremors — the most common movement disorder — will be widely described in section 2.1.

Alzheimer's disease

Alzheimer's disease (AD), also known simply as Alzheimer's, is a chronic neurodegenerative disease characterized by progressive cognitive deterioration together with declining activities of daily living and neuropsychiatric symptoms or behavioral changes. The disease was first described by, and later named after, German psychiatrist and pathologist Alois Alzheimer in 1906. Alzheimer's is the most common cause of *dementia*, a general term for memory loss and other cognitive abilities, serious enough to interfere with daily life. Alzheimer's disease accounts for 60 % to 80 % of dementia cases [22].

Alzheimer's disease is characterized by loss of neurons and synapses in the cerebral cortex and certain subcortical regions. This loss results in gross atrophy of the affected regions, including degeneration in the temporal and parietal lobes, and parts of the frontal cortex. Studies using MRI and PET have documented reductions in the size of specific brain regions in people with AD as they progressed from mild cognitive impairment to Alzheimer's disease, and in comparison with similar images from healthy older adults. Two abnormal structures called plaques and tangles are prime suspects in damaging and killing nerve cells: the former are deposits of a protein fragment (called *beta-amyloid*) that build up in the spaces between nerve cells, whereas the latter are twisted fibers of another protein (called *tau*) that build up inside cells (see Figure 1.12). Though autopsy studies show that most people develop some plaques and tangles as they age, those with Alzheimer's tend to develop far more and in a predictable pattern, beginning in the areas important for memory before spreading to other regions. It is still unknown exactly what role plaques and tangles play in Alzheimer's disease. Most experts believe they somehow play a critical role in blocking communication among nerve cells and disrupting processes that cells need to survive.

The most common early symptom of the AD is loss of short-term memory (amnesia), which usually manifests as minor forgetfulness that becomes steadily more pronounced with illness progression, with relative preservation of older memories. As the disorder progresses, cognitive impairment extends to the domains of language (aphasia), skilled movements (apraxia), and recognition (agnosia), and functions such as decision-making and planning become impaired.

Alzheimer's disease most often begins in people over 65 years of age, although 4 % to 5 % of cases are early-onset Alzheimer's which begin before this [23]. According

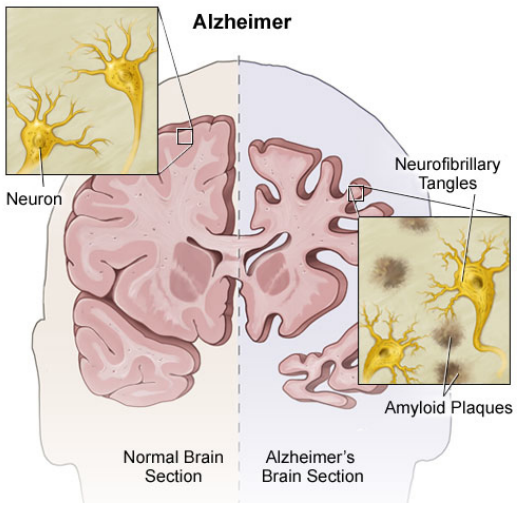


Figure 1.12 – Differences between normal brain and Alzheimer's brain. Plaques and tangles are the structures that interfere with neurons.
Source: *Stanford Healthcare*, Alzheimer's Disease.

to the World Alzheimer Report 2016 [24], there were 46.8 million people worldwide living with dementia in 2015 and this number will reach 131.5 million in 2050. Moreover, in 2015 dementia resulted in about 1.9 million deaths [25]. Cohort longitudinal studies provide rates between 10 and 15 per thousand person–years for all dementias and 5–8 for AD [26]. An estimated 5.7 million Americans of all ages are living with Alzheimer's dementia in 2018: this number includes an estimated 5.5 million people age 65 and older and approximately 200 000 individuals under age 65 who have younger-onset Alzheimer's.

Stroke

According to the World Health Organization (WHO), a *stroke* is a "neurological deficit of cerebrovascular cause that persists beyond 24 hours or is interrupted by death within 24 hours" [27]. Cerebral stroke is the most frequent neurological disease: it is a medical emergency, and must be promptly diagnosed and treated in a hospital because of the high risk of disability and death that it entails. A stroke can be classified into two different categories: ischemic and hemorrhagic. The former is caused by an interruption of the blood flow, whereas the latter is the result of a rupture of a blood vessel. About 87 % of strokes are ischemic, the rest are hemorrhagic.

Ischemic stroke is defined as an acute cerebrovascular disorder caused by the sudden occlusion of a cerebral artery, with consequent ischemia of that cerebral area. In this situation, the blood supply to a part of the brain is decreased, causing

a series of dysfunctions to the affected brain tissues. Nerve cells are able to survive only for few minutes when the influx of oxygenated and glucose-rich blood to the brain is arrested: if the flow is not restored in time, a series of metabolic processes are triggered, leading to the brain death in a short time. The clinical manifestations of an ischemic stroke are highly variable and depend exclusively on the location and extent of the ischemic lesion. This category includes strokes that might happen for the following reasons:

- Trombosis (the obstruction of a blood vessel by a blood clot forming locally);
- Embolism (obstruction due to an embolus from elsewhere in the body);
- Systemic hypoperfusion (general reduction of blood flow throughout the body);
- Cerebral venous thrombosis.

On the other hand, hemorrhagic stroke is determined by the presence of an intracerebral hemorrhage, located mainly in the deep hemispheric part. This hemorrhage is the result of a rupture of a blood vessel, usually in small arteries and arterioles. The hematoma expands and compresses the surrounding tissues, which are in turn damaged and dislocated. This pressure can lead to a lack of blood supply to the affected tissues resulting in an infarct. Furthermore, the blood released by the cerebral hemorrhage appears to have toxic effects on vessels and tissues of the brain. The most frequent causes of vessel rupture are trauma, hypertensive crisis, rupture of cerebral aneurysms of arteriovenous malformations.

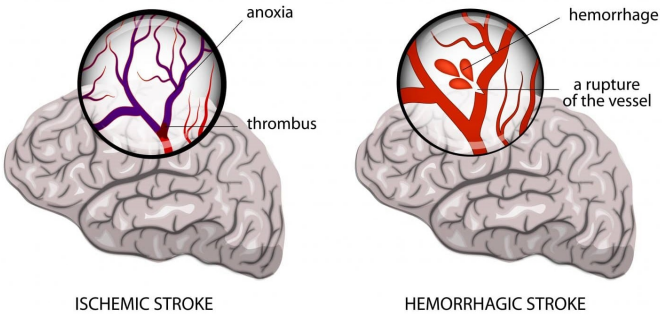


Figure 1.13 – On the left, an ischemic stroke, caused by a trombosis. On the right, an hemorrhagic stroke, caused by a rupture of a vessel.
Source: *The Stroke Lawyers*, Hemorrhagic and ischemic stroke.

Cerebral stroke was the second most frequent cause of death worldwide in 2016, accounting for nearly 6 million deaths [28]. Between 1990 and 2010 the number of strokes decreased by approximately 10 % in the developed world and increased

by 10 % in the developing world. The rate of incidence increases exponentially with increasing age: it ranges from 1.5 to 4.0 per 100 000 people per year on the population of all ages, reaching 8.5 in the population between 65 and 85, up to an incidence of 20–35 per 100 000 people per year over 85. Stroke risk factors include advanced age, elevated blood pressure, previous strokes, diabetes, high cholesterol, smoking, and atrial fibrillation.

References

[1] E. Kandel, J. Schwartz, and T. Jessel, "Nerve cells and behavior", in *Principles of Neural Science*, McGraw-Hill Professional, 2000, ch. 2, ISBN: 978-0-8385-7701-1.

[2] ——, "The cytology of neurons", in *Principles of Neural Science*, McGraw-Hill Professional, 2000, ch. 4, ISBN: 978-0-8385-7701-1.

[3] D. Drachman, "Do we have brain to spare?", *Neurology*, 2005. DOI: 10.1212/01.WNL.0000166914.38327.BB.

[4] F. A. Azevedo, L. R. Carvalho, L. T. Grinberg, *et al.*, "Equal numbers of neuronal and nonneuronal cells make the human brain an isometrically scaled-up primate brain", *Journal of Comparative Neurology*, 2009. DOI: 10.1002/cne.21974.

[5] S. A. Freeman, A. Desmazières, *et al.*, "Mechanisms of sodium channel clustering and its influence on axonal impulse conduction", *Cellular and Molecular Life Sciences*, 2016. DOI: 10.1007/s00018-015-2081-1.

[6] D. K. Hartline and D. R. Colman, "Rapid conduction and the evolution of giant axons and myelinated fibers", *Current biology*, 2007. DOI: 10.1016/j.cub.2006.11.042.

[7] (2013). Nervous System Diseases, [Online]. Available: https://www.healthdirect.gov.au/nervous-system-diseases.

[8] A. Compston and A. Coles, "Multiple sclerosis", *The Lancet*, vol. 372, 2008. DOI: 10.1016/S0140-6736(08)61620-7.

[9] World Health Organization, *Atlas: Multiple Sclerosis Resources in the World 2008*. Geneva : World Health Organization, 2008, pp. 15–16, ISBN: 92-4-156375-3.

[10] M. Battaglia and D. Bezzini, "Estimated prevalence of multiple sclerosis in italy in 2015", *Neurological Science*, 2017. DOI: 10.1007/s10072-016-2801-9.

[11] D. Chari, "Remyelination In Multiple Sclerosis", in *The Neurobiology of Multiple Sclerosis*, ser. International Review of Neurobiology, vol. 79, Academic Press, 2007, pp. 589–620. DOI: 10.1016/S0074-7742(07)79026-8.

[12] F. D. Lublin, S. C. Reingold, *et al.*, "Defining the clinical course of multiple sclerosis", *Neurology*, vol. 83, no. 3, pp. 278–286, 2014. DOI: 10.1212/WNL.0000000000000560. [Online]. Available: http://n.neurology.org/content/83/3/278.

[13] J. Kurtzke, "Rating neurologic impairment in multiple sclerosis: an expanded disability status scale (EDSS)", *Neurology*, 1983.

[14] V. L. Feigin *et al.*, "Global, regional, and national burden of neurological disorders during 1990–2015: A systematic analysis for the global burden of disease study 2015", *The Lancet Neurology*, 2017. DOI: 10.1016/S1474-4422(17)30299-5.

[15] C. Ferrarese *et al.*, *Core curriculum. Malattie del sistema nervoso.* 1st ed. McGraw-Hill Italia, 2011, ch. 14, ISBN: 9788838639678.

[16] *U.S. Census. Projections of the population by selected age groups and sex for the United States: 2010 to 2050: national population projections*, Washington, DC: U.S. Census Bureau (Population Division), U.S. Department of Commerce, 2011.

[17] World Health Organization. (2012). Who: Ageing and life course, [Online]. Available: http://www.who.int.

[18] J. V. Hindle, "Ageing, neurodegeneration and Parkinson's disease", *Age and Ageing*, vol. 39, no. 2, pp. 156–161, 2010. DOI: 10.1093/ageing/afp223. [Online]. Available: http://dx.doi.org/10.1093/ageing/afp223.

[19] A. H. Schapira, "Etiology and pathogenesis of parkinson disease", *Neurologic Clinics*, 2009. DOI: 10.1016/j.ncl.2009.04.004.

[20] J. A. Obeso, M. C. Rodríguez-Oroz, B. Benitez-Temino, F. J. Blesa, J. Guridi, C. Marin, and M. Rodriguez, "Functional organization of the basal ganglia: Therapeutic implications for parkinson's disease", *Movement Disorders*, 2008. DOI: 10.1002/mds.22062.

[21] L. V. Kalia and A. E. Lang, "Parkinson's disease", *The Lancet*, 2015. DOI: 10.1016/S0140-6736(14)61393-3.

[22] Alzheimer's Association. (2018). What Is Alzheimer's?, [Online]. Available: https://www.alz.org/alzheimers-dementia/what-is-alzheimers.

[23] M. F. Mendez, "Early-onset Alzheimer's Disease: Nonamnestic Subtypes and Type 2 AD", *Archives of Medical Research*, vol. 43, no. 8, pp. 677–685, 2012, ISSN: 0188-4409. DOI: 10.1016/j.arcmed.2012.11.009.

[24] A. D. International. (2016). World Alzheimer Report 2016, [Online]. Available: https://www.alz.co.uk/research/WorldAlzheimerReport2016.pdf.

[25] "Global, regional, and national life expectancy, all-cause mortality, and cause-specific mortality for 249 causes of death, 1980–2015: a systematic analysis for the Global Burden of Disease Study 2015", *The Lancet*, vol. 388, no. 10053, pp. 1459–1544, 2016. DOI: 10.1016/S0140-6736(16)31012-1.

[26] A. Di Carlo, M. Baldereschi, L. Amaducci, *et al.*, "Incidence of Dementia, Alzheimer's Disease, and Vascular Dementia in Italy. The ILSA Study", *Journal of the American Geriatrics Society*, vol. 50, no. 1, pp. 41–48, DOI: 10.1046/j.1532-5415.2002.50006.x.

[27] World Health Organization, *Cerebrovascular disorders: a clinical and research classification.* Geneva : World Health Organization, 1978, ISBN: 9789241700436.

[28] ——, (2018). The top 10 causes of death, [Online]. Available: http://www.who.int/en/news-room/fact-sheets/detail/the-top-10-causes-of-death.

Chapter 2

Wearable-based Tremor Classifier

Essential tremor is the most common form of tremor presenting an outpatient neurology practice and yet it may be often difficult to differentiate with tremor in Parkinson's disease. Precise diagnoses are recommended since using appropriate medication is fundamental for efficacy and avoiding serious side effects. In this sense, a sensitive and specific imaging tool is represented by the Single Photon Emission Computerized Tomography (SPECT) of the dopamine transporter, but it is expensive and not advisable as screening means. For this reason, usage of different technologies such as the ones based on wearable devices are widespread: in fact, these tools are developing such effective and affordable supports for clinicians.

In this chapter the development of a portable and reliable tremor classification system based on a wearable device, enabling clinicians to differentiate between essential tremor and Parkinson's disease associated tremor, is reported. Starting with the definition of most common tremors, the chapter includes the entire development process: from the characterization of the selected wearable device, up to the data collection and the realization of the classification model. Part of the material reported in this chapter has been published by the author of this thesis in *2017 7th IEEE International Workshop on Advances in Sensors and Interfaces (IWASI)* [1] and in *2018 7th International Conference on Modern Circuits and Systems Technologies (MOCAST)* [2].

2.1 Definition and Classification

Movement disorders are clinical syndromes presenting either an excess of movement or a lack of voluntary and involuntary movements [3]. Among the hyperkinetic movement disorders, tremors are the most common: they are defined as involuntary muscles contraction and relaxation causing oscillations of one or more body parts around a joint axis.

Tremors are caused by the alternating or synchronous activation of the antagonist muscle groups and usually involve hands, head, face, vocal cords, trunk or legs. They can be pathological or physiological: the latter, usually barely perceptible, becomes evident in many people during physical or mental stress. Tremors differ in presentation mode (e.g. intermittent or costant), acuity (e.g. gradual or abrupt) and severity. This latter factor may not be related to the severity of the underlying disease: for instance, essential tremor is generally considered benign and does not reduce life expectancy, but symptoms may be disabling.

Since the percentage of elderly people in the Western Countries is going to increase over the coming decades [4, 5], tremors related with aging are becoming a critical aspect for the improvement of the daily-life conditions of the elderly people. In this sense, essential tremor and parkinsonian tremor are the most common age-related movement disorders presenting an outpatient neurology practice [6, 7].

2.1.1 Types of Tremor

The tremor is generally described based on the frequency of the oscillations (high or low) and the amplitude of the movement (fine or coarse). The tremor can be classified by its activation condition, that is, which position most accentuates it.

Resting tremor It is visible when the subject is in a rest condition and occurs when a part of the body has a complete support against gravity (the weight lies on a stable resting point). It is minimal or absent during activity.

Contraction tremor It manifests during supported contraction, e.g. a tight fist while the arm is resting and supported.

Postural tremor It occurs during maintenance of a position against gravity, e.g. with the arms elevated such as in a "bird-wing" position.

Action (kinetic) tremor It occurs during voluntary movement and it intensifies when the subject tries to use the affected muscles.

Intention tremor It manifests as a marked increase in tremor amplitude during a terminal portion of targeted movement, e.g. as the patient's finger approaches their nose.

According to their nature, tremors may also be divided in physiological and pathological. The physiological tremor occurs in otherwise healthy people. It is a quiet rapid (8–12 Hz) action or postural tremor that tends to strike both hands almost to the same extent. It could manifest because enhanced by fatigue,

anxiety, or the use of certain drugs, or in some pathological conditions such as hyperthyroidism. On the other hand, pathological tremor is caused by a pathology and can be described as a primary disorder (essential tremor, Parkinson's disease), or secondary to a disease (e.g. stroke).

2.1.2 Tremor Syndromes

In addition to type, tremors may be classified according to both their origin and their clinical features. Signs and symptoms depend on tremor type and cause.

Essential Tremor Essential tremor (ET) is the most common form of tremor. It has been calculated that ET affects around 2.2 % of the general population in the United States [8]. Although crude prevalence of this disorder varies considerably from study to study, essential tremor can occur at any age increasing in prevalence with it: 4–6 % among individuals aged 40 years and older and 6–9 % among individuals aged 60 years and older [8]. It involves limbs, head, voice, and is characterized by postural and kinetic tremors (see Figure 2.1), with a medium frequency (5–8 Hz). The tremor may range in severity from mild and functionally unimportant to severe and disabling. Essential tremor is a progressive disorder, showing slight annual increase in amplitude over time [6]. During the progression of the disease, many patients also have intention tremor that appears while approaching a target. Sometimes it appears even at rest, making it difficult to perform a differential diagnosis with parkinsonian tremors. Moreover, there are many clinical data on the apparent connections between essential tremor and Parkinson's disease: it has been observed that these patients are 2—4 times more likely to develop Parkinson's disease [9].

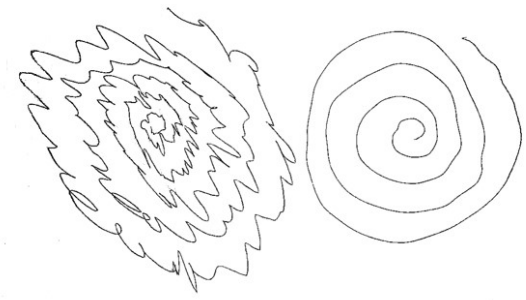


Figure 2.1 – Spirals drawn by a man suffering with unilateral essential tremor. The spiral on the left was drawn by the subject using his left hand, and the one on the right with his right hand.
Source: *Wikimedia Commons*, [CC BY 4.0]

Parkinsonian Tremor The second most common form of tremor is represented by parkinsonian tremor (PT), for the Parkinson's disease is one of the commonest neurodegenerative diseases in human beings (see section 1.3.2). Tremor is one of its cardinal features and it is often the presenting motor sign of PD, although alone it is not sufficient for the diagnosis; in fact, about 25 % of patients with PD never develop tremor (akinetic-rigid form). The typical parkinsonian tremor appears at rest with moderate amplitude and low frequency (4–6 Hz), and it is usually asymmetric (one side of the body is more affected than the other). It is generated by agonist and antagonist muscles activation alternating in a precisely tuned manner: this often leads to a stereotypical series of movements such as the typical pill-rolling tremor. Some patients may experience an additional postural tremor appearing immediately on stretching out the arms, usually asymmetrical with a slightly more rapid frequency (6–8 Hz) than resting tremor.

Other Tremors The work described in this chapter is focused on the above-mentioned syndromes because of their prevalence in the population and the patient's availability. Nonetheless, it is worth noting that other syndromes associated with tremors exist. The following list reports some of the better known forms of tremor.

Cerebellar tremor The intention tremor par excellence. It is a slow (2–7 Hz), broad tremor of the extremities that occurs at the end of a purposeful movement, such as trying to press a button. Cerebellar tremor is caused by lesions in or damage to the cerebellum resulting from stroke, tumor, or disease such as multiple sclerosis.

Dystonic tremor It occurs in individuals of all ages who are affected by dystonia, a movement disorder in which sustained involuntary muscle contractions cause twisting and repetitive motions and/or painful and abnormal postures or positions. Dystonic tremor is an action tremor, manifesting either postural or kinetic, and is not seen during complete rest. Its amplitude and frequency are irregular, usually under 7 Hz, and often variable.

Myoclonus It is a sudden, rapid, brief muscle jerks (positive myoclonus) or releases (negative myoclonus) of one or more motor units. As opposed to tremors, this movement disorder is arrhythmic, nevertheless sometimes it may be difficult to distinguish it from tremors only by clinical observation.

Orthostatic tremor It is characterized by fast (>12 Hz) rhythmic muscle contractions that occur in the legs and trunk immediately after standing. This

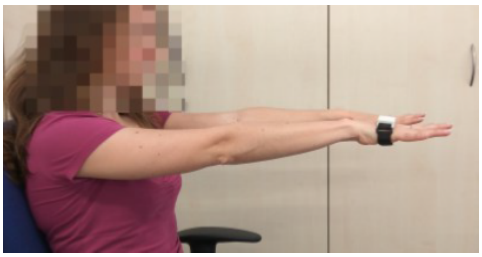
tremor may also occur in patients who have ET, and there might be an overlap between these categories of tremor.

2.1.3 Differentiating ET and PT

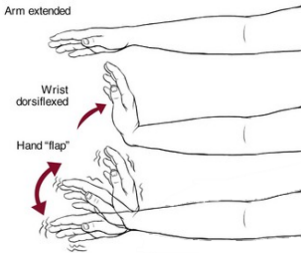
The approach to the diagnosis includes both the evaluation of the patient medical history and the physical examinations to determine whether the tremor occurs during action or at rest. In this sense, the tremor is typically assessed during the execution of four different standardized tasks (see Figure 2.2):

1. Rest task: the patient is asked to lay their arms onto the legs.
2. Postural task: the patient is asked to hold their arms parallel to the floor.
3. Postural/Kinetic task: starting from task 2, the patient is asked to move their hands up and down by rotating their wrist.
4. Kinetic task: the patient is asked to draw a spiral on a paper sheet.

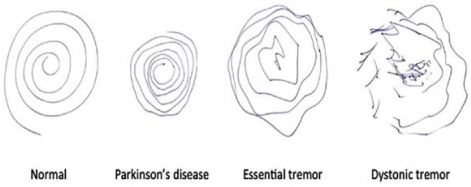
(a) Rest task.



(b) Postural task.



(c) Postural/Kinetic task.



(d) Kinetic task.

Figure 2.2 – Standardized tasks for tremor differentiation.

While the other forms of tremor are associated to specific neurological findings that may clearly differentiate them on the clinical basis, in some cases the similar amplitudes and the overlapping features characterizing ET and PT (see Figure 2.3) can make the differentiation a difficult task for clinicians, who can only rely on their past experience to classify and then properly treat the tremor. It is well-known that

no therapy is currently available slowing down the progression of PD or even prevent its manifestation, therefore all treatments available up to now are of symptomatic nature. Nonetheless, it remains fundamental to detect PD in its early phase because establishing a treatment effectively reduces or prevents disability and the need of support maintaining the patients' capability to work and to regain health-related quality of life. Treatment of ET provides for different medicinal products, though also for this movement disorder they are all symptomatic. Besides, PD drugs in ET patients are ineffective and expose them to potential and serious side effects.

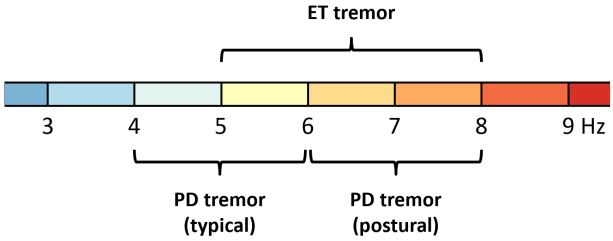


Figure 2.3 – Frequency intervals for different tremor syndromes.

Single photon emission computerized tomography (SPECT) of the dopamine transporter (DAT) — a marker of nigrostriatal degeneration — is a validated medical imaging support that helps clinicians in the diagnosis of patients with an uncertain degenerative parkinsonian syndrome [10, 11]. However, such a diagnostic tool is not viable for blanket screening both for its cost (about 600 €) and for the use of a radioactive labeled marker which is potentially noxious. Considering that the percentage of elderly people in the Western Countries will increase over the coming decades, and since both ET and PD are related with aging [6, 7], these two neurological conditions will significantly increase. It is therefore appropriate to investigate efficient and inexpensive methods for their diagnosis and management.

Recently, sensor-based tremor assessment has been performed in several studies. Alam et al. [12] quantified parkinsonian tremor severity by analyzing inertial data from wearable sensors. Surangsrirat et al. [13] classified PD tremors and ET on the basis of temporal fluctuations of the angular speed, which was collected by using a 3D gyroscope. Morrison et al. [14] examined the differences in postural tremor dynamics with respect to age and neurological disease by using four uniaxial accelerometers. In these studies, data from each subject were collected during the execution of different tasks: some studies relies on resting and kinetic tasks [12], others on resting and postural tasks [13], still others on resting tasks only [14]. Garcia et al. [15] designed an accelerometer-based system for hand tremor diagnosis and classification. Since their intention was to cover different types of tremor other than ET and parkinsonian tremor, they collected data in all the three above-

mentioned tasks (rest, postural and kinetic). Finally, Alam [12] and Surangsrirat [13] exploited machine learning techniques such as Support Vector Machine (SVM) in order to achieve a tremor classifier.

2.2 Inertial Platform: MuSe

The wearable system used in this project is the result of improvements made on the neMEMSi platform, a miniaturized multi-sensor device presented in different works [16, 17]. This new platform, named *MuSe*, has been designed by 221e srl and University of Bergamo in order to add functionalities capable for enhancing its operation, without incrementing the cost and the external volume.

2.2.1 Overview

MuSe is a miniaturized multi-sensor Attitude and Heading Reference System (AHRS) that integrates sensing, processing, storage and communication units in a single 25 mm × 25 mm form factor PCB. The device exploits some of the latest technologies in sensing and processing capabilities and power consumption.

- The processing core is the STM32F411 32-bit Cortex-M4 microcontroller by STMicroelectronics. It features a current consumption per MHz down to 100 µA/MHz and a floating point unit (FPU) core scalable at a frequency up to 100 MHz.

- The sensing unit is represented by the System-in-Package (SiP) LSM9DS1 geomagnetic module, manufactured by STMicroelectronics. It integrates a ±16 g (g-force) 3D accelerometer, a ±2000 dps gyroscope and a ±16 Gauss 3D magnetometer in a 3.5 mm × 3 mm package. Moreover, MuSe embeds two additional high performance modules, a ±4000 dps 3D gyroscope (ITG3701 by InvenSense) and a ±400 g 3D accelerometer (H3LIS331DL by STMicroelectronics) for high angular rates and accelerations detection.

- The wireless communication is provided by the SPBT2632C2A micro-sized Bluetooth V3.0 module by STMicroelectronics, which enables the use of the Serial Port Profile (SPP) for data transmission.

- Finally, the S25FL127S 128 Mbit serial NOR flash memory (Cypress Semiconductor) allows the user to perform continuous data log without the need of an active radio connection.

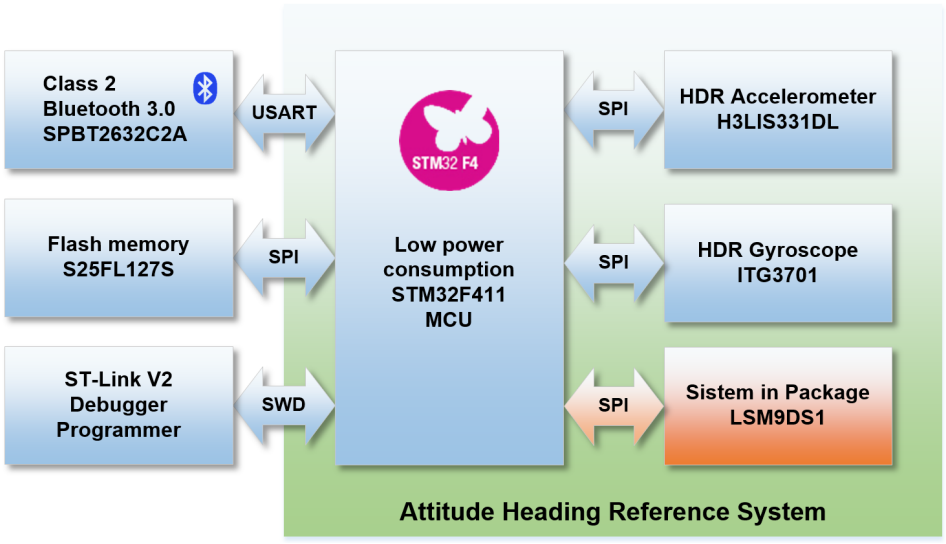


Figure 2.4 – Block diagram of the MuSe platform.

MuSe also integrates a calibration procedure compensating magnetic interferences deriving from both soft-iron and hard-iron elements. Processing results as well as raw data, can be real-time transmitted to a secondary device, or saved into the local memory and downloaded in a later time. Both data streaming and logging functionalities can be set to operate at up to 200 Hz.

2.2.2 Sensor Fusion

The AHRS features are achieved by means of a quaternion-based Kalman filter: this sensor fusion algorithm estimates the orientation of the platform by combining accelerations, angular rates and magnetic field measurements.

Orientation Representation

In mathematics, a rotation is a transformation of the Euclidean plane or space that moves objects rigidly and leaves at least one fixed point (the origin of space). The points that remain fixed in the transformation form a subspace: when this set is a point (the origin) or a straight line, these are called the center and the axis of the rotation respectively. There are several ways to represent a rotation in space, including *Euler angles* and *quaternions*.

Euler angles were introduced by the mathematician Leonhard Euler. They describe the position of an XYZ reference system integral with a rigid body by means of a series of rotations starting from a fixed reference system xyz, with which it shares the origin. In aeronautics and robotics a specific variant of Euler angles (called Tait-Bryan angles) is used, in which each angle assumes a particular name

based on the axis it is associated to:

- *Roll*, representing a rotation about the x axis of an angle φ;
- *Pitch*, representing a rotation about the y axis of an angle ϑ;
- *Yaw*, representing a rotation about the z axis of an angle ψ.

Although this orientation representation is very easy to understand, Euler angles rotation system are affected by a problem named *gimbal lock*, occurring when one of the degrees of freedom is lost: this leads to singularities, that is, orientations in space that are not uniquely described. For this reason, almost all of the 3D applications, graphics and game engines use quaternions to represent orientation.

In mathematics, quaternions are a numerical system that extends complex numbers. They were described for the first time by the Irish mathematician William Rowan Hamilton in 1843 and applied to mechanics in three-dimensional space. A quaternion is an expression of the form

$$a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k},$$

where a, b, c and d are real numbers, and $\mathbf{i}$, $\mathbf{j}$, and $\mathbf{k}$ are the fundamental quaternion units. By analogy with complex numbers, given a quaternion $q = a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k}$, the number a is the real part (or scalar part) of q, and $b\mathbf{i} + c\mathbf{j} + d\mathbf{k}$ is called the imaginary part (or vector part) of q. For this reason, a quaternion can also be expressed in the form:

$$q = (r, \vec{v}), \quad r \in \mathbb{R}, \vec{v} \in \mathbb{R}^3.$$

According to Euler's rotation theorem, any rotation or sequence of rotations of a rigid body or coordinate system is equivalent to a single rotation by a given angle ϑ around a fixed axis $\vec{u} = (u_x, u_y, u_z) = u_x\mathbf{i} + u_y\mathbf{j} + u_z\mathbf{k}$. Therefore, any rotation can be represented by a quaternion. Consecutive rotations can be mixed in a single rotation by applying the Hamilton product to their associated quaternions: this results in a less computational overhead in comparison to the composition of rotation by using Euler angles.

Kalman Filter

The sensor fusion algorithm embedded in the MuSe platform follows the same guidelines as the one embedded in its predecessor [18]. The algorithm is based on a state vector defined as follows:

$$x = \begin{bmatrix} q_w & q_x & q_y & q_z \end{bmatrix}^T, \tag{2.1}$$

where q_w is the real component and q_x, q_y and q_z are the imaginary components. The quaternion at time t is achieved during the prediction step by numerically integrating the angular rates provided by the gyroscope:

$$q_t = q_{t-1} + \dot{q}_t \, \Delta t, \tag{2.2}$$

where q_{t-1} is the previous step quaternion, Δt is the sampling period and $\dot{q}_t$ is the quaternion rate computed as the Hamilton product between q_{t-1} and the quaternion equivalent to the current angular rates reading $\omega_t = [0, \omega_x, \omega_y, \omega_z]^T$.

After the prediction step and the computation of the gain K_t, the current state $x_{t|t}$ is estimated by updating the predicted state $x_{t|t-1}$ with the observation vector $z_t\,(x_t)$, obtained by minimizing the function

$$\mathbf{e}(x_t) = \begin{bmatrix} x_t^* \otimes \mathbf{a_E} \otimes x_t - \mathbf{a_B} \\ x_t^* \otimes \mathbf{m_E} \otimes x_t - \mathbf{m_B} \end{bmatrix}, \tag{2.3}$$

which represents the differences between the known global acceleration and magnetic references ($\mathbf{a_E}$ and $\mathbf{m_E}$) represented in the body frame, and their measured values ($\mathbf{a_B}$ and $\mathbf{m_B}$).

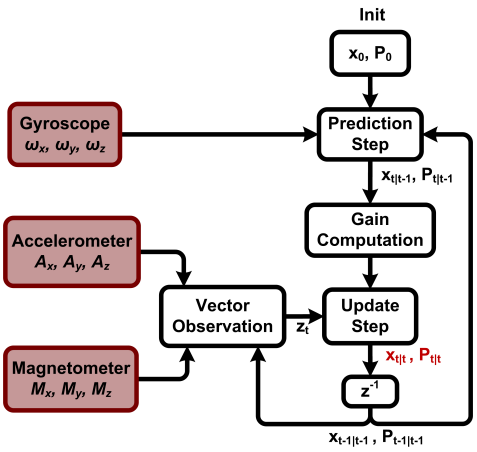


Figure 2.5 – Sensor fusion algorithm embedded in the MuSe platform, based on a quaternion-based Kalman filter.

2.3 Device Characterization

The MuSe platform was characterized to determine its reliability as a frequency measurement unit. For this purpose, the platform was mounted on the cylindrical piston of a Universal Testing Machine (UTM), able to vertically move following a specific waveform with selectable frequencies and amplitudes (see Figure 2.6). Since

the tremor has an undulatory behavior, the selected waveform was a sinusoid. Tests were performed with different frequencies and amplitudes combinations: the former ranged from 3 Hz to 8 Hz with a step of 0.5 Hz, whereas the latter ranged from 0.5 mm to 3 mm (leading to peak-to-peak amplitudes of 1 mm to 6 mm respectively) with a step of 0.5 mm. The piston accelerations were collected by the platform with a sampling rate of 50 Hz. For each frequency-amplitude combination, three different sessions with a duration of 10 seconds each were performed.

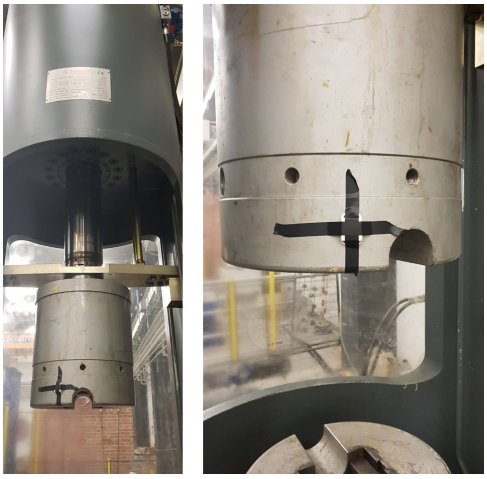


Figure 2.6 – The cylindrical piston of the UTM used for the characterization. The MuSe device has been fixed with black tape.

The analysis was performed by means of the MATLAB software. The vertical accelerations were imported and transformed into their equivalent frequency representation by means of the FFT algorithm. Before doing that, each acceleration signal was properly preprocessed. Firstly, the mean value was removed from the original vertical acceleration. Then, a Blackman-Harris window was applied to the detrended 500 sample-long signal. Finally, since the length of the FFT function input signal defines the frequency resolution, the windowed signal was zero-padded to obtain a total length of 2048 samples: this value of NFFT (i.e., the number of samples used in the FFT algorithm) leads to a frequency resolution of 0.024 Hz over the 0–25 Hz band. For each frequency spectrum, the amplitude and the frequency of the detected dominant peak were estimated. Table 2.1 and Table 2.2 report the estimated frequencies and amplitudes respectively, averaged over the three sessions: the missing values in the table depend on the specifications of the equipment, which cannot perform wide movements at high frequencies. The same analysis procedure was performed with different values of NFFT, in order to observe the mean estimation error of the dominant frequencies when using a smaller number of samples. For this purpose, a group of acquisitions with different frequency set points were selected. After the windowing operation, each signal was zero-padded in

order to obtain signals with three different number of total samples, i.e., 2048, 1024 and 512: then, the above-mentioned procedure was conducted for the estimation of the dominant frequencies. Table 2.3 reports the results of the analysis.

Table 2.1 – Mean frequencies (Hz) of the dominant frequencies.

Frequency (Hz)	**Piston Movement Amplitude (mm)**						**Marginal Mean Frequency (Hz)**
	0.5	**1**	**1.5**	**2**	**2.5**	**3**	
3	3.00 ± 0	3.00 ± 0	3.00 ± 0	3.00 ± 0	3.00 ± 0	3.00 ± 0	$\mathbf{3.00 \pm 0}$
3.5	3.50 ± 0.014	3.49 ± 0	3.49 ± 0	3.49 ± 0	3.49 ± 0	3.49 ± 0	$\mathbf{3.49 \pm 0.01}$
4	3.99 ± 0.01	4.00 ± 0	4.00 ± 0	4.00 ± 0	4.00 ± 0	4.00 ± 0	$\mathbf{4.00 \pm 0.01}$
4.5	4.50 ± 0.01	4.49 ± 0	4.49 ± 0	4.49 ± 0	4.49 ± 0	-	$\mathbf{4.49 \pm 0.01}$
5	5.00 ± 0	5.00 ± 0.01	5.00 ± 0	5.00 ± 0	-	-	$\mathbf{5.00 \pm 0.01}$
5.5	5.49 ± 0	5.50 ± 0.01	5.49 ± 0	5.49 ± 0	-	-	$\mathbf{5.50 \pm 0.01}$
6	6.00 ± 0.01	6.00 ± 0	6.00 ± 0	6.00 ± 0	-	-	$\mathbf{6.00 \pm 0.01}$
6.5	6.49 ± 0	6.49 ± 0	6.49 ± 0	-	-	-	$\mathbf{6.49 \pm 0}$
7	7.01 ± 0	7.01 ± 0	7.01 ± 0	-	-	-	$\mathbf{7.01 \pm 0}$
7.5	7.50 ± 0	7.50 ± 0	7.50 ± 0	-	-	-	$\mathbf{7.50 \pm 0.01}$
8	8.01 ± 0	8.01 ± 0	8.01 ± 0	-	-	-	$\mathbf{8.01 \pm 0}$

Table 2.2 – Mean amplitudes (a.u.) of the dominant frequencies.

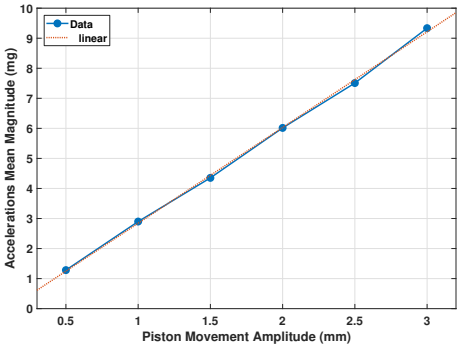
Frequency (Hz)	**Piston Movement Amplitude (mm)**					
	0.5	**1**	**1.5**	**2**	**2.5**	**3**
3	1.28 ± 0.07	2.90 ± 0.05	4.35 ± 0.09	6.02 ± 0.23	7.51 ± 0.06	9.34 ± 0.08
3.5	1.80 ± 0.03	4.15 ± 0.11	6.21 ± 0.12	8.19 ± 0.19	10.35 ± 0.30	12.53 ± 0.32
4	2.28 ± 0.08	5.51 ± 0.49	7.91 ± 1.44	10.36 ± 2.47	13.15 ± 2.71	14.28 ± 2.15
4.5	3.02 ± 0.05	6.85 ± 0.12	10.27 ± 0.20	13.80 ± 0.32	16.72 ± 0.86	-
5	4.04 ± 1.05	8.15 ± 1.08	12.44 ± 1.24	17.37 ± 3.40	-	-
5.5	4.53 ± 0.06	10.24 ± 0.07	14.05 ± 2.69	20.44 ± 0.57	-	-
6	5.063 ± 0.06	12.31 ± 0.45	19.27 ± 0.59	21.95 ± 0.56	-	-
6.5	6.14 ± 0.15	14.32 ± 0.17	22.38 ± 0.43	-	-	-
7	7.01 ± 0.10	16.98 ± 0.32	24.68 ± 1.01	-	-	-
7.5	7.68 ± 0.50	19.06 ± 0.84	26.63 ± 2.75	-	-	-
8	8.72 ± 0.39	20.88 ± 0.51	28.12 ± 1.03	-	-	-

Table 2.3 – Mean frequency error for different set points and NFFTs.

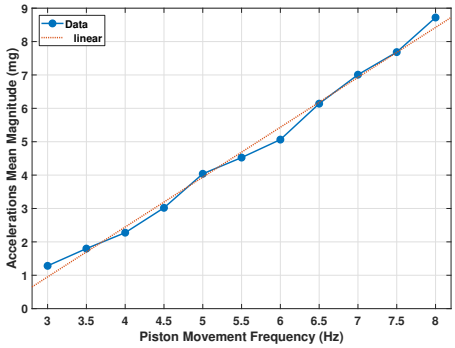
NFFT	**Frequency Set Point (Hz)**						**Mean Error (Hz)**
	3	**4**	**5**	**6**	**7**	**8**	
2048	0.003	0.001	0.005	0.006	0.007	0.008	**0.005**
1024	0.027	0.004	0.020	0.006	0.018	0.008	**0.012**
512	0.027	0.004	0.020	0.035	0.031	0.008	**0.022**

The inertial platform, combined with the analysis procedure, has shown to be an accurate system for the estimation of the dominant frequency of a vibration in

the range from 3 Hz to 8 Hz, having a mean estimation error lower than 0.01 Hz. The identification of the correct dominant frequency seems to be independent from the vibration amplitude as expected. Regarding the amplitudes, there is a linear relation between the movement amplitudes for a given frequency and their respective estimated FFT magnitudes, as reported in Figure 2.7a: the higher the movement amplitude, the higher the magnitude. In the same way, a linear relation between the movement frequencies for a given amplitude and the estimated FFT magnitudes is observable (Figure 2.7b): this is due to the fact that the selected set point of the cylindrical piston was a position, thus the higher the frequency, the higher the acceleration needed to perform the displacement in such a small time. Lastly, lowering the NFFT slightly reduces the accuracy of the system, with a mean frequency estimation error up to 0.022 Hz, as expected. However, since the desired resolution of the system is 0.1 Hz, the entire analysis can be executed with only 512 sample-long signals: this means that when the final classification algorithm will be developed and embedded, it will require lower computational costs and power consumptions.



(a) Mean magnitudes as a function of piston movement amplitudes for a given frequency of 3 Hz.



(b) Mean magnitudes as a function of piston movement frequencies for a given amplitude of 0.5 mm.

Figure 2.7 – Linear relations between FFT magnitudes and both peaks' features (amplitude and frequency).

2.4 Data Collection

The measuring system is composed of a single MuSe platform and a software for data collection. The device case is provided with an adjustable strap and was mounted on the back of the hand presenting a dominant tremor (see Figure 2.8). Typically, an intense pressure on the hand muscles can help the subject to temporary lessen the intensity of the tremor. Thus, in order not to alter the measurement, the strap was tightened just enough to keep the device in contact with the back of the hand.

Figure 2.8 – The device is mounted on the back of the hand by means of an elastic strap.

Measurements were performed on 24 subjects with an average age of 70 years (SD = 12), having a well-established diagnosis: 17 subjects affected by Parkinson's disease and 7 with essential tremor. The diagnoses were done by a neurologist on the basis of the standard diagnosis procedure (subject's medical history, physical and neurological examinations). Each subject was monitored with the described system while sitting at a table and performing the four standardized tasks described in section 2.1.3. Each task stresses a particular tremor: for instance, the rest position (task 1) should emphasize a PT, whereas postural tremor (task 2) and kinetic tremor (task 4) should be typically evoked in ET. The subject were instructed before the execution of each task. Data were collected at a sampling rate of 50 Hz for a period of 15 seconds per task.

2.4.1 Acquisition Software

In order to enable the communication between a smartphone (or tablet) and the wearable device, as well as to retrieve the data collected by the device itself, an Android application called "MuSe Connection" was developed within this thesis work. It provides a GUI (Graphical User Interface) allowing the user to establish the Bluetooth connection with the MuSe platform, to manage the device settings and to start the real-time transmission of the collected data.

ConnectionService Class

The first step to communicate with the MuSe device is to establish a Bluetooth connection through the Serial Port Profile (SPP). For this purpose, a specific Android class called *ConnectionService* was developed. This class is the core of the application since it represents the interface between the Android device and the wearable one. The class provides methods to:

- Open and close a Bluetooth connection with the MuSe device.

- Send serial commands to the connected device.
- Notify the parent class when events occur (i.e., message sent, new incoming data).

On creation, a new ConnectionService object requires (1) a *BluetoothDevice* object containing all the information about the device to connect to, such as its name and its MAC address, and (2) the reference to an *Handler* object, which will be the target of the messages sent from the ConnectionService class.

Once the object has been created, the connection can be established by calling the *connect()* method. When this happen, the ConnectionService object starts a new thread called *ConnectThread*, which creates an RFCOMM Bluetooth socket ready to start a secure outgoing serial connection to the specified remote device and tries to connect to it. If the operation succeeds, the ConnectionService object retrieves the opened socket, cancels the thread and starts a new one called *ConnectedThread*. This thread runs for the duration of the connection with the device, handling all incoming and outgoing transmissions. In the constructor method, it gets the input and output streams from the specified socket, then it keeps reading new bytes from the input stream by using a polling process. Once enough bytes have been completely received, it decodes the bytes sequence, stores the resulting data object in a list and notifies the external Handler object that a new packet is ready.

In order to manage the connection to the Bluetooth remote device, the ConnectionService class provides several methods based on the MuSe serial protocol. The following is a short list of methods and commands, the only ones actually used in the MuSe Connection application. For the entire command list and specification, see Appendix A.

startStreaming() Sending the command ?!CALIB050!?, this method allows the parent class to start the streaming of both raw data (from accelerometer, magnetometer and gyroscope) and processed data (quaternions) at an output data rate of 50 Hz.

stopTransmission() Sending the command ?!STOP_TX!?, this method stops the transmission of data.

getBatteryPercentage() Sending the command ?!GETBATTQ!?, this method retrieves the remaining charge of the device battery, in percentage.

All of the command-related methods are wrappers of the *write(byte[] buffer)* method of the ConnectedThread thread. This function receives the command to be

written as input parameter, sends the byte array over the output stream and notifies the Handler object that a message has been successfully written. In the MuSe Connection application for instance, once the main activity receives a notification that the stop message has been sent to the platform, it proceeds to create a local copy of the list containing the received data, in order to save it on file later.

In addition to calling the *write* function, each command-related method sets the dimension and the delimiter of the packet to be received. These parameters are different from command to command, according to the MuSe protocol: in this way, the polling mechanism implemented in the ConnectedThread thread is able to determine when a specific bytes sequence ends and to decode it in the proper way. For instance, the packet containing raw and processed data starts with the sequence "tt" and has a size of 54 bytes, whereas the battery level starts with the sequence "BQ" and has a size of 4 bytes.

The MuSe Connection application only relies on the streaming mode provided by the MuSe platform. Nevertheless, the ConnectionService implements also the methods for the request and the download of the files logged in the on-board MuSe memory, as well as the confirmation mechanism used by its protocol. The device collects data from the on-board sensors and group in 512 bytes-long packets. In order to avoid loss of packets during the transmission, the platform requires an acknowledge message (ACK) from the receiver every 20 transmitted packet, to confirm the success of the transfer. The platform waits until the ACK is received, then it resume the transmission sending the next 20 packets. The receiver can also send to the device a NACK (Not ACK), indicating that something went wrong during the transmission of the last bunch of packets: in this case, the MuSe platform send again the last 20 packets, and wait for an ACK to proceed. If neither an ACK nor a NACK is received by the platform within a defined time interval (timeout), it aborts the transmission.

This confirmation procedure is integrated in the ConnectedThread thread. Incoming packets are enqueued in a buffer of size 20: once it is full (or the transmission is terminated), the thread sends an ACK message to the remote device and notifies the parent class that packets have been successfully received. Every time a packet is enqueued but the buffer is not yet full (and the transmission is not completed), the thread waits for a certain period of time: if no packet is received within this time interval, a NACK message is sent to the MuSe, indicating the loss of some packet, and the buffer is emptied.

Graphical User Interface

On application startup, MuSe Connection shows the starting activity, shown in Figure 2.9. The application provides the user with two different acquisition modes, which can be started by clicking on the respective button.

Tremor Test This mode is used for data collection during the execution of the planned tasks. It also allows the user to manage the patient repository.

Single Acquisition This mode allows the user to perform a one-shot acquisition and save data in a specific folder. Data used for the characterization of the MuSe platform were collected by using this acquisition mode.

Figure 2.9 – MuSe Connection starting activity.

However, in order to connect the smartphone to a MuSe platform, the user first needs to turn on Bluetooth on the Android device. This operation can be performed directly from the application by pressing the Bluetooth icon in the upper right-hand corner of the screen: in this way, a new activity called *Bluetooth Settings* starts on top of the previous one. This activity (see Figure 2.10) allows the user to:

- Turn the Android device's Bluetooth on/off.
- Search for new Bluetooth devices.
- Select a new Bluetooth device from the list and pair it with the Android device.

The user can activate the Bluetooth on the Android device by clicking on the switch. Then, the MuSe platform must be paired to the device, if it has not already.

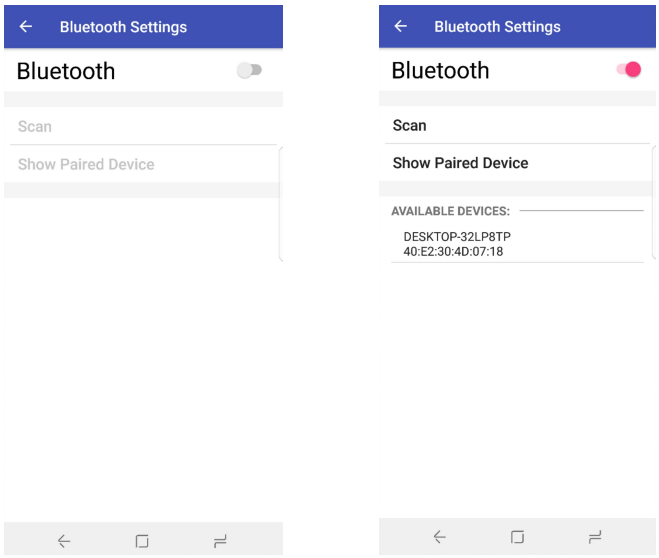


Figure 2.10 – Bluetooth Settings activity.

To do this, the user can search for new Bluetooth devices by pressing the "Scan" button: the list below will be filled in with the nearby Bluetooth devices, and the desired one can be paired by clicking on the relative item of the list. Other than this interface, the user can still manage the Bluetooth and the device pairing in the smartphone settings.

Once the MuSe platform has been paired to the Android device, the user can start to collect data by means of the desired acquisition mode. Since the *Single Acquisition* mode is just a wrapper of the ConnectionService class' methods *start()* and *stop()*, it won't be described in this section.

On the button click, the *Tremor Test* acquisition mode starts. The first step requires the user to insert a new patient or to select one of the patients previously saved: the Patient Manager activity, depicted on the left side of Figure 2.11, manages these operations. Patient's information text fields, such as name and surname, can be filled in, then a new patient can be created by pressing the "Add/Search" button: by doing this, the application creates the patient's directory and a new patient's descriptive text file inside. If a patient with the same personal info (name, surname, date of birth) already exists in the application folder, no new directory will be created; instead, its descriptive file will be updated with the new notes.

After the patient insertion/selection, its directory is used as save data location and the Tremor Test activity appears (see Figure 2.11). The user can connect the MuSe device by selecting it from a drop-down list; then each task execution can be collected by selecting the current test and pressing the "Start" button: the acquisition will automatically stop after 15 seconds, and the next test will be

selected. It is also possible to re-run a test, replacing the previously acquired data. Once all of the four tasks have been executed, data can be actually stored in a new subdirectory of the patient's folder by clicking "Save": a prompt message will then ask the user if another session should be started with the same patient, and on a negative answer they will be redirected to the starting activity.

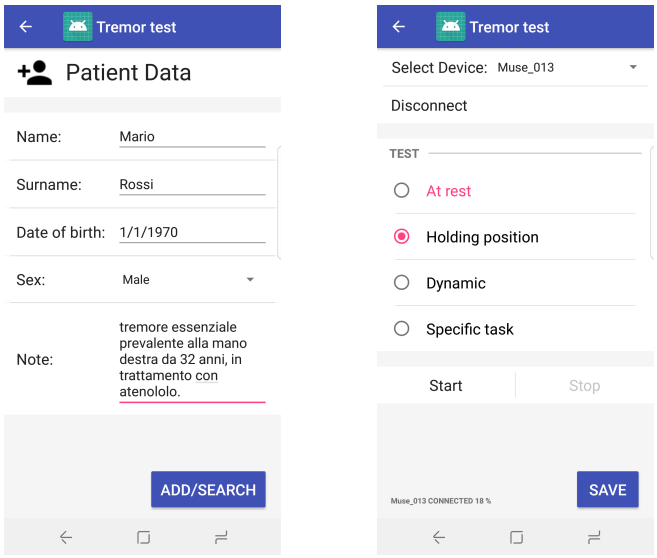


Figure 2.11 – On the left, the Patient Manager activity. On the right, the Tremor Test activity right before the execution of the second task.

2.5 Feature Extraction

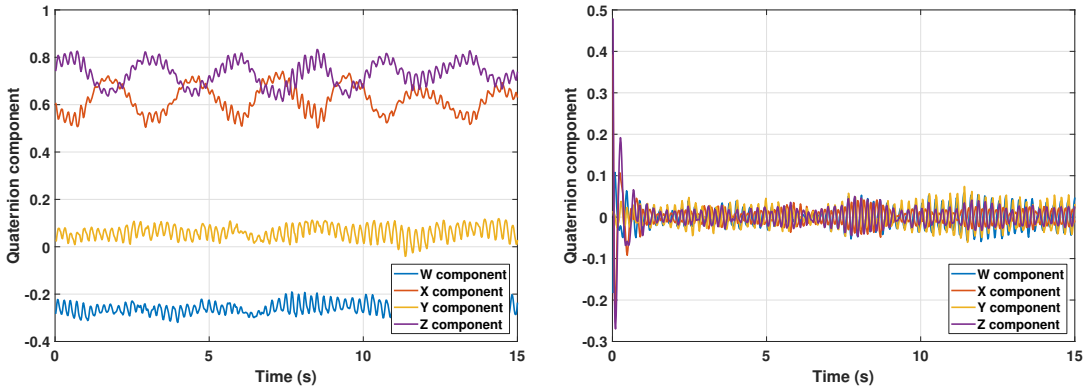
In order to develop an automatic tremor classification system, raw signals collected from a subject need to be compacted and transformed into a vector containing the relevant information from the input data: in this way, the classification task can be performed by using a reduced representation instead of the complete initial data. In machine learning, pattern recognition and in image processing, *feature extraction* starts from an initial set of measured data and builds derived values (features) intended to be informative and non-redundant, facilitating the subsequent learning and generalization steps, and in some cases leading to better human interpretations.

In this section, two feature extraction approaches are described, both based on the frequency representation of the collected signals. Such features will be used later to build the classification models (see section 2.6): since the quality and the usefulness of the extracted features are directly bound to the accuracies of the related classification models, the comparison between the two approaches is postponed and included in the discussion of the classifiers results.

When the MuSe device is used in streaming mode, the on-board sensor fusion algorithm uses the raw data collected by the sensors (i.e., accelerometer, magnetometer and gyroscope) to compute the orientation of the platform; then, both the raw and the processed data are transmitted in real time by the Bluetooth module. For each subject, 15 seconds of data were collected at a sample rate of 50 Hz and transmitted to the developed Android application: the received data were then stored in four different text files, one for each task. All data analysis was performed offline using MATLAB R2017b.

2.5.1 Data Preprocessing

Data collected by the device are related to the movement of the subject's hand, thus this includes both voluntary movements required in the specific task (such as drawing the spiral) and involuntary movements induced by the tremor. Since the former is performed at frequencies below the typical tremor frequency band, data were first filtered by means of a Chebyshev Type-II bandpass filter with cutoff frequencies of 2.5 Hz and 15 Hz. This digital filter was purposely designed to have a maximum passband ripple of 1 dB and a minimum stopband attenuation of 80 dB. In order to remove the transient response of the filter, each data acquisition was trimmed (2.5 seconds in both sides) obtaining a 10 seconds-long window, which corresponds to a signal of 500 samples. As an example, Figure 2.12 shows the effect of the Chebyshev filter on quaternions collected during the execution of task 3.



(a) Quaternions collected during the execution of task 3.

(b) The output of the Chebyshev Type-II band-pass filter.

Figure 2.12 – Effects of the Chebyshev filter on collected quaternions. The slower sine waves (hand movements) have been attenuated by the effect of the filter, whereas the fastest ones (tremor) remain unaltered.

2.5.2 Spectrum Fit Approach

The first feature extraction approach relies on the shape of the signal's Fourier representation. In fact, according to the results of some preliminary analysis, the typical shape of the frequency spectrum was similar to a "bell" curve: thus, data in the frequency domain were fitted with a Gaussian curve, and the parameters of the function were identified as signal descriptors.

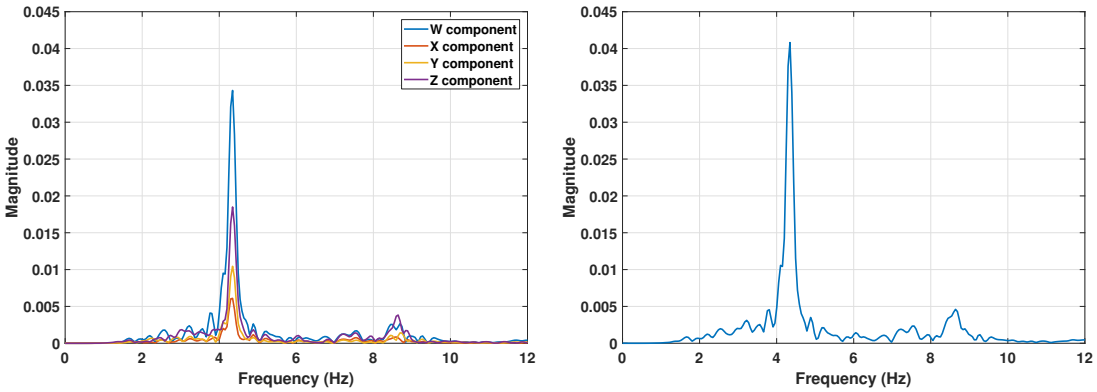
Quaternion-based Procedure

As described in section 2.1, tremors are defined as muscles activities causing oscillations of one or more body parts around a joint axis. Following the nature of tremors and their effects on the body parts, the first procedure was based on the hands rotations represented by the MuSe platform orientation, namely quaternions. These numbers are an extension of complex numbers, having a scalar part (W component) and a 3D vector part (X, Y and Z components): for this reason, they can easily give a simple way to encode the axis-angle representation of a rotation.

The analysis was performed in the frequency domain, thus a Fast Fourier Transform algorithm was applied to each task signal in order to obtain its equivalent frequency representation. The selected resolution was 1024 points: considering the device's ODR of 50 Hz, this leads to a frequency bin of 0.0488 Hz. Since each signal is composed of quaternions, a single task is represented by four data arrays in the frequency domain, one for each quaternion component (Figure 2.13a). Depending on how the tremor influences subject movements (that is, around which axis it is dominant), one quaternion component might encode more information than others. In order to combine all of these pieces of information, amplitudes of the four FFT outputs were added in quadrature, i.e., for each sample, the square root of the sum of the squares was determined, leading to one signal per task (Figure 2.13b). This procedure also has the advantage to make the system independent of the particular device's mounting position.

The typical shape of the Fourier representation of a tremor is a "bell" curve, having a maximum in correspondence of the dominant frequency and then symmetrically spanning over a certain range of frequencies around it. The position and the shape of the curve depend on the nature of the tremor — whether ET or parkinsonian tremor. Considering these two types of tremors, the fundamental component should lie in a range between 4 Hz and 8 Hz. For these reasons, each signal was fitted with a Gaussian curve in the form

$$f(x) = ae^{-(\frac{x-b}{c})^2} \tag{2.4}$$



(a) Fourier representation of the four quaternion components.

(b) Composed signal from the four FFT outputs.

Figure 2.13 – Fourier representation of collected quaternions in the 0–12 Hz band.

where a is the height of the curve's peak, b is the position of the center of the peak and c is related to the width of the "bell". The goal of the fit was to determine the value of these parameters, which define the frequency distribution of the analyzed tremor. Figure 2.14 shows the signal and the fitted Gaussian curve of a subject affected by PT performing task 1: in this case, the center of the peak correctly falls in the PT frequency range.

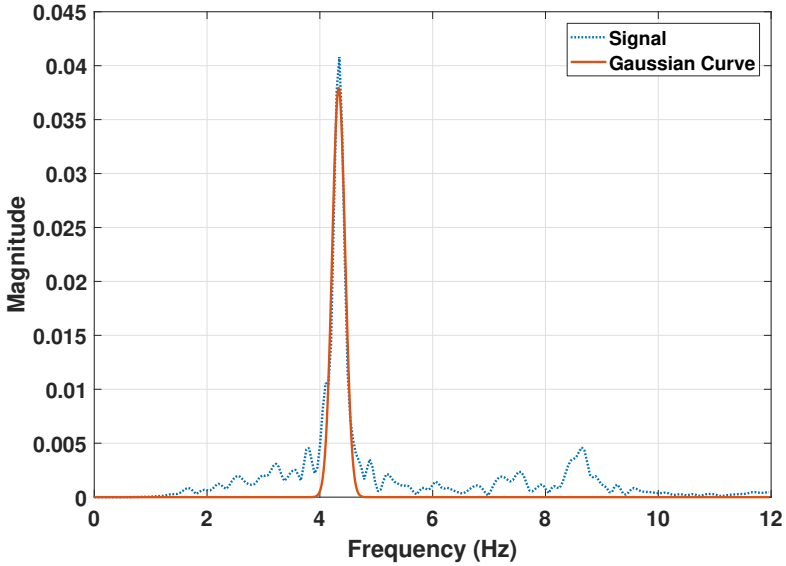


Figure 2.14 – A Gaussian curve fitting the Fourier representation of quaternions collected from a parkinsonian patient.

Different solutions were simultaneously adopted to improve the Gaussian fit:

- Parameters a, b and c were bound to be nonnegative.
- Samples corresponding to frequencies above 12 Hz were excluded from the

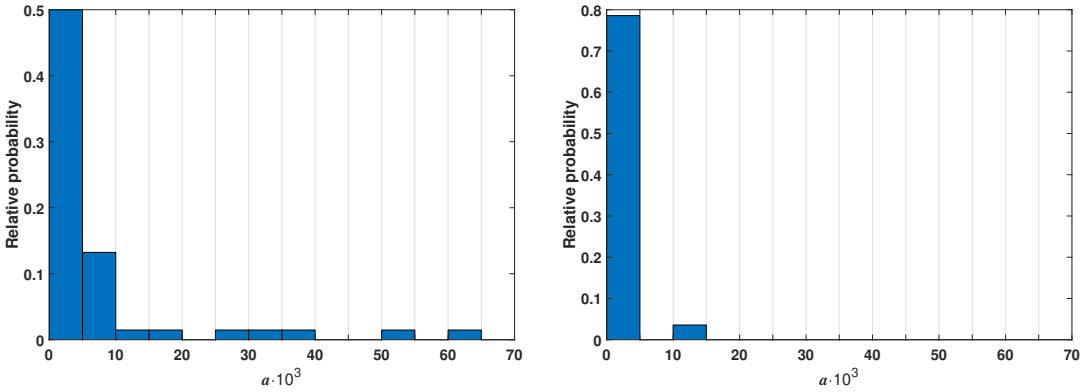
curve fit procedure, otherwise the algorithm would have adjusted the Gaussian curve to fit its right tail with those uninteresting data, and that would have led to a worse fit of the curve in the low frequency range.

- Different weights were assigned to signal samples, in order to favor the fitted curve to lie in the frequency range of interest (3–10 Hz band).

Nevertheless, some Fourier representations of the acquired tremors were not well-described by a Gaussian function. Even so, the fit algorithm provided for a Gaussian curve, despite having either very low peak heights or widths larger than the frequency range of interest itself, or both. For instance, an almost flat tremor spectrum could be fit with a very wide Gaussian curve, and the so-estimated central peak position would not be significant. For this reason, the estimated width of fitted Gaussian function was used to determine the quality of the fit process: 23 curves out of 96 were characterized by a high value of c, and were thus excluded from the analysis due to their low significance. The threshold value ($3\sqrt{2}$ Hz) was chosen as it represents a 3 Hz wide "bell" curve, a function wider enough to include both the tremor peak and its frequency spread regardless of the tremor type.

Figure 2.15a and Figure 2.15b depict the distribution of the tremor amplitudes of all tasks, for subjects affected by Parkinson's disease and subjects with essential tremor respectively. As can be noticed, while the ET subjects samples lay within a small interval of low amplitudes, the PT samples spread in a wider range, characterized by a high number of subjects showing a low-amplitude tremor and very few subjects affected by high-amplitude tremors. This reflects a non-uniform distribution of the available PD sample with respect to the entire PD population, and suggests the need of enlarging the sample. When doing this, it will be important to remember that the parkinsonian tremor amplitude is not necessarily related to the progression of the disease (i.e. the longer a patient has been affected by PD, the more severe their tremor) but to its specific subtype, as already described in section 1.3.2 (i.e. tremor-dominant or non-tremor dominant PD).

Since parameters do not come from a normal distribution, they could not be described by means of the notation $\bar{x} \pm s$, with $\bar{x}$ denoting the sample mean and s the sample standard deviation. A more generic approach, based on their quartiles, was adopted to describe the parameter distributions. For this purpose, box plots are a simple method for graphically depicting groups of numerical data. Box plots, introduced by John W. Tukey in 1969, are non-parametric: they display variation in samples of a statistical population without making any assumptions of the underlying statistical distribution. The box extends from the first or "lower"



(a) PT subjects sample. **(b)** ET subjects sample.

Figure 2.15 – Tremor amplitude distributions.

quartile (25th percentile) to the third or "upper" quartile (75th percentile), with a band inside the box representing the second quartile (the median). The length of the box is called interquartile range or IQR, being equal to the difference between 75th and 25th percentiles. Whiskers branch off the box, and their ends can represent several possible alternatives, the most common being Tukey version: the lowest datum still within 1.5 IQR of the lower quartile, and the highest datum still within 1.5 IQR of the upper quartile (see Figure 2.16).

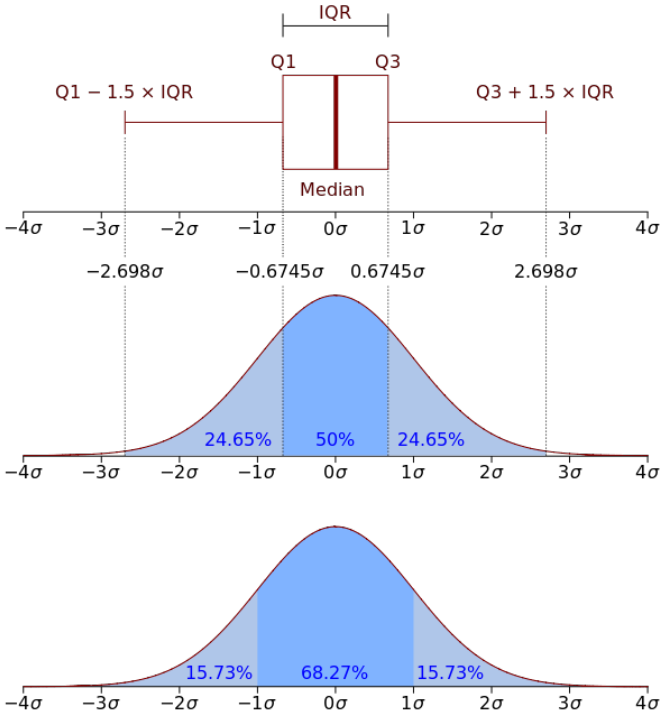


Figure 2.16 – Boxplot and a probability density function (pdf) of a Normal N(0,1σ^2) population.
Source: *Wikimedia Commons*, by Jhguch [CC BY 2.5]

Table 2.4 summarizes the distribution quartiles of Gaussian curve parameters, for subjects showing parkinsonian tremor and subjects showing ET. The fitted Gaussian curves are centered in the correct frequency range of the associated tremor: specifically, the inter-task median of frequency centers (b parameter) is 4.94 Hz for parkinsonian tremors (typical frequency range: 4–6 Hz) and 5.71 Hz for essential tremors (typical frequency range: 5–8 Hz). The heights of the curves' peaks (a parameter) reflect the sample distributions previously reported (see Figure 2.15, with PT laying in a higher amplitude range than the ET ones.

Table 2.4 – Quartiles of Gaussian curve parameters. For each distribution, the size of the sample (#) has been reported.

Parameter	PT			ET		
	$\tilde{x}$	$(P_{25}; P_{75})$	#	$\tilde{x}$	$(P_{25}; P_{75})$	#
$a \cdot 10^3$	2.02	(1.02; 5.60)	50	1.09	(0.23; 3.61)	23
b (Hz)	4.94	(4.42; 6.05)	50	5.71	(4.68; 6.36)	23
c (Hz)	0.81	(0.31; 2.41)	50	1.39	(0.77; 3.23)	23

For a detailed view, Table 2.5 summarizes the quartiles of the parameter distributions grouped by task: this view can be useful since it is expected to observe different tremor manifestation with respect to the specific task performed. As reported in section 2.1.2, parkinsonian tremor usually appears while the subject is resting, whereas essential tremor shows up while executing postural and kinetic tasks: by comparing PD and ET parameters for each task, it is possible to confirm these clinical statements. Task 1 results show that the fitted Gaussian curves are typically higher and more narrow in PD tremors with respect to the ET ones, indicating a greater manifestation of the former type of tremor at rest as expected. Moreover, a second observation is that no other PT task has a greater value of a nor a lesser value of c than those obtained in the first task; likewise, no other ET task has a lesser value of a nor a greater value of c than the first task ones. As described in section 2.1.2, in some cases PD patients may show a postural tremor with a slightly higher frequency (6–8 Hz) than resting tremor. Results from task 2 may be partially explained by taking into account this aspect of the disease. In fact, in this task parkinsonian tremor and essential tremor Gaussian curves have comparable heights and width, giving no significant distinction between them. On the contrary, results from tasks 3 and 4 — postural/kinetic and kinetic tasks respectively — highlight the presence of essential tremors among subjects, characterized by higher and more narrow Gaussian curves in essential tremors with respect to the PD ones. The high values of the PT median frequency center may be ascribed to the fact that the filtering action could have not completely removed the movement-related

frequencies, dependent on the ways the subjects performed this task, leading to an increase of the FFT output in the band of interest.

Table 2.5 – Quartiles of Gaussian curve parameters, grouped by task. For each distribution, the size of the sample (#) has been reported.

Task	Parameter	PT			ET		
		$\tilde{x}$	$(P_{25}; P_{75})$	#	$\tilde{x}$	$(P_{25}; P_{75})$	#
1	$a \cdot 10^3$	5.13	(2.81; 10.88)	17	0.16	(0.14; 0.19)	7
	b (Hz)	4.87	(4.40; 5.40)	17	4.60	(2.85; 4.82)	7
	c (Hz)	0.31	(0.15; 0.56)	17	3.20	(1.91; 3.68)	7
2	$a \cdot 10^3$	1.80	(0.48; 6.33)	13	1.79	(1.30; 4.68)	5
	b (Hz)	5.73	(4.80; 6.41)	13	6.21	(5.69; 6.27)	5
	c (Hz)	0.65	(0.30; 1.69)	13	0.53	(0.28; 1.43)	5
3	$a \cdot 10^3$	1.81	(1.10; 3.22)	9	3.71	(2.91; 4.05)	5
	b (Hz)	6.04	(4.86; 6.46)	9	6.46	(5.98; 6.70)	5
	c (Hz)	2.57	(1.39; 3.45)	9	1.39	(1.23; 2.49)	5
4	$a \cdot 10^3$	1.28	(0.81; 1.94)	11	1.89	(0.69; 3.64)	6
	b (Hz)	4.55	(4.17; 5.19)	11	5.37	(4.71; 5.83)	6
	c (Hz)	1.65	(1.05; 3.92)	11	0.91	(0.41; 3.87)	6

The Choice of Data Source

Tremors are rotation of a body segment around an axis, and as such they can be well represented by the quaternion mathematical structure. Nevertheless, this information do not come for free in a motion tracking system, since it is the result of the processing of raw data coming from different sensors, such as accelerometers and gyroscopes. The analysis procedure reported in the previous section, in fact, requires the estimation of the platform orientation by means of an embedded Kalman filter. It is well-known that this algorithm allows to obtain a simple and reliable position tracking system, such as the one reported in [17]: however, it also comes with some disadvantages such as increasing computational costs and, as a result, increasing power consumption. By using raw data in place of device's orientations, an embedded classification procedure could possibly avoid the use of such a complex algorithm, resulting in lower power consumption.

For this purpose, the very same procedure used to extract features from quaternions was applied on raw data (i.e., accelerations and angular rates) collected from the accelerometer and the gyroscope available on the MuSe platform. Following the results of section 2.3 in fact, the goal of this study was to assess whether the results previously obtained could be achieved in a less expensive way, by using raw data in place of quaternions.

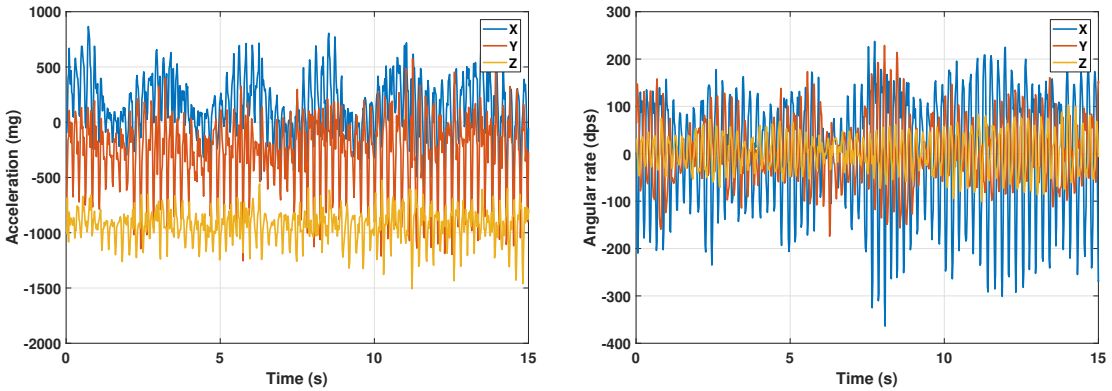


Figure 2.17 – Accelerations (left) and angular rates (right) collected during the execution of task 3.

Table 2.6 and Table 2.8 summarize the distribution quartiles of Gaussian curve parameters obtained by analyzing acceleration and angular rate signals respectively, whereas Table 2.7 and Table 2.9 report their detailed views. Here again, the fitted Gaussian curves are centered in the correct frequency range of the associated tremor: the inter-task median of frequency centers is 5.84 Hz for PT and 6.34 Hz for ET when computed from accelerometer data, and 5.68 Hz for PT and 6.23 Hz for ET when computed from gyroscope data. Moreover, although the median height of the curves' peaks is similar between PT and ET (regardless from the data source), the parkinsonian IQRs lay in a higher amplitude interval with respect to the ET IQRs for both the data sources, as previously obtained from quaternions analysis.

Table 2.6 – Quartiles of Gaussian curve parameters from accelerometer data.

Parameter	**PT**			**ET**		
	$\tilde{x}$	$(P_{25}; P_{75})$	#	$\tilde{x}$	$(P_{25}; P_{75})$	#
a (mg)	27.61	(14.62; 64.99)	50	24.67	(6.13; 44.75)	18
b (Hz)	5.84	(4.72; 6.63)	50	6.34	(5.66; 6.84)	18
c (Hz)	0.69	(0.23; 2.34)	50	0.95	(0.38; 1.48)	18

In order to assess the interchangeability, Pearson's linear correlation indexes r among pairwise data sources were computed, for each parameter and for each task, regardless the tremor nature. Moreover, right tailed significance tests were performed under the null hypothesis of correlation being equal to zero, against the alternative hypothesis of correlation being greater than zero. The significance level α was set to 0.05. Table 2.10 reports the correlation indexes of data sources for each parameter and task, and the related p-values computed from the significance tests. Thirty-four (34) out of the performed thirty-six (36) tests resulted in a p-value less than 0.05, suggesting that the null hypothesis of no correlation may be rejected.

Table 2.7 – Quartiles of Gaussian curve parameters from accelerometer data, grouped by task.

Task	Parameter	PT			ET		
		$\tilde{x}$	$(P_{25}; P_{75})$	#	$\tilde{x}$	$(P_{25}; P_{75})$	#
1	a (mg)	39.73	(20.75; 119.36)	17	4.62	(2.48; 5.66)	4
	b (Hz)	4.87	(4.40; 5.54)	17	5.99	(5.49; 6.65)	4
	c (Hz)	0.19	(0.14; 0.55)	17	0.88	(0.15; 2.24)	4
2	a (mg)	14.94	(5.18; 60.91)	14	20.41	(10.46; 44.27)	5
	b (Hz)	6.24	(5.73; 7.01)	14	6.26	(5.70; 6.45)	5
	c (Hz)	1.24	(0.33; 3.41)	14	0.48	(0.32; 1.37)	5
3	a (mg)	21.70	(15.29; 32.93)	12	40.30	(27.00; 51.69)	5
	b (Hz)	6.67	(6.22; 7.21)	12	6.84	(6.57; 6.95)	5
	c (Hz)	2.33	(2.01; 3.14)	12	1.31	(1.20; 1.85)	5
4	a (mg)	28.56	(24.03; 68.01)	7	35.23	(24.67; 52.53)	4
	b (Hz)	5.51	(4.91; 6.38)	7	5.64	(5.06; 6.42)	4
	c (Hz)	0.75	(0.43; 2.53)	7	0.69	(0.30; 1.05)	4

Table 2.8 – Quartiles of Gaussian curve parameters from gyroscope data.

Parameter	PT			ET		
	$\tilde{x}$	$(P_{25}; P_{75})$	#	$\tilde{x}$	$(P_{25}; P_{75})$	#
a (dps)	5.78	(2.06; 14.84)	54	6.43	(1.78; 14.02)	19
b (Hz)	5.68	(4.71; 6.36)	54	6.23	(5.66; 6.79)	19
c (Hz)	0.81	(0.25; 2.22)	54	0.85	(0.35; 1.28)	19

Table 2.9 – Quartiles of Gaussian curve parameters from gyroscope data, grouped by task.

Task	Parameter	PT			ET		
		$\tilde{x}$	$(P_{25}; P_{75})$	#	$\tilde{x}$	$(P_{25}; P_{75})$	#
1	a (dps)	10.41	(2.92; 25.62)	17	0.78	(0.43; 1.63)	4
	b (Hz)	4.88	(4.40; 5.48)	17	5.84	(5.26; 6.53)	4
	c (Hz)	0.25	(0.15; 0.69)	17	0.70	(0.14; 1.80)	4
2	a (dps)	7.42	(1.23; 19.68)	14	6.43	(4.82; 18.11)	5
	b (Hz)	6.16	(4.90; 6.61)	14	6.23	(5.70; 6.60)	5
	c (Hz)	0.56	(0.20; 1.61)	14	0.51	(0.24; 1.17)	5
3	a (dps)	5.96	(4.35; 8.94)	11	14.04	(11.43; 17.30)	5
	b (Hz)	6.31	(5.78; 7.07)	11	6.82	(6.43; 6.87)	5
	c (Hz)	2.06	(1.53; 2.96)	11	1.25	(1.10; 1.63)	5
4	a (dps)	2.46	(0.99; 5.22)	12	7.78	(2.75; 13.85)	5
	b (Hz)	5.86	(5.17; 6.33)	12	5.83	(5.29; 6.78)	5
	c (Hz)	2.08	(0.59; 3.49)	12	0.46	(0.37; 1.31)	5

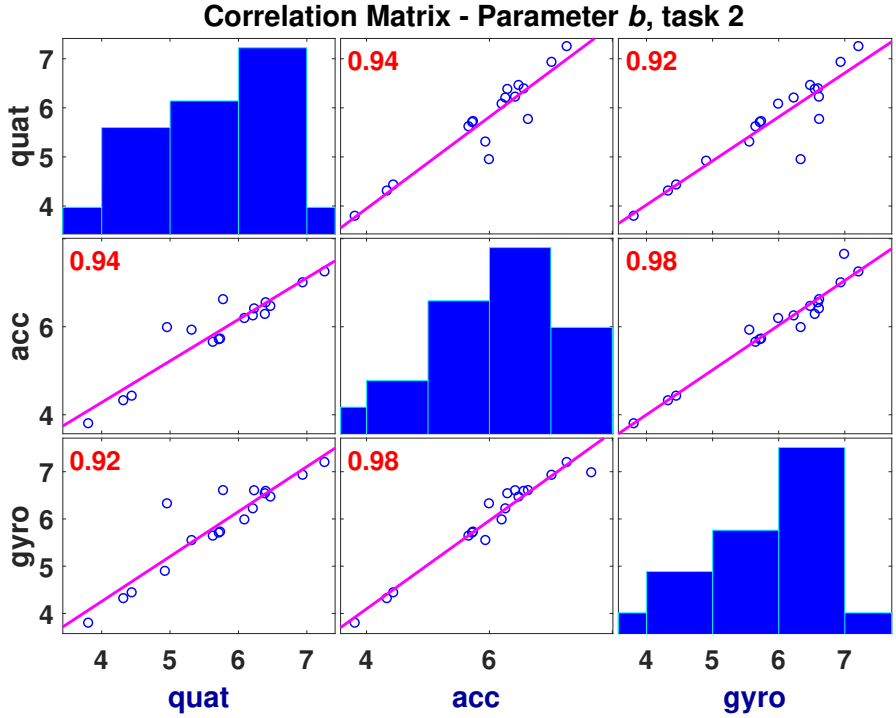


Figure 2.18 – Correlation matrix and indexes among data sources. Points in the charts represent the central frequency of the Gaussian curves during the execution of task 2, for both PT and ET subjects.

Table 2.10 – Correlation indexes among different data sources, for each parameter and task. Values in brackets represent the p-values obtained from the significance tests, under the null hypothesis of correlation being equal to zero ($\alpha = 0.05$).

Task	Parameter	r (p-value)		
		Quat VS Acc	Quat VS Gyro	Acc VS Gyro
1	a	0.94 (0.00)	0.98 (0.00)	0.95 (0.00)
	b	0.85 (0.00)	0.87 (0.00)	0.99 (0.00)
	c	0.75 (0.00)	0.65 (0.00)	0.93 (0.00)
2	a	0.99 (0.00)	0.99 (0.00)	0.99 (0.00)
	b	0.94 (0.00)	0.92 (0.00)	0.98 (0.00)
	c	0.91 (0.00)	0.98 (0.00)	0.93 (0.00)
3	a	0.96 (0.00)	0.99 (0.00)	0.98 (0.00)
	b	0.84 (0.00)	0.84 (0.00)	0.97 (0.00)
	c	0.89 (0.00)	0.82 (0.00)	0.96 (0.00)
4	a	0.52 (0.05)	0.75 (0.00)	0.77 (0.00)
	b	0.85 (0.00)	0.69 (0.00)	0.99 (0.00)
	c	0.61 (0.02)	0.80 (0.00)	0.42 (0.10)

Discussion

The spectrum fit approach was performed by using three different data sources: quaternions, accelerations and angular rates. Ten (10) seconds of each acquisition were filtered by means of a Chebyshev Type-II bandpass filter (2.5–15 Hz) and represented in the frequency domain by means of an FFT algorithm. Since orientations, accelerations and angular rates have more than one dimension, for each task the spectral densities of the components were added in quadrature. Finally, each frequency signal was fitted with a Gaussian curve, and the height of the peak, the central frequency and the width of the curves were extracted.

The correlation indexes and significance test results support the hypothesis of data sources being interchangeable, thus potentially reducing the computational cost of an integrated classification algorithm. The estimated center of the peak correctly fall in the proper frequency range for both parkinsonian and essential tremors. Moreover, data collected during the execution of the resting task show a predominance of parkinsonian tremors over ET, whereas data provided from postural and kinetic tasks highlighted the presence of essential tremor in the subject, as expected. Nevertheless, this feature extraction approach suffers from several issues due to the Gaussian curve fitting.

One of the main problems is that this technique relies on the hypothesis that a tremor spectrum can be always described by means of a "bell" shaped curve. However, this in not always possible. In some cases (see Figure 2.19) the tremor spectrum had a quasi-flat-top trend, thus the fitted Gaussian curve spanned over an interval wider than the actual range of interesting frequencies, and the curve peak was not correctly identified: in order not to distort the overall distribution of the parameters, curves with a width larger than a predefined threshold were considered to be erroneous and thus were excluded from the dataset. But once the classification system will be operative, there might be cases where subjects show tremor spectra similar to the excluded ones, and since no data of this kind were included in the training set, the risk of misclassifying that tremor increases.

A second problem is related to the quality of the fit. In some cases, a few small peaks around the spectrum's dominant peak induced the fit process to produce a wider and lower curve, thus underestimating the actual height of the dominant peak (see Figure 2.20). This could be avoided by reducing the fit interval to a neighborhood of the dominant (maximum) peak, the size of which could be defined a priori (for instance, 1 Hz per side). However, this trick would work only for sharp dominant peaks (it could potentially lead to the exclusion of part of a wider peak)

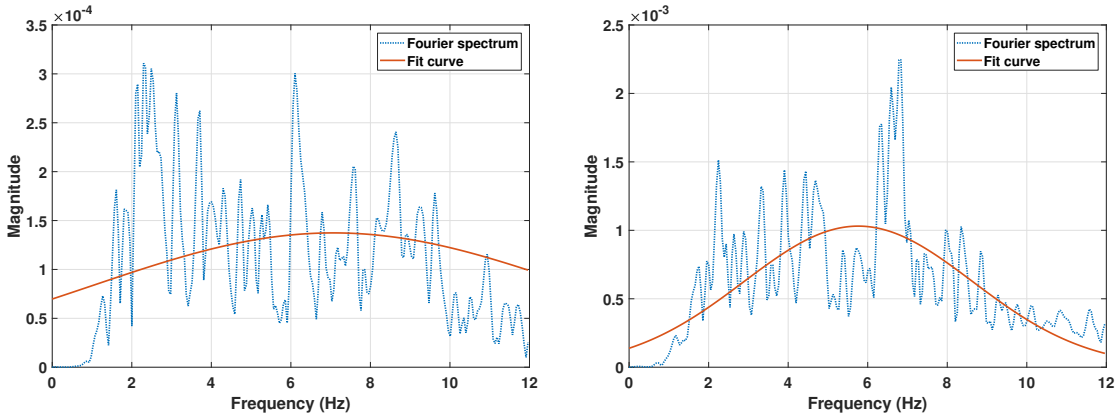


Figure 2.19 – Two cases of quasi-flat-top tremor spectra. In the right figure, the fitted curve has a peak center below 6 Hz, although a dominant peak is evident in the 6–7 Hz frequency range.

and would be dependent on the size of the fit interval (it may still include undesired peaks). For these main reasons, a more general approach, described in the next section, was used.

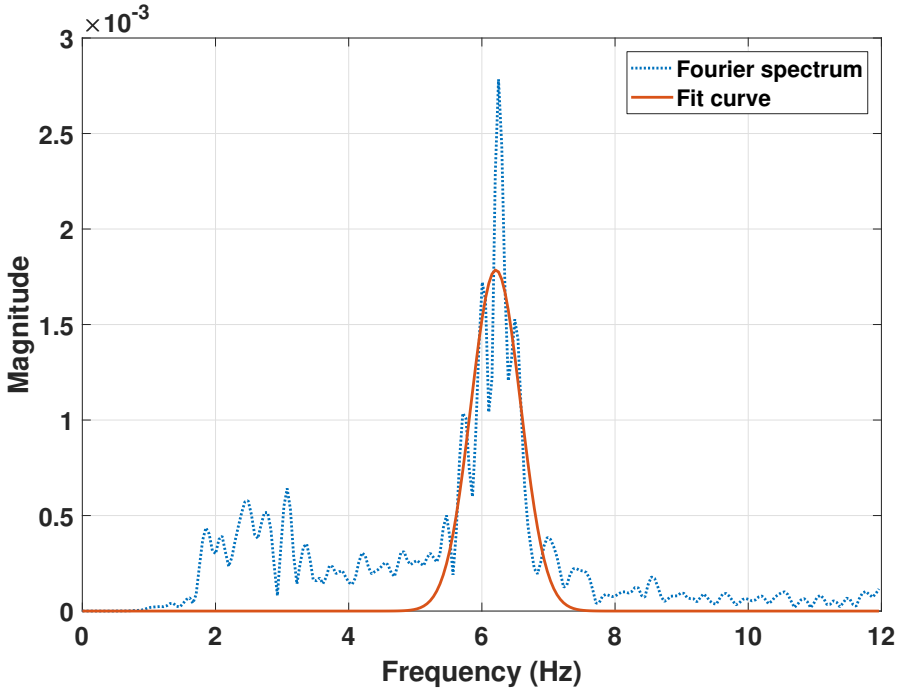


Figure 2.20 – ET spectrum of quaternion data, task 2. The fitted Gaussian curve underestimates the spectral peak due to the presence of low peaks in the neighborhood of the dominant one.

2.5.3 Power Features Approach

The main issue of the previous approach is that although a typical tremor frequency spectrum can be described by means of a "bell" shaped curve such as the Gaussian one, this not always stands. Moreover, even when the spectrum follows the typical pattern, the fitted curve may not accurately represent the FFT output. In order to be independent from the specific frequency spectrum pattern, a more general approach was adopted, based on the spectral features of the raw signals collected by the MuSe platform. Both accelerometer and gyroscope data were considered, in order to successively compare results obtained using both sensors with those obtained by using a single sensor only.

Spectral Estimation

The first step within this approach is to estimate the distribution (over frequency) of the power contained in the desired signal — in this case, accelerations and angular rates. Estimation of power spectra is useful in a variety of applications, including the detection of signals buried in wideband noise.

Let $\{x(t);\, t = 0, 1, 2, ..., N\}$ denote a discrete-time data sequence. $\{x(t)\}$ is obtained by sampling a continuous-time signal x_c with a sampling time interval T_s, that is, $x(t) = x_c(t \cdot T_s)$. The time index t is therefore expressed in units of sampling interval.

The *power spectral density* (PSD) of a stationary signal $x(t)$ describes where the average power is distributed as a function of frequency. It can be defined in one of the following equivalent ways.

- The PSD is the expected value of the discrete-time Fourier transform (DTFT) magnitude squared, over a large time interval:

$$\begin{aligned} S_{xx}(\omega) &= \lim_{T\to\infty} \mathbb{E}\left[\frac{1}{T}\left|X(\omega)\right|^2\right] \\ &= \lim_{T\to\infty} \mathbb{E}\left[\frac{1}{T}\left|\sum_{t=1}^{N} x(t)e^{-j\omega t}\right|^2\right]. \end{aligned} \tag{2.5}$$

- The PSD is the DTFT of the covariance sequence R_k:

$$S_{xx}(\omega) = \sum_{k=-\infty}^{\infty} R(k)e^{-j\omega k}. \tag{2.6}$$

The inverse function, which recovers $\{R(k)\}$ from given $S_{xx}(\omega)$, is

$$R(k) = \frac{1}{2\pi} \int_{-\pi}^{\pi} S_{xx}(\omega) e^{j\omega k} \, \mathrm{d}\omega \, . \tag{2.7}$$

From either of the above definitions, it can be noticed that $S_{xx}(\omega)$ is a periodic function, with period equal to 2π. Hence, $S_{xx}(\omega)$ is completely described in the interval

$$\omega \in [-\pi, \pi] \quad \text{(radians per sampling interval)}. \tag{2.8}$$

Alternatively, the PSD can be viewed as a function of the frequency $f = \omega/2\pi$ (cycles per sampling interval) which, according to eq. (2.8), can be considered to take values in the interval $f \in [-1/2, 1/2]$. For a defined sampling frequency F_s, it follows that the corresponding interval is

$$\left[-\frac{F_s}{2}, \frac{F_s}{2}\right] \quad \text{(Hz)}. \tag{2.9}$$

For a zero mean signal, $R(0) = \mathbb{E}\left[|x(t)|^2\right]$ measures the average power of $\{x(t)\}$. Also, considering the eq. (2.7) viewed as a function of frequency

$$R(0) = \frac{1}{F_s} \int_{-\frac{F_s}{2}}^{\frac{F_s}{2}} S_{xx}(f) \, \mathrm{d}f \, . \tag{2.10}$$

Thus, $S_{xx}(\omega)$ can indeed be named power spectral *density*, as it represents the distribution of the average signal power over frequencies. In other words, the quantity

$$\frac{1}{F_s} S_{xx}(f) \, \mathrm{d}f$$

represents the power content of the signal in an infinitesimal frequency band. The units of the PSD are power per unit of frequency, e.g. W/Hz. The average power of a signal in a particular frequency band $[f_1, f_2]$ can be found by integrating the PSD over that band:

$$\bar{P}_{[f_1,f_2]} = \frac{1}{F_s} \int_{f_1}^{f_2} S_{xx}(f) \, \mathrm{d}f = \frac{1}{F_s} \int_{-f_2}^{-f_1} S_{xx}(f) \, \mathrm{d}f \, . \tag{2.11}$$

For real-valued signals, the PSD is symmetric about the DC, and thus $S_{xx}(f)$ for $0 \leq f \leq F_s/2$ is sufficient to completely characterize the PSD. However, to obtain the average power over the entire Nyquist interval, it is necessary to introduce the

concept of the *one-sided* PSD: this is given by

$$S_{one-sided}(f) = \begin{cases} 0 & -F_s/2 \leq f < 0, \\ 2 \cdot S_{xx}(f) & 0 \leq f \leq F_s/2. \end{cases} \tag{2.12}$$

The average power of a signal over a non-negative frequency band can then be computed by integrating the one-sided PSD over the band of interest.

Sometimes the spectrum of the process may be defined over a discrete set of frequencies (or "bins"), instead of being a continuous function of frequency: in this case, it is generally referred to as *power spectrum*. The width of each bin represents the frequency resolution Δf of the spectrum and is defined as the bandwidth of the signal divided by the number of process samples available:

$$\Delta f = \frac{bandwidth}{\#\, samples}. \tag{2.13}$$

The power spectrum amplitude is the integration of the PSD over each frequency bin, representing that total band power. If the PSD units are W/Hz, its integration with respect to frequency yields units of Watts, as expected for the power.

If two identical signals are sampled at different sampling frequencies $f_1 > f_2$ within the same period T, the total number of samples will be $N_1 > N_2$. Since the number of available samples defines the spectrum frequency bin, the first signal will have a finer resolution with respect to the second one, i.e. $\Delta f_1 < \Delta f_2$, when the two signals are compared over the same bandwidth. Thus, since the amplitude of the power spectrum depends on the width of the frequency bin, the shape of the spectra will be different, with the first one being lower than the second even though the two signals are equal. Instead, the PSD will be the very same, being independent of the sampling frequency: for this reason, the PSD is typically used when comparing signals sampled at different frequencies.

Welch's Method

To perform the spectrum estimation of a signal, several different methods are available in literature. One way of estimating the PSD of a process is to simply find the discrete-time Fourier transform of the samples of the process (usually done on a grid with an FFT) and appropriately scale the magnitude squared of the result. This estimate is called the *periodogram*. Although it can be a useful tool for spectral estimation in situations where the signal-to-noise raio (SNR) is high, the periodogram is not a consistent estimator of the PSD, that is, the variance at a

given frequency does not tend to zero as the number of samples tends to infinity. For this reason, today the periodogram is just a component of more sophisticated methods, such as the Welch's method.

Named after P. D. Welch, this technique is used for estimating the power of a signal at different frequencies. The method consists of the following steps:

- The original signal is split up into K (usually overlapping) data segments of length M.
- Each overlapping segment is windowed in the time domain by means of a nonrectangular window. Because the segments usually overlap, data values at the beginning and end of the segment tapered by the window in one segment, occur away from the ends of adjacent segments. This guards against the loss of information caused by windowing. This step is what makes the Welch method a "modified" periodogram.
- For each segment, the periodogram is computed.
- The result of the periodograms are averaged for the K data segments. The ending result is an array of power measurements versus frequency bin.

Because the technique uses PSD estimates of different segments of the time series, the modified periodograms represent approximately uncorrelated estimates of the true PSD and averaging reduces the variability. However, the combined use of short data records (segments) and nonrectangular windows results in reduced resolution of the estimator: in summary, there is a tradeoff between variance reduction and resolution.

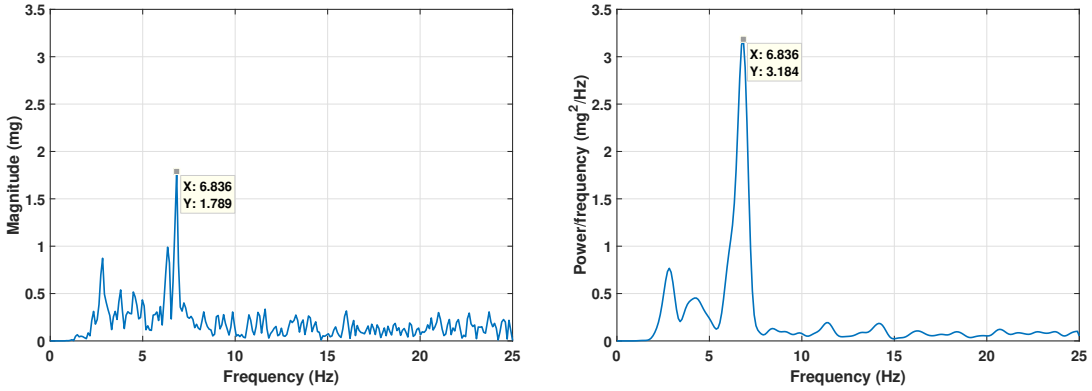


Figure 2.21 – Comparison between Fourier transform (left) and Welch's PSD (right) of an acceleration signal. The spectrum variability is mitigated by the effect of the Welch's averaging, but the resolution is coarser. The PSD amplitude equals approximately the FFT magnitude squared, in agreement with eq. (2.5).

Power Features

The aforementioned results obtained from the data sources correlation tests suggest that quaternions, accelerations and angular rates share very similar frequency components. As such, the estimation of power features was executed on raw signals provided by accelerometer and gyroscope only.

For each acquired signal and for each axis, the power spectral density was computed by means of the Welch's method. Data were split into overlapping segments of length 125 samples, with an overlap percentage of 50 %. Segments were windowed by using the Hann window function and zero padded in order to get 512 discrete Fourier transform (DFT) points. Finally, the modified periodograms were computed and averaged to get the power spectral density of the signal along an axis. Once the power spectral densities of the three axes were obtained, their amplitudes were added in quadrature in order to combine the spectra and obtain one PSD per signal.

The following power features were extracted from PSDs of each task, both for accelerometer and gyroscope data:

Peak Power (PP) The maximum power that was observed.

Peak Frequency (PF) The frequency where the maximum power occurred.

Peak Area (PA) The area under the observed power peak, within an interval of ± 0.3 Hz.

Median Power Frequency (MPF) The frequency dividing the PSD into two intervals, each with 50 % of the total power.

Total Power (TP) The power of the signal in the 2–10 Hz frequency interval (wide enough to include all tremors' frequency components).

By introducing some of the problem knowledge in the feature extraction process, a new feature called *Relative Power* (RP) was computed in addition to the above-mentioned one. In fact, since PT and ET are a resting tremor and a postural/kinetic tremor respectively (as described in section 2.1), the Total Power of a signal collected during the execution task 1 is expected to be higher for a PT patient with respect to an ET patient's signal's power; on the contrary, the Total Power of an ET subject's signal collected during the execution of task 2 is expected to be higher than a PT subject's one. For this reason, the Relative Power was defined as the ratio between

the Total Power in task 1 and the Total Power in task 2:

$$RP = \frac{TP_1}{TP_2} = \frac{\int_2^{10} S_1(f)\,\mathrm{d}f}{\int_2^{10} S_2(f)\,\mathrm{d}f} \tag{2.14}$$

Table 2.11 and Table 2.12 summarize the distribution quartiles of power features obtained by analyzing acceleration and angular rate signals, respectively. Generally speaking, results show that PT patients are characterized by higher-amplitude tremors with respect to ET patients, as resulted from the spectrum fit-based approach. Regarding the peak position, frequencies at which power maxima occurred (PF) as well as the estimated median power frequencies (MPF) correctly fall into the frequency range reported in literature for ET; instead, they are slightly greater than the expected ones for PD subjects. Lastly, results related to the relative power RP confirm the aforementioned hypothesis, being greater in PD patients than in ET. These overall considerations stand for both accelerometer- and gyroscope-based results.

Table 2.11 – Quartiles of power features from accelerometer data.

Feature	**Unit**	**PT**			**ET**		
		$\tilde{\boldsymbol{x}}$	$(\boldsymbol{P_{25}; P_{75}})$	#	$\tilde{\boldsymbol{x}}$	$(\boldsymbol{P_{25}; P_{75}})$	#
PP	mg^2/Hz	934.60	(166.87; 4412.44)	68	277.58	(52.82; 3197.82)	28
PF	Hz	6.20	(4.88; 6.98)	68	6.74	(5.71; 7.86)	28
PA	mg^2	586.98	(112.69; 2897.51)	68	205.19	(32.55; 2399.78)	28
MPF	Hz	6.67	(5.45; 7.96)	68	7.19	(6.22; 9.73)	28
TP	mg^2	1239.33	(252.95; 5147.43)	68	527.53	(106.11; 3818.33)	28
RP	-	1.51	(0.43; 24.44)	17	0.04	(0.02; 0.07)	7

Table 2.12 – Quartiles of power features from gyroscope data.

Feature	**Unit**	**PT**			**ET**		
		$\tilde{\boldsymbol{x}}$	$(\boldsymbol{P_{25}; P_{75}})$	#	$\tilde{\boldsymbol{x}}$	$(\boldsymbol{P_{25}; P_{75}})$	#
PP	dps^2/Hz	53.20	(6.12; 258.85)	68	18.44	(1.29; 324.96)	28
PF	Hz	5.66	(4.74; 6.49)	68	6.34	(5.47; 6.93)	28
PA	dps^2	36.83	(4.02; 166.13)	68	12.25	(0.95; 219.60)	28
MPF	Hz	6.02	(4.97; 6.75)	68	6.79	(5.82; 7.14)	28
TP	dps^2	88.61	(12.29; 280.81)	68	32.76	(3.92; 370.64)	28
RP	-	0.58	(0.23; 98.16)	17	0.01	(0.01; 0.02)	7

2.6 Building the Classifiers

The goal of the study was to create a model able to distinguish between parkinsonian and essential tremors. Section 2.5 described how to extract useful information from

raw signals by using two different approaches (i.e., spectrum fit approach and power features approach). This section illustrates how machine learning techniques were applied to the obtained feature sets, in order to build and train different models.

According to the definition, distinguishing between parkinsonian tremors and essential tremors can be addressed to as a classification problem: the tremor diagnosis is the known response, and belongs to one of a set of two discrete classes — PT and ET. For an overview of machine learning classification problems, refer to Appendix B.

2.6.1 ET Patients VS PT Patients Classifier

By using procedures described in section 2.5, several datasets containing the extracted features were built. In particular, they were obtained by applying one of the two feature extraction approaches (either spectrum fit one or power features one) to data from one of the available data sources (either orientations, accelerometer or gyroscope signals). The original sample contained a total of 24 patients with a well-known tremor diagnosis, among which 17 are affected by parkinsonian tremor and 7 by essential tremor: thus, each dataset was composed of 24 rows, or *observations*, and a variable number of features depending on the specific approach.

Considerations reported below, as well as the model building procedure, hold for all the datasets: for simplicity, the following discussion will therefore refer to a single data set. Small variations were applied to the procedure for the creation of models based on the spectrum fit approach: such variations will be explained in detail in the relevant paragraph.

Due to the small size (24 observations), the starting dataset could not be partitioned into a training data set and a test data, as the standard procedure suggests. In order to cope with this situation, a k-fold partitioning technique was used, with the k parameter set to 5. By randomly dividing the sample into k different folds, however, there is a chance that one or more folds do not contain observations from a specific class, thus conditioning the model training and the overall misclassification error. To avoid this, a *stratified* version of the k-fold partitioning technique was adopted: in this version, observations are grouped in such a way that the folds preserve the percentage of samples for each class, in order for each of them to be a good representative of the whole.

The original dataset contained 17 observation of the class “Parkinsonian” and only 7 observations labeled as “Essential”. This is an example of the so-called *unbalanced* datasets for a two-class problem, where the size of the classes are very unequal to each other. In unbalanced datasets, the former typically prevails over the

latter when training a new model: this will be biased towards a good classification of observations belonging to the majority class because of their high numerosity, to the detriment of those belonging to the minority one. There are a few ways to deal with unbalanced datasets:

- Undersampling involves removal of some random observation belonging to the majority class, to result in a balanced distribution of all classes. However, this approach is not ideal when the size of the original dataset is small, since it further contributes to reduce the total number of available observations.
- On the opposite, the oversampling approach involves the creation of new minority class' observations either by duplicating existing samples or by interpolating features of two observations of the minority class. Still, this method is not recommended since the artificial data may affect the model.
- A third way to cope with unbalanced datasets is to keep the dataset unchanged and, instead, to apply higher misclassification costs to the minority class in the model training phase. In this way, even if the model normally tends to better predict data belonging to the majority class, it will now avoid misclassifying observation of the minority class due to the higher cost.

As far as this study is concerned, neither the first nor the second solutions were used: instead, a modified version of the third solution was adopted. In fact, (1) since the correct classification of both parkinsonian and essential tremors should have the very same importance (consequences of a wrong prediction may be dangerous in both situations), and (2) since the class imbalance in the original dataset was only due to the fact that the used sample was not large enough (the size of both classes should be similar), misclassification costs were not modified during the model training. Instead, a specific metric was used to evaluate the model and to select the best one during the optimization phase: the *balanced accuracy* (BACC). For a two class problem, this metric is defined as

$$BACC = \frac{1}{2} \cdot \frac{TP}{TP + FN} + \frac{1}{2} \cdot \frac{TN}{TN + FP}, \tag{2.15}$$

where TP, TN, FP and FN represent the number of true positives, true negatives, false positives and false negatives respectively (see Appendix B for more details). The balanced accuracy weights in the same way both the majority and the minority classes, so that the final value is independent of the size of each of them. Since the ratio between essential and parkinsonian tremors is approximately 1:2.5, misclassify an observation labeled as "Essential" equals to misclassify 2.5 "Parkinsonian" observations.

Classification models were trained by using five different learning algorithms for each dataset. Other than selecting the model families, when building a high-quality, predictive classification model, it is important to select the right features and tune hyperparameters, i.e., model parameters that are not estimated.

The hyperparameters optimization was performed by using the Bayesian optimization strategy and cross-validation. Given an initial set of hyperparameters' value combinations (taken at random), this strategy iteratively computes the value of an objective function by cross-validating the model with the current hyperparameters, then it searches for a new promising combination of values which should minimize the objective function based on a built probabilistic model (see Figure 2.22). The procedure stops when certain conditions are achieved: in this case, a maximum number of 50 iterations was set to be the stop condition. This strategy was preferred to the grid and random searches because its flexibility typically grants better results (although being more expensive). The following list reports the tuned hyperparameters for each learning algorithm.

k-Nearest Neighbors	Number of neighbors; distance metric; distance weight; Minkowski distance exponent; standardization.
Support Vector Machine	Box constraint; kernel scale; kernel function; polynomial kernel function order; standardization.
Naïve Bayes	Data distributions; kernel smoothing window width; kernel smoother type.
Discriminant Analysis	Discriminant type (linear, quadratic); linear coefficient threshold (delta); amount of regularization (gamma).
Decision Tree	Minimum leaf size; maximum number of splits; split criterion.

On the other hand, the feature selection was performed by using a sequential forward strategy and cross-validation. This strategy belongs to the wrapper methods family and uses a predictive model to score feature subsets. Starting from an empty set, the algorithm creates candidate feature subsets by sequentially adding each of the features not yet selected. For each candidate feature subset, the model is cross-validated, where the mean criterion error is computed by means of a specific loss function. After computing the mean criterion values for each candidate feature subset, the algorithm chooses the candidate feature subset that minimizes the mean criterion value. This process continues until adding more features does not decrease the criterion. In particular, the chosen loss function was the complementary of

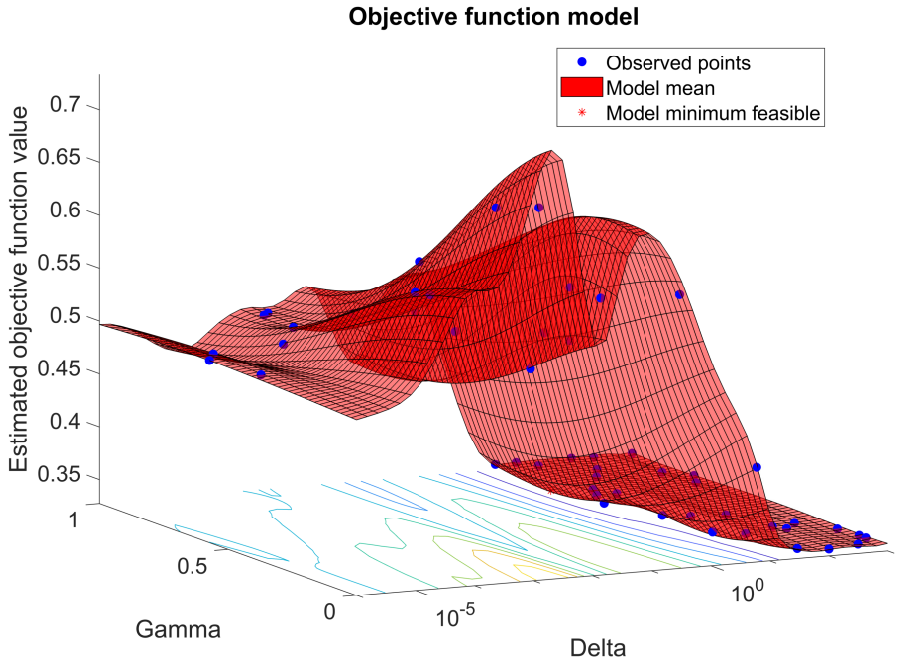


Figure 2.22 – Optimization of an SVM model's hyperparameters, delta and gamma. Each hyperparameter combination produces an objective function value, which has to be minimized.

the balanced accuracy. Since the wrapper method requires a predictive model, the following procedure was adopted for each learning algorithm:

1. A first model was trained with the entire set of features. Few hyperparameters were optimized by means of Bayesian optimization algorithm and cross-validation.

2. The sequential forward feature selection was computed by using the trained, loosely optimized model as the predictive model. The algorithm used cross-validation and the balanced accuracy complementary as loss function.

3. The final model was trained with the selected feature subset. The full hyperparameters optimization was performed with the same conditions as before (Bayesian optimization, cross-validation).

This procedure was adopted for each learning algorithm except for the decision tree. This model family, in fact, already integrates a feature selection approach, used to decide the branching condition at each split (before finding the weight). Thus, for this learner, only the third point was executed, with the entire feature set.

Spectrum Fit-based Classifiers

Datasets used in this paragraph were generated from the feature extraction process described in section 2.5.2. In particular, following the results obtained with regard to the data sources interchangeability, only the two datasets obtained from the analysis of acceleration signals and angular rate signals were considered. Each of those datasets contained 12 features, that is, the parameters a, b and c of the fitted Gaussian curve for each task. In addition, a new dataset with a total of 24 features was generated by merging the acceleration-based and the angular rates-based ones.

According to the feature extraction process, some values in the datasets were set to be *NaN* (Not a Number) because of the bad goodness of the fit (high values of parameter c). As long as an observation contains at least one valid feature value, it can still be used to train a classifier: if not, there would be no available feature to classify the observation. Since two out of the 24 subjects were characterized by having no valid features after the extraction process, those observations were excluded from the dataset used to build the models.

A total of 22 observations were grouped by using stratified 5-fold partitioning: this partition was used for the step 1 of the building procedure, that is, the training of the model with the whole feature set. However, a new partitioning was needed for the cross-validation within the second step (feature selection), since the iteratively selected features could have contained some NaN values. Thus, for each iteration, observations presenting all invalid features were temporarily excluded from the dataset and a new stratified 5-fold partitioning were computed, valid for that iteration only: unfortunately in this way, in some iterations some folds no longer included observations from both of the two classes due to the reduced size of the dataset, and this produced a biased estimation of their relative loss error.

Table 2.13, Table 2.14 and Table 2.15 report the optimal classifiers trained with the acceleration dataset, the angular rate dataset and the mixed one respectively. The features subscripts indicate the task which they refer to. Moreover, in the mixed dataset the abbreviations “Acc” and “Gyro” were used to distinguish features extracted from the accelerometer and gyroscope signals respectively. Each table contains the optimal feature subset and hyperparameters, as well as the out-of-sample, cross-validation accuracy of the final model.

Power Features-based Classifiers

Similar to what reported in the previous one, datasets used in this paragraph were generated from the feature extraction process described in section 2.5.3. Both the acceleration and the angular rates data sets contained a total of 21 features: 5 per

Table 2.13 – Classification models trained on Gaussian curve parameters from accelerations. For each model, the selected feature subset, the optimized hyperparameters and the out-of-sample accuracy are reported.

Learner	Feature set	Hyperparameters	Accuracy (%)
k-Nearest Neighbors	a_3	Neighbors Number: 9 Distance Metric: cosine Distance Weight: inverse Standardize: yes	82.35
Support Vector Machine	a_4	Box Constraint: 3.4968 Kernel Function: linear Standardize: yes	63.64
Naïve Bayes	a_1 b_1 c_4	Distribution: normal	90.91
Discriminant Analysis	a_1	Type: quadratic Gamma: 0	85.71
Decision Tree	b_1	Min Leaf Size: 10 Max Number of Splits: 3 Split Criterion: deviance	77.27

Table 2.14 – Classification models trained on Gaussian curve parameters from angular rates. For each model, the selected feature subset, the optimized hyperparameters and the out-of-sample accuracy are reported.

Learner	Feature set	Hyperparameters	Accuracy (%)
k-Nearest Neighbors	a_3	Neighbors Number: 8 Distance Metric: city block Distance Weight: inverse Standardize: no	87.50
Support Vector Machine	a_1	Box Constraint: 2.2712 Kernel Function: linear Standardize: yes	80.95
Naïve Bayes	c_1 b_2 a_3	Distribution: kernel Kernel Type: Epanechnikov Width: 0.030354	86.36
Discriminant Analysis	a_1	Type: linear Delta: 0.078064 Gamma: 0.49373	80.95
Decision Tree	a_3	Min Leaf Size: 3 Max Number of Splits: 8 Split Criterion: gdi	77.27

gdi: Gini's diversity index.

Table 2.15 – Classification models trained on Gaussian curve parameters from both accelerations and angular rates. For each model, the selected feature subset, the optimized hyperparameters and the out-of-sample accuracy are reported.

Learner	Feature set	Hyperparameters	Accuracy (%)
k-Nearest Neighbors	Acc a_3 Acc b_2 Gyro a_2 Gyro a_3	Neighbors Number: 2 Distance Metric: Spearman Distance Weight: inverse Standardize: no	84.62
Support Vector Machine	Gyro b_4	Box Constraint: 636.48 Kernel Function: linear Standardize: no	70.59
Naïve Bayes	Acc a_1 Acc b_1 Acc c_4	Distribution: normal	90.91
Discriminant Analysis	Acc a_1	Type: quadratic Gamma: 0	85.71
Decision Tree	Acc b_1	Min Leaf Size: 10 Max Number of Splits: 6 Split Criterion: deviance	77.27

each task's PSD, plus the Relative Power. In addition, a new dataset with a total of 42 features was generated by merging the acceleration-based and the angular rates-based ones.

In this case, since the extracted features were all valid for all of the 24 observations, only one stratified 5-fold partitioning was performed, and the entire procedure (loosely optimized-model generation, feature selection, final tuning) made use of the same partitioning results whenever cross-validation was required.

Table 2.16, Table 2.17 and Table 2.18 report the optimal classifiers trained with the acceleration dataset, the angular rate dataset and the mixed one respectively. The features subscripts indicate the task which they refer to. Moreover, in the mixed dataset the abbreviations "Acc" and "Gyro" were used to distinguish features extracted from the accelerometer and gyroscope signals respectively. Each table contains the optimal feature subset and hyperparameters, as well as the out-of-sample, cross-validation accuracy of the final model..

Discussion

The building procedure produced several classifiers able to differentiate among parkinsonian tremor and essential tremor on the basis of features extracted from different signals and approaches. For each learning algorithm, feature selection and fine hyperparameters tuning were performed to obtain a good predictive model.

Table 2.16 – Classification models trained on accelerations power features. For each model, the selected feature subset, the optimized hyperparameters and the out-of-sample accuracy are reported.

Learner	Feature set	Hyperparameters	Accuracy (%)
k-Nearest Neighbors	PP_1 PP_2 PP_4	Neighbors Number: 2 Distance Metric: correlation Distance Weight: equal Standardize: no	79.17
Support Vector Machine	PP_1	Box Constraint: 833.91 Kernel Function: Gaussian Kernel Scale: 0.020372 Standardize: yes	95.83
Naïve Bayes	RP	Distribution: normal	95.83
Discriminant Analysis	PF_1 MPF_4	Type: quadratic Gamma: 0	91.67
Decision Tree	PP_1	Min Leaf Size: 2 Max Number of Splits: 6 Split Criterion: gdi	87.50

gdi: Gini's diversity index.

Table 2.17 – Classification models trained on angular rates power features. For each model, the selected feature subset, the optimized hyperparameters and the out-of-sample accuracy are reported.

Learner	Feature set	Hyperparameters	Accuracy (%)
k-Nearest Neighbors	PP_1 PF_1 TP_1	Neighbors Number: 6 Distance Metric: cosine Distance Weight: inverse Standardize: no	87.50
Support Vector Machine	PP_1 TP_1	Box Constraint: 237.02 Kernel Function: Gaussian Kernel Scale: 0.26942 Standardize: no	91.67
Naïve Bayes	PP_1 RP	Distribution: normal	100.00
Discriminant Analysis	MPF_1 PP_4	Type: linear Delta: $2.325 \cdot 10^{-5}$ Gamma: 0.42864	83.33
Decision Tree	RP	Min Leaf Size: 4 Max Number of Splits: 17 Split Criterion: gdi	91.67

gdi: Gini's diversity index.

Table 2.18 – Classification models trained on both accelerations and angular rates power features. For each model, the selected feature subset, the optimized hyperparameters and the out-of-sample accuracy are reported.

Learner	Feature set	Hyperparameters	Accuracy (%)
k-Nearest Neighbors	Acc PP_1 Gyro TP_2	Neighbors Number: 11 Distance Metric: Chebyshev Distance Weight: inverse Standardize: no	95.83
Support Vector Machine	Acc PP_1	Box Constraint: 972.96 Kernel Function: Gaussian Kernel Scale: 901.76 Standardize: no	95.83
Naïve Bayes	Acc RP	Distribution: normal	95.83
Discriminant Analysis	Acc PF_1 Acc MPF_4	Type: quadratic Gamma: 0	91.67
Decision Tree	Acc PP_1	Min Leaf Size: 2 Max Number of Splits: 5 Split Criterion: deviance	87.50

Finally, the cross-validation resulted in the accuracies reported in the previous tables.

The use of multiple sensors is usually preferred in an acquisition system, as this increases the available data: this, in turn, can help to improve the quantity and the quality of extracted information. Nevertheless, in miniaturized acquisition systems such as wearable devices, the area occupancy and the power consumption of a sensor are two important factors that may overcome the choice of having more data available. Thus, many wearable devices do not provide both the sensors used in this work (namely, accelerometer and gyroscope), or alternatively the full product comes with a high price. In order to investigate the portability of the developed algorithms and to determine its reliability when some sensors are not available, three different feature sets were built from each extraction approach: two datasets were based on raw data collected by means of accelerometer and gyroscope respectively, while the third one contained information from both data sources.

Generally speaking, classifiers based on power-related features performed better than those based on parameters of the fitting Gaussian curves, having in average greater accuracies. These results, along with previously discussed problems associated with this latter approach, lead to prefer the classification based on signal power features with respect to the one based on the spectrum fit.

Regarding datasets based on power features, the feature selection procedure revealed that the most used one is the Peak Power of task 1 (recurring in 9 out of

15 classifiers), followed by Relative Power (4 uses) and Peak Frequency of task 1 (3 uses). These results highlight the importance of the resting task (and partially, the postural task) in the differentiation among parkinsonian and essential tremors, as expected. The goal of task 1, in fact, is to stress the former tremor: in a population of only patients and for a two class problem, data collected within this task are sufficient to correctly classify almost all subjects. However, the rest of the features, as well as the tasks, might become relevant if the classification problem either extends to other tremor forms, or includes healthy subjects to be classified as such.

As can be expected, the use of multiple sensors led to better performances, with accuracies of models trained with the mixed dataset slightly greater than those trained with data from a single sensor in average. However, solutions only based on data either from accelerometer or gyroscope achieved out-of-sample accuracies around 90 %, which is still a good result. Although best results were achieved by Support Vector Machine and Naïve Bayes classifiers, the choice of the final model should take into account not only the estimated prediction accuracy, but also intrinsic features of the specific learning algorithm. Some of these are:

- Training time;
- Interpretability;
- Prediction time;
- Flexibility.

For instance, Support Vector Machine classifiers have typically a high accuracy, but Naïve Bayes classifiers require less training time. On the other hand, both may lack of interpretability, an attribute for which decision trees and k-Nearest Neighbors are the best solutions.

The author of this thesis reserves the right not to choose the best classifier at the time of writing, since this final decision will depend on several other factors. Firstly, the sample needs to be enlarged with new patients in order to get a higher number of observations, possibly equal among parkinsonian and essential tremors. Moreover, the model selection will depend on the computational power of the system that will host it, being either the wearable device itself or the Android system running the classification app.

2.6.2 Control Subjects VS Patients Classifier

In order to enlarge the capabilities of the developed classification system, an additional preliminary study was conducted: the goal was to verify whether or not

the feature extraction procedures would provide sufficient information to differentiate between healthy subjects and tremor-affected patients.

At this purpose, 14 healthy subjects with an average age of 45 years (SD = 20) underwent the same patient test procedure: they were monitored by means of a MuSe platform mounted on the back of their non-dominant hand, while executing the four standardized tasks aforementioned. After collecting orientation and inertial measurements, data were analyzed as described in previous sections. In order to have a first insight about how control cases could have been differentiated from patients, the frequency spectra of signals related to both categories were compared, as depicted in Figure 2.23. As can be noticed, the absence of tremor in control cases clearly results in lower signals power and, as a consequence, in lower frequency spectra than the patients ones (those from task 1 or 2, dependently on the tremor type).

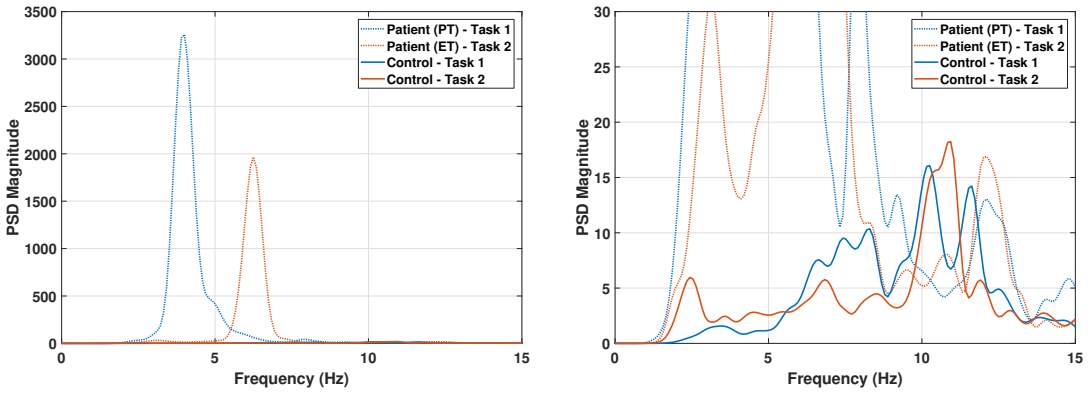


Figure 2.23 – Comparison of power spectral densities from patients' and control cases' acceleration signals. A zoom of the chart (on the right) highlights how spectra related to task 1 and task 2 are significantly higher in patients than in the healthy subject.

A first group of control cases datasets were built by using the spectrum fit approach described in section 2.5.2: specifically, following the results of data sources interchangeability evaluation, only inertial data (accelerations and angular rates) were considered. The feature extraction process resulted in 12 out of 14 observations characterized by having NaN values only (thus, with no valid feature available), independently of the data source. This result was expected, confirming that hands movements of healthy subjects do not show remarkable frequency components in the tremors band of interest: as such, no Gaussian curve could be fit, and no classifier could be built due to the lack of valid features. However, although the spectrum fit approach alone would seem sufficient to differentiate control cases and patients due to the observed behavior (12 out of 14 potentially classified correctly),

it is not advisable to base the classification response on a non-identification as the NaN values. In future, in fact, such result might be due either to non-expected scenarios or to some fit problems, as was the case of data from the two patients that were excluded from the dataset in section 2.6.1.

On the other hand, the second group of control cases datasets were built according to the analysis procedure described in section 2.5.3. Several features were extracted from the power spectral densities of accelerometer and gyroscope data, and then used to create three different datasets (one per sensor, and one including both). Then, features from the patients' datasets were added to the control cases ones, and the "PT" and "ET" categories were merged to form one label only, named "Patient". As a result, three datasets consisting of 38 observations (24 patients and 14 healthy subjects) were built and used to train several classifiers according to the methods described in section 2.6.1: Table 2.19 reports the out-of-sample accuracies of the resulting models, for each learner and dataset used (for the sake of brevity, selected features and optimized hyperparameters were omitted).

Table 2.19 – Out-of-sample accuracies (%) of the Controls VS Patients classifiers, trained on power features-based datasets.

Learner	**Dataset**		
	Acc	**Gyro**	**Both**
k-Nearest Neighbors	81.58	78.95	78.95
Support Vector Machine	78.95	84.21	78.95
Naïve Bayes	81.58	84.21	84.21
Discriminant Analysis	78.95	84.21	78.95
Decision Tree	84.21	92.21	89.47

The preliminary accuracies were lower than expected, averagely ranging from 80 to 85 %. Nevertheless, a good result was achieved by training a decision tree on the feature set based on angular rates, with an out-of-sample accuracy of 92.21 % obtained by using two features only, specifically TP_4 and PF_4 (see section 2.5.3 for more details). If considering the "Patient" label as the positive class and the "Control" label as the negative class, the model's k-fold sensitivity and specificity were 95.83 % and 85.71 %, respectively: this means that only one patient out of 24 was misclassified as healthy subject, and two control cases out of 14 were misclassified as patients. Such result is encouraging, but further investigations are needed in the next future (1) to assess whether or not new stronger features are necessary, and (2) to possibly modify the model optimization procedure (e.g. changing the metric used in the feature selection process). Moreover, parallel to the enlarging of the patients sample size already foreseen, new healthy subjects will be recruited to increase the range of control cases.

References

[1] P. Locatelli and D. Alimonti, "Differentiating essential tremor and parkinson's disease using a wearable sensor — a pilot study", in *2017 7th IEEE International Workshop on Advances in Sensors and Interfaces (IWASI)*, 2017, pp. 213–218. DOI: 10.1109/IWASI.2017.7974254.

[2] ——, "Advances in wearable sensor-based automatic tremor classification", in *2018 7th International Conference on Modern Circuits and Systems Technologies (MOCAST)*, 2018, pp. 1–4. DOI: 10.1109/MOCAST.2018.8376602.

[3] S. Fahn *et al.*, *Principles and Practice of Movement Disorders*. Elsevier Health Sciences, 2011.

[4] *U.S. Census. Projections of the population by selected age groups and sex for the United States: 2010 to 2050: national population projections*, Washington, DC: U.S. Census Bureau (Population Division), U.S. Department of Commerce, 2011.

[5] World Health Organization. (2012). Who: Ageing and life course, [Online]. Available: http://www.who.int.

[6] A. Gironell *et al.*, "Severity Stages in Essential Tremor: A Long-Term Retrospective Study Using the Glass Scale", *Tremor and Other Hyperkinetic Movements*, vol. 5, p. 299, 2015, ISSN: 2160-8288. DOI: 10.7916/D8DV1HQC.

[7] J. V. Hindle, "Ageing, neurodegeneration and Parkinson's disease", *Age and Ageing*, vol. 39, no. 2, pp. 156–161, 2010. DOI: 10.1093/ageing/afp223. [Online]. Available: http://dx.doi.org/10.1093/ageing/afp223.

[8] E. Louis and R. Ottman, "How Many People in the USA Have Essential Tremor? Deriving a Population Estimate Based on Epidemiological Data", *Tremor and Other Hyperkinetic Movements*, vol. 4, p. 259, 2014. DOI: 10.7916/D8TT4P4B.

[9] E. Louis, "Environmental Epidemiology of Essential Tremor", *Neuroepidemiology*, vol. 31, pp. 139–149, 2008. DOI: 10.1159/000151523.

[10] G. Kägi, K. P. Bhatia, and E. Tolosa, "The role of DAT-SPECT in movement disorders", *Journal of Neurology, Neurosurgery & Psychiatry*, vol. 81, no. 1, pp. 5–12, 2010. DOI: 10.1136/jnnp.2008.157370.

[11] H. Benamer, J. Patterson, D. Grosset, *et al.*, "Accurate differentiation of parkinsonism and essential tremor using visual assessment of [123I]-FP-CIT SPECT imaging: The [123I]-FP-CIT study group", vol. 15, pp. 503–510, 2000.

[12] M. N. Alam, B. Johnson, J. Gendreau, K. Tavakolian, C. Combs, and R. Fazel-Rezai, "Tremor quantification of Parkinson's disease - a pilot study", in *2016 IEEE International Conference on Electro Information Technology (EIT)*, 2016, pp. 755–759. DOI: 10.1109/EIT.2016.7535334.

[13] D. Surangsrirat, C. Thanawattano, *et al.*, "Support vector machine classification of Parkinson's disease and essential tremor subjects based on temporal fluctuation", in *Conference proceedings: Annual International Conference of the IEEE Engineering in Medicine and Biology Society. IEEE Engineering in Medicine and Biology Society. Conference*, vol. 2016, 2016, pp. 6389–6392.

[14] S. Morrison, K. Newell, and J. Kavanagh, "Differences in postural tremor dynamics with age and neurological disease", vol. 235, 2017.

[15] R. G. Garcia, A. H. Ballado, *et al.*, "Hand tremor analyzer using accelerometry for preliminary diagnosis, classification and monitoring of selected movement disorders", in *2016 6th IEEE International Conference on Control System, Computing and Engineering (ICCSCE)*, 2016, pp. 392–396. DOI: 10.1109/ICCSCE.2016.7893605.

[16] D. Comotti, M. Galizzi, and A. Vitali, "neMEMSi: One step forward in wireless attitude and heading reference systems", in *2014 International Symposium on Inertial Sensors and Systems (ISISS)*, 2014, pp. 1–4. DOI: 10.1109/ISISS.2014.6782521.

[17] D. Comotti, M. Caldara, M. Galizzi, P. Locatelli, and V. Re, "Inertial based hand position tracking for future applications in rehabilitation environments", in *2015 6th International Workshop on Advances in Sensors and Interfaces (IWASI)*, 2015, pp. 222–227. DOI: 10.1109/IWASI.2015.7184959.

[18] D. Comotti, M. Ermidoro, M. Galizzi, and A. Vitali, "Development of a wireless low-power multi-sensor network for motion tracking applications", in *2013 IEEE International Conference on Body Sensor Networks*, 2013, pp. 1–6. DOI: 10.1109/BSN.2013.6575466.

Chapter 3

Wearable-based Motor Skills Assessment

Parkinson's disease is a neurodegenerative disorder of the central nervous system that often impairs the sufferer's motor skills and speech. It is characterized by muscle rigidity, tremor, a slowing of physical movement (bradykinesia), and in extreme cases, a loss of physical movement (akinesia). As the incidence of Parkinson's rises significantly with age, and people are living longer, the prevalence of Parkinson's is set to rise dramatically in the future: the Global Burden of Disease Study 2015 [1] estimates that there may be nearly 13 million people with Parkinson's by 2040. On the other hand, among healthy people and in developed countries, sedentary lifestyle is a major health risk factor. In elderly people, for instance, such mobility limitation is worsened by the reduced self-confidence and the fear of falling, leading to a further motor deterioration.

This chapter presents the application of wireless Body Sensor Networks as simple and easy-to-use individual motor function assessment tools, designed to be used for both patients and healthy subjects in different monitoring conditions. The wearable devices were exploited to monitor the body during the execution of some standard motor skills assessment tests, such as the Extended Timed Up and Go test, the Six-Minute Walk Test and a set of stability tests. It is important to underline that in this thesis the goal of these applications is restricted to an overall evaluation of the functional motor skills of a subject: as such, the systems and the related algorithms developed should not be considered as gait analysis tools. Instead, such a detailed analysis level could be achieved in the near future by integrating additional features in the current applications. Lastly, the presented algorithms were developed based on data collected only from a sample of healthy subjects of different ages, due to the unavailability of parkinsonian patients up to the time of writing (autumn 2018). An observational longitudinal study involving 20 PD patients is foreseen to be started in the next future: further changes and

improvements of the developed methods might be implemented as soon as data from such patients will be available, in order to account for non-standard walking patterns.

Part of the material reported in this chapter has been published by the author of this thesis in *2014 IEEE 11th International Conference on Wearable and Implantable Body Sensor Networks (BSN)* [2] and in *2015 IEEE 12th International Conference on Wearable and Implantable Body Sensor Networks (BSN)* [3].

3.1 Rationale

The availability of miniaturized and low-cost MEMS sensors drove the development of inertial platforms and AHRS, achievable with the combination of sensors with a microcontroller able to determine the 3D orientation and other relevant parameters related to motion. By providing a wireless interface, such devices are particularly attractive as body motion tracking systems. The rehabilitation field can particularly profit from wireless body motion tracking, achieving low-cost but accurate analysis of body motor functionalities [4]. Such technologies are also attractive for mobile health applications, by allowing a patient to perform rehabilitation exercises at home and providing a feedback about the performance at the same time [5].

A possible application of a body motion tracking system involves the assessment of motor skills among parkinsonian population. As already described in section 1.3.2, Parkinson's disease is one of the most diffused neurodegenerative human diseases, with an estimated prevalence of 120–180 cases per 100 000 individuals in industrialized countries and a prevalence increasing with age. Although PD involves different structures in the central and peripheral nervous systems, its main characteristics are clinically defined on the basis of motor signs such as muscular rigidity, tremors at rest, bradykinesia, hypokinesia/akinesia, and postural instability. The existing drug therapies act on the dopaminergic nigrostriatal system and they are used to control the motor disabilities related to the disease. However, even with an optimal medical management, patients with PD still experience a deterioration of body functions during the disease progression. For this reason, a multidisciplinary management of the disease is necessary, and rehabilitation therapies have been included as an adjuvant to pharmacological and neurosurgical treatments. In recent years intensive rehabilitation (physical exercise, occupational therapy, treadmill with auditory and visual cues) emerged as a key role in the PD patients quality of life, in particular concerning motor function stabilization or improvement [6, 7]. Depending on the PD severity, the goals of physiotherapy are different. For patients with H&Y 1–2 the

aim is to prevent inactivity, fear to either move or fall, physical capacity and delay activity limitations, whilst in patients with more severe disease (H&Y 3–4) the goal is to improve motor coordination, such as gait and balance. Patients confined to bed or wheelchair (H&Y 5) need basic physiotherapy to maintain vital functions and prevent pressure sores. The PD patient's motor function assessment is a delicate issue since it has impact on medical decisions and thus on the patient quality of life. Monitoring human motor functions with instruments is indeed an objective way to overcome subjectivity. Optical systems for gait analysis are considered the gold standard, but due to the complexity and to their cost they are not used as a standard procedure on PD patients. In recent years wearable systems started to be adopted in clinical environments with promising results. Regarding patients affected by Parkinson's disease, Giuberti et al. [8] exploited two Shimmer inertial platforms in order to investigate the link between a kinematic characterization of a Leg Agility task and their UPDRS score; Gago et al. [9] made use of a body sensor network composed of 5 inertial measurement units to characterize the patient's postural stability and analyze the response to levodopa; Pierleoni et al. [10] presented an inertial sensor-based system for quantitative assessment of the tremor's type and its severity in the above mentioned subjects.

In addition to subjects with motor disabilities due to some specific diseases, the assessment of motor skills could play a fundamental role also in healthy subjects. As life expectancies increase and the major causes of death shift to the chronic diseases, populations are facing modern risks due to sedentary lifestyle. As a result, in 2009 insufficient physical activity was the 4th leading global risk for mortality in the world [11], responsible for 6% of deaths globally. Moreover, the increase in the elderly population has in turn rapidly increased fall-induced injuries, which are major issues for health and social care providers in the world [12]. Falls are the most serious and frequent home accidents among older people. As a matter of fact, they are the major reason for admission to hospital, even when no serious injury has occurred: Heinrich et al. reported the evidence of the economic burden of falls on society [13]. The safety of walking as well as coordinating abilities depend on the single stance stability of the subject [14]. In fact, in elderly people a single stance instability situation leads to a deterioration of motor experience: the risk associated to this condition induces a lack of self-confidence in the subject, which will reduce their motor habits and change their lifestyle becoming more hypokinetic. There is observational evidence that mid-life and older adults who participate in regular physical activity have reduced risk of moderate and severe functional limitations [15]. Pahor et al. [16] have demonstrated the potential for structured physical activity

as a feasible and effective intervention to reduce the burden of disability among vulnerable older people, in spite of functional decline in late life. Furthermore, improving physical conditions and motor abilities of people at risk contributes to the abatement of health and hospitalization costs. In order to prevent disability and fall-induced injuries, physical activity has to be considered as a therapeutic tool prescribed by doctors on the basis of diagnostic assessments on the patient. Such activity will have the goal to maintain and improve the motor condition of the subject, including coordination, balance, proprioception and reaction time.

3.2 What is a Body Sensor Network?

In the last decade, the technological progress in sensor miniaturization, their rapid diffusion and the consequent cost reduction supported the development of pervasive sensing for healthcare, wellbeing, sports and other applications that require "ubiquitous" and "pervasive" monitoring of physical and physiological parameters in any environment and without activity restriction and behaviour modification. In particular, the world has faced a rapid surge of interest in new sensing and monitoring devices for healthcare and the use of wireless wearable devices for clinical applications [17]. To address general issues related to wearable and implantable sensors and harness such technologies, the term BSN — Body Sensor Networks was coined by Prof Guang-Zhong Yang of Imperial College in the early 2000's.

Generally speaking, a body sensor network consists of a group of sensors attached to a subject in order to acquire physiological data. Recent advances in wireless technologies enabled mobile and continuous monitoring of patients, even during their daily life activities. In such healthcare systems, the information sensed at the patient's body is wirelessly transmitted to a base station, located in the nearby of the subject, and then delivered to a remote diagnosis center through a communication infrastructure. These systems may include the monitoring of either several physiological signals (e.g. electrocardiogram, heart rate, blood pressure, pulse oximetry) or physical signals, such as those related to movement (e.g. accelerations, angular velocities, forces). Figure 3.1 depicts an example of body sensor network: in this case, the base station is represented by the subject's smartphone.

The miniaturized biosensor design suitable for both wearable and implantable devices, low-power wireless communication and power scavenging techniques from the body represent some of the key research activities in the BSN field. On the

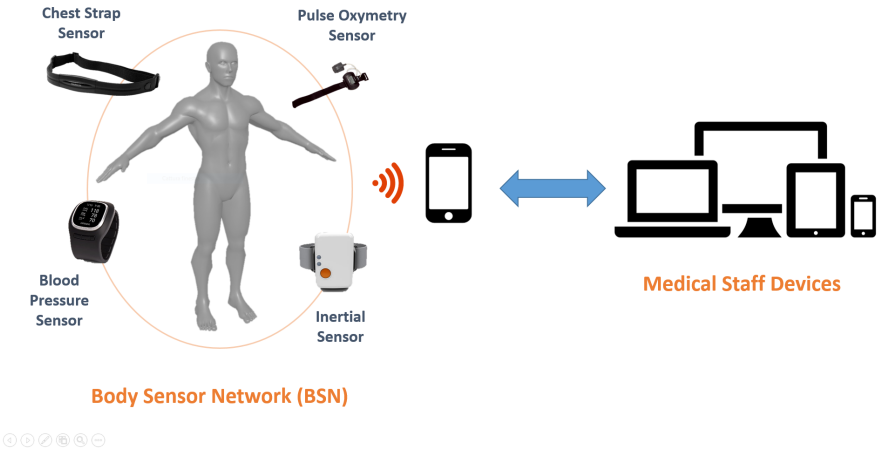


Figure 3.1 – Representation of a Body Sensor Network.

other hand, major technical difficulties need to be addressed: in fact, continuous sensing and monitoring require long-term stability of the sensors and low-power operation, as well as techniques to save and harvest energy.

In this chapter, a body sensor network composed of at most 5 magneto-inertial platforms named MuSe is considered. By means of elastic straps, these devices were mounted on the subjects in five possible locations: shanks, wrists and chest. A MuSe device integrates a 3D accelerometer, a 3D gyroscope and a 3D magnetometer, as well as an elaboration unit: by combining data from such sensors, the orientation of the device can be determined by means of an integrated sensor fusion algorithm (see section 2.2 for more details). However, an important aspect needs to be discussed before continuing in the section: the problem of the reference systems.

3.2.1 Quaternion-based Reference Systems Alignment

The mounting position of each MuSe device directly affects the acquired data and consequently the body orientation estimation. An alignment method was studied in order to determine the orientation of each node with respect to a common reference system independently of the device mounting position. The orientation of each platform is computed depending on the data measured by on-board sensors and is represented by means of quaternions. In this chapter, the following convention

$$q_{b,c}^{a} \tag{3.1}$$

will denote a quaternion representing the rotation from a reference system b to c with respect to the reference system a. Based on the quaternion properties and on

the current notation, the operation

$$q^a_{b,c} = q^a_{a,b} \otimes q^b_{b,c} \otimes {q^a_{a,b}}^* \tag{3.2}$$

makes it possible to change the reference system of the rotation described by $q^b_{b,c}$, from b to a. The symbol $\otimes$ represents the Hamilton product between quaternions.

Each node of the network can be referred to one of the following systems (see Figure 3.2):

- The global reference system (0), corresponding to the earth reference system and represented by the identity quaternion $\begin{bmatrix}1 & 0 & 0 & 0\end{bmatrix}$;
- The *person*'s reference system (1), representing a local reference system achievable from the global one by applying a yaw offset:

$$q^0_{0,1} = \begin{bmatrix}q_{1w} & 0 & 0 & q_{1z}\end{bmatrix}; \tag{3.3}$$

- The reference system of each device (2), which depends on the specific positioning conditions (mounting), and is achieved from the global reference system by the specific rotation $q^0_{0,2}$;
- The reference system of each device during the experiment (3), identified by the quaternion $q^0_{0,3}$ estimated on board.

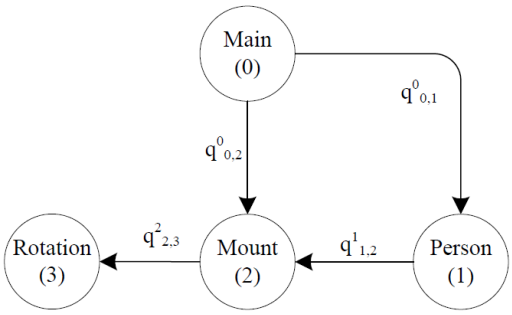


Figure 3.2 – Quaternion-based description of the different reference systems links.

In order to represent all the rotations with respect to the reference system of the subject, the quaternion $q^1_{2,3}$ is needed, which is given by:

$$q^1_{2,3} = q^1_{1,2} \otimes q^2_{2,3} \otimes {q^1_{1,2}}^*, \tag{3.4}$$

where

$$q^1_{1,2} = {q^0_{0,1}}^* \otimes q^0_{0,2} \tag{3.5}$$

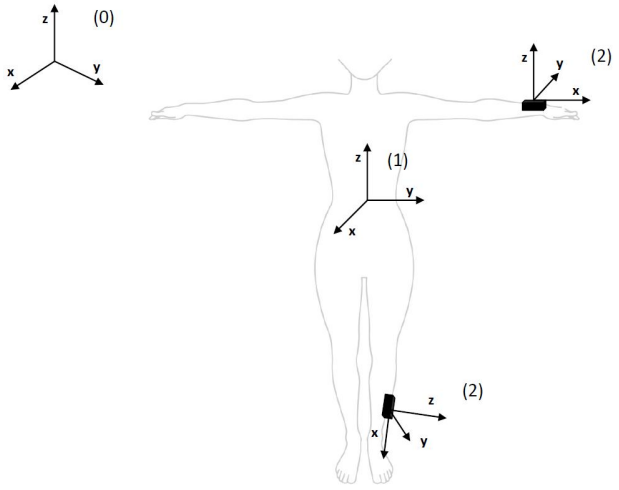


Figure 3.3 – Physical representation of the starting reference systems (0), (1) and (2).

denotes the rotation between the subject's reference system and the module mounting position, whereas

$$q_{2,3}^{2} = {q_{0,2}^{0}}^{*} \otimes q_{0,3}^{0} \tag{3.6}$$

represents the module orientation during the exercise with respect to its initial position. This latter quaternion is based on the orientation estimated by each AHRS. By using eq. (3.5) and eq. (3.6), an alternative notation of eq. (3.4) based only on quaternions referred to the main reference system can be achieved:

$$\begin{aligned} q_{2,3}^{1} &= q_{1,2}^{1} \otimes q_{2,3}^{2} \otimes {q_{1,2}^{1}}^{*} \\ &= {q_{0,1}^{0}}^{*} \otimes q_{0,2}^{0} \otimes {q_{0,2}^{0}}^{*} \otimes q_{0,3}^{0} \otimes \left[{q_{0,1}^{0}}^{*} \otimes q_{0,2}^{0} \right]^{*} \\ &= {q_{0,1}^{0}}^{*} \otimes q_{0,3}^{0} \otimes {q_{0,2}^{0}}^{*} \otimes q_{0,1}^{0}. \end{aligned} \tag{3.7}$$

This calibration method establishes the orientation of each node of the network with respect to the reference system of the subject. In particular each rotation around x, y and z axis of the body reference system are addressed as roll, pitch and yaw.

3.2.2 Acceleration-based Vertical Alignment

Provided that the initial yaw offset between the subject's reference system and the earth reference system is known, the alignment procedure previously described allows to determine the mounting position of each node of the network with respect to the subject's reference system, and then to compensate these differences for the whole acquisition process. All the data required by the procedure come in the form

of quaternion: in order to obtain such information, raw data from magnetometer as well as a sensor fusion algorithm are needed (see section 2.2.2).

Since magnetometer measurements are strongly influenced by soft- and hard-iron interferences caused by magnetic objects surrounding the device, an accurate calibration needs to be performed before starting any data collection session. If acquisitions are always performed in the same volume of space and no changes in the surrounding environment occur, nodes of the network need to be calibrated only the very first time: this is the typical case for acquisitions performed in supervised environments, such as a movie set or a hospital gym. However, two problems need to be taken into account.

1. The calibration compensates only for interferences in the vicinity of the device, in a sphere of less than 1 m-radius: for acquisition sessions involving walking, as in the 6MWT or in the ETUG test, orientation estimates may still be altered by magnetic interferences when the device is distant from the calibration point.

2. The calibration procedure is not always an easy and quick task, especially when dealing with a network composed of several devices. In a telemedicine context, where the patient is asked to perform some rehabilitation exercises at home while wearing the BSN, the combination of an unsupervised environment and a possible user inexperience could lead to an incorrect calibration procedure and therefore to altered orientation estimates.

For this reason, a different procedure for axes alignment was studied. By using only data from the accelerometer, it is possible to determine the difference between the orientation of the device's vertical axis and the subject's one: in this thesis, this procedure is therefore referred to as *vertical alignment*.

When an inertial device is still (static conditions), the on-board accelerometer measures the gravity acceleration $\vec{g}$. Provided that this sensor and the platform share the same reference system, if the vertical axis of the device is perpendicular to the ground, the measured acceleration will equal the gravity one so that $\vec{a} = (a_x, a_y, a_z) = (0, 0, 1)$. If not, part of the gravity acceleration will be measured on all the three axes, depending on the particular device's orientation. Thanks to Euler angles, the orientation of the device can be described as a series of successive rotations in the following order: a yaw rotation by an angle ψ around the original z axis; a pitch rotation by an angle ϑ around the new Y axis; and a roll rotation by an angle φ around the new X axis. Starting from a reference acceleration vector $(\bar{a}_x, \bar{a}_y, \bar{a}_z)$ measured in static conditions, it is then possible to estimate the pitch

and the roll angles that define that orientation, by using the following trigonometry formulas respectively:

$$\vartheta = -\operatorname{asin}\left(\frac{\bar{a}_x}{\sqrt{\bar{a}_x^2 + \bar{a}_y^2 + \bar{a}_z^2}}\right) \tag{3.8}$$

$$\varphi = \operatorname{atan}\left(\frac{\bar{a}_y}{\bar{a}_z}\right) \tag{3.9}$$

By compensating for these two angles, the device will finally have its z axis perpendicular to the ground. It is however important to underline that this procedure can not compensate for any yaw offset, due to the fact that an accelerometer measures the same gravity acceleration components independently from any initial rotation around the vertical axis: for this reason, no full reference system alignment can be performed by using this procedure.

The pitch and roll angles compensation can be achieved by using quaternions. Recalling their definition (see section 2.2.2), a quaternion represents a rotation of an object by an angle α around the rotation axis $\vec{u} = (u_x, u_y, u_z)$ by means of the following notation:

$$q = \left[\cos\frac{\alpha}{2},\ u_x \cdot \sin\frac{\alpha}{2},\ u_y \cdot \sin\frac{\alpha}{2},\ u_z \cdot \sin\frac{\alpha}{2}\right]. \tag{3.10}$$

By using eq. (3.10) it is possible to get the equivalent quaternion representation of the measured pitch and roll angles

$$\begin{aligned} q_\vartheta &= \left[\cos\frac{\alpha}{2},\ 0,\ \sin\frac{\alpha}{2},\ 0\right], \\ q_\varphi &= \left[\cos\frac{\alpha}{2},\ \sin\frac{\alpha}{2},\ 0,\ 0\right], \end{aligned} \tag{3.11}$$

as well as their combination:

$$q_{va} = q_\vartheta \otimes q_\varphi. \tag{3.12}$$

In this way, by applying the eq. (3.2) it is possible to obtain the value of any acceleration measured in the device's reference system as if it were measured in a reference system with the vertical axis parallel to the direction of the gravity acceleration. The central quaternion required by eq. (3.2) is obtained by casting the acceleration vector to a quaternion with null real part:

$$q_{acc} = q_{va} \otimes [0\ a_x\ a_y\ a_z] \otimes {q_{va}}^* \tag{3.13}$$

On the contrary, the "rotated" acceleration vector can be extracted from the q_{acc} vector part. Once the quaternion q_{va} is computed from accelerations information,

eq. (3.13) can be also applied to the measured angular velocities to change their reference system. Figure 3.4 shows the effects of the vertical alignment procedure when performed on accelerations.

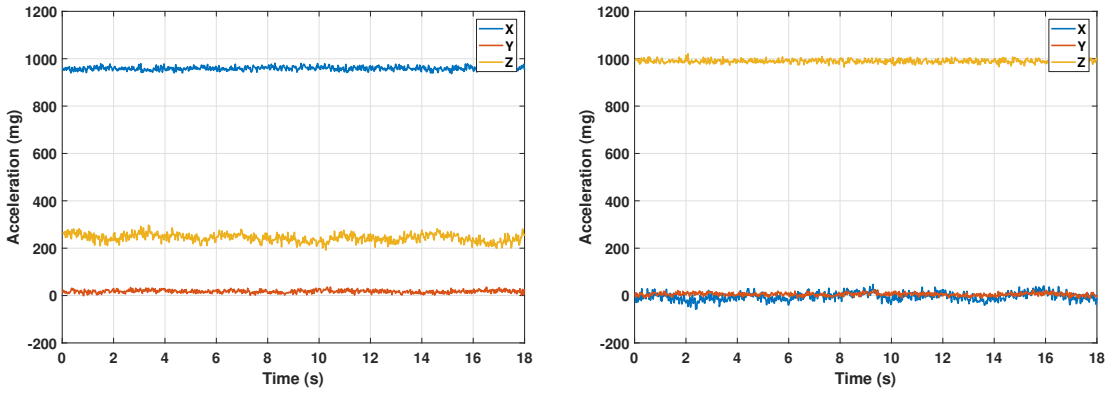


Figure 3.4 – Effect of the vertical alignment procedure performed on acceleration measurements. On the left, the gravity acceleration is measured on the x and z axis due to the mounting position of the device. After the alignment procedure (right), the gravity acceleration is mainly measured on the z axis, as expected.

3.3 Common Walking Features

Among the different motor skills assessment tests, those based on walking share some fundamental features, and with them, the extraction algorithms. In particular, two classes of main events typically occur during a walking test: steps and turnings. The correct detection of these events is of fundamental importance for a motor skills assessment tool, since most of the exercise parameters directly rely on them.

In this section, different implemented feature extraction approaches based both on quaternions and raw inertial data are described. The reported methods refer to possible data collected by a body sensor network composed of 5 nodes: one node mounted on the chest of the person, two nodes on their wrists and two nodes on their lower legs. Moreover, the presented methods are thought to be used on data expressed in the reference system of the subject: therefore, a preparatory phase of reference systems alignment is required before running the algorithms. For more details on how to perform this latter task, see sections 3.2.1 and 3.2.2 for quaternion-based and acceleration-based alignment procedures, respectively.

Once again it is important to underline that, since quaternions are computed by merging both inertial and magnetic data, procedures based on quaternion require an accurate calibration before the data acquisition in order to compensate for any magnetic interferences that may be present in the surrounding environment.

3.3.1 Step Detection

The identification of steps is a fundamental task when dealing with walking tests. In a supervised environment such as an hospital gym, this task is performed either by manually counting, or by using some motion tracking systems based on cameras. In the first case, the step counting may be affected by human errors, in particular for tests in which long distances are traveled. On the other hand, the second automatic approach is more accurate, but may be very expensive and requires a set prepared ad hoc. In this context, a simpler and cheaper approach is represented by inertial sensors mounted on specific body parts, such as feet, lower legs or hips — that is, body parts mainly involved when walking.

Different parameters can be extracted which are related to steps, in particular from the gait cycle: among these, the stance and swing phases duration as well as the stride length are the most commonly estimated ones. However, the goal of the developed procedures is only to identify steps, because by simply counting their number during the execution of the exercise it is possible to infer several motor skills information about the subject. Therefore, no detailed gait parameter will be extracted in the following algorithms: further work will be needed in the next future to integrate these new functionalities.

Quaternion-based Approach

Steps are associated with legs movements, in particular those performed within the sagittal plane. By converting the quaternions estimated by the devices mounted on the shanks into Euler angles, the orientation of the lower legs is well described by the pitch angles.

Figure 3.5 depicts the typical pattern of the lower leg angles during a walk. Depending on the pitch convention, the minimum (maximum) angle is achieved when the leg is fully extended forward. Then the angle increases (decreases) as the foot gets closer to the ground, until an inflexion point is achieved in correspondence to the foot mid-stance, i.e. when the foot is completely in contact with the ground and the body weight is supported only by one leg. Finally, the angle reaches its maximum (minimum) when the leg is flexed backward and the swing phase starts.

Each pitch maximum (or minimum) point therefore corresponds to a single step, and its identification can be easily performed by using a peak detection algorithm. Several approaches to the peak detection are described in literature, ranging from naïve threshold-based methods, to more complex differentiator filters combined with a zero-crossings algorithm. The Todd-Andrews method is another frequently used peak detection algorithm, in particular when this task needs to be performed

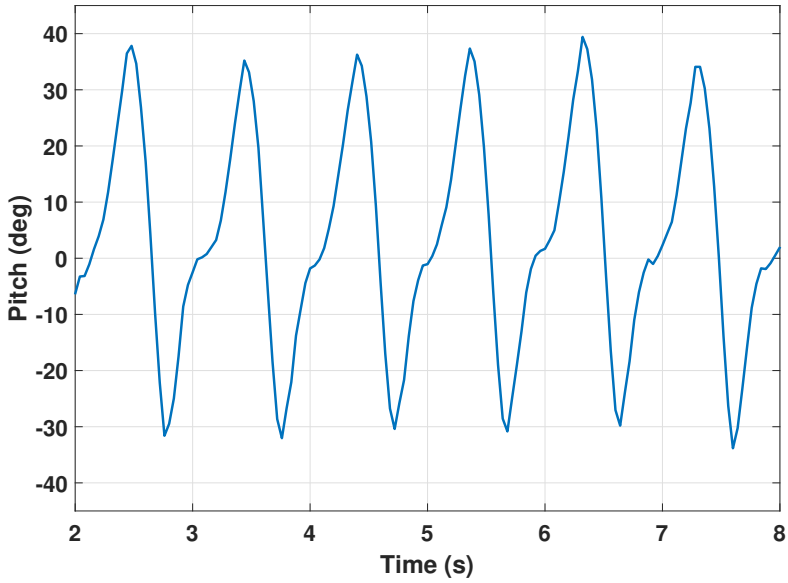


Figure 3.5 – Pitch angles of a subject's lower leg during a walk.

in real-time [18]. The peak detection technique selected for this analysis stage was a user-defined algorithm provided by the MATLAB software: it combines a first sample differentiation with specific requirements that a local maximum has to meet to be considered a valid peak, such as being greater than a minimum height, or being far enough from another valid peak. As an example, Figure 3.6 shows the results of the MATLAB peak detection algorithm when used with the signal depicted in Figure 3.5: the only requirement set for the peak search was a minimum prominence (i.e. how much the peak has to stand out from adjacent data) of 15 degrees. Valleys were detected by running the algorithm on the reversed signal.

Inertial Data-based Approach

Other than being associated with a generic leg movement, a step can be intended as the combination of two main actions: a leg rotation and a foot impact onto the ground. Because of this and due to the nature of these two actions, an inertial data-based approach was studied. Dependently on the available node of the BSN and on the on-board sensors, the developed algorithm can work with one among three possible data sources:

- The angular rates of the lower legs around the lateral axis, which encode the leg rotational movement;
- The vertical accelerations of the lower legs, which encode the impact of the foot onto the ground;

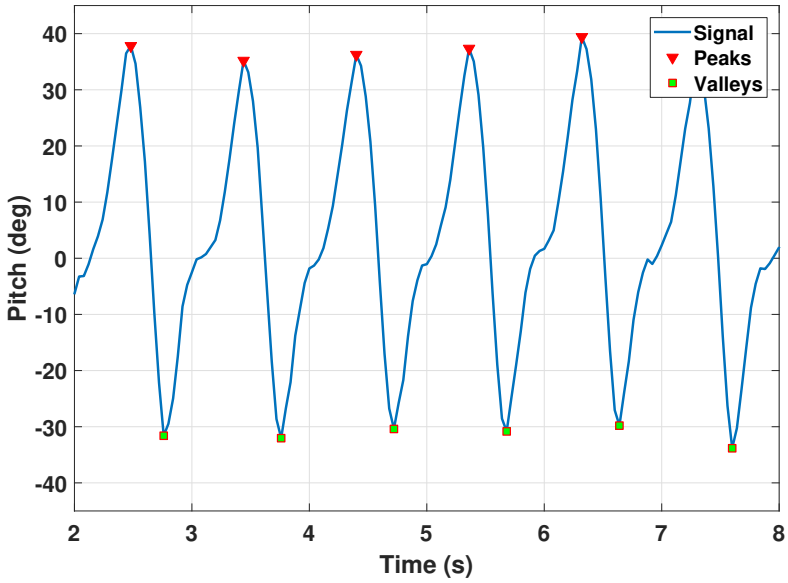


Figure 3.6 – Detection of peaks and valleys among the pitch angles of the lower leg. The minimum peak prominence was set to be 15 degrees.

- The vertical accelerations of the trunk, which are partially affected by the foot impact.

Regarding the last two data sources, the acceleration-based alignment procedure automatically aligns the z axis of the device to the subject's vertical axis, as already described: therefore, the relevant information can be easily obtained by considering the z component of the accelerations measured by the devices mounted on the chest and the legs of the subject. Unfortunately, this does not apply to angular velocities. In fact, the relevant information is the angular speed around the subject's lateral axis, and since the vertical alignment procedure does not account for the yaw compensation, the x and y axes of the device do not necessarily correspond to the subject's ones: the desired information may thus be encoded either within the x component only, the y component only, or a combination of both, depending on the specific device's mounting position. The typical locations for the lower leg devices are either the front of the shin or its side. If devices were mounted with the very same orientation among different patients, the algorithm could be focused to work with data along a known specific axis. In this thesis, in order to account for different mounting positions among subjects, the gyroscope axis selected to perform the step detection was the one showing the greatest signal standard deviation.

Once the relevant axis is selected, steps can be detected by identifying gait events within the signal: in literature, the most commonly detected are the initial foot contact (IC) and the final foot contact (FC), that is, the instants at which the heel touches the ground, and the foot is completely flat, respectively. In several studies,

such as those presented by Zijlstra [19] and González [20], this kind of analyis was based on some low-pass filtering either on vertical or anterior-posterior accelerations, followed by differentiation and zero-crossing detection to determine local minima or maxima points corresponding to IC and FC events. However, the application of a digital filter can distort the signal and affect the accuracy of the subsequent estimate. Recently, authors are thus using other techniques to find gait events. For instance, Trojaniello and Cereatti [21] identified IC and FC instants as minima points in the angular velocity and anterior-posterior acceleration respectively, by using two inertial platforms mounted on each shank: the events search was performed during specific time intervals only, defined as the swing duration of the non-supporting leg. Again, McCamley and Donati [22] proposed an approach based on the *continuous wavelet transform* (CWT) of the trunk vertical accelerations to detect IC and FC events. Both these procedures were validated and results confirmed their high accuracy when used in supervised assessments and indoor free walking [23], with the shanks-based method slightly better than the trunk-based one. Although its lower accuracy however, in the work described within this thesis the McCamley's method was preferred to the one presented in [21], because of its more generic approach: one of the goals, in fact, was to investigate the possibility of using the same procedure independently from the data source (either shanks- or trunk-mounted devices).

Continuous Wavelet Transform Let $\psi(t)$ be a finite energy continuous function having an average of zero, this is called a *mother wavelet* (or *wavelet function*). This function can be scaled (or dilated) by a factor of a and translated by a factor of b to give

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\,\psi\left(\frac{t-b}{a}\right). \tag{3.14}$$

The CWT of a signal $x(t)$ at scale $a \in \mathbb{R}^+$ and translational value $b \in \mathbb{R}$ is defined as

$$X_w(a,b) = \int_{-\infty}^{\infty} x(t)\,\overline{\psi_{a,b}(t)}\,\mathrm{d}t\,, \tag{3.15}$$

where the overline represents the complex conjugate operation. For real-valued mother wavelets, $\overline{\psi_{a,b}(t)} = \psi_{a,b}(t)$ In definition, the continuous wavelet transform is a convolution of the input data sequence with a set of functions generated by the mother wavelet.

As reported in [24], if the mother wavelet is defined as the reverse of the nth derivative of a smoothing function $\vartheta(t)$ having a fast decay and a non-zero constant integral,

$$\psi(t) = (-1)^n\,\frac{d^n\vartheta(t)}{dt^n}, \tag{3.16}$$

then the CWT X_w represents the derivative of the signal $x(t)$ smoothed by the scaled, weighted average kernel

$$\vartheta_a(t) = \frac{1}{\sqrt{a}}\,\vartheta\left(\frac{-t}{a}\right). \tag{3.17}$$

As a result, the CWT with the given wavelet function $\psi(t)$ has the combined properties of data smoothing and differentiation. A small value of the scale a results in high noise sensitivity, while a large value of a is related to a large averaging domain and thus results in strong noise cancellation.

Taking advantage of this feature, by selecting a proper smoothing function $\vartheta(t)$ and the relative scale, a signal can be smoothed by first integration and then CWT-based differentiation (see Figure 3.7). The advantages of this smoothing action with respect to standard IIR and FIR filters, is that neither signal leakage nor delays occur. As reported in [22], instants of IC can then be estimated from the minima of the smoothed vertical acceleration, while peaks in the signal derived from a second differentiation can be used to estimate instants of FC. As the current goal of the presented procedure is to detect steps, only the first half of the McCamley's technique was adopted, that is, the consecutive application of integration and CWT-based differentiation to smooth the signal. However, if the identification of gait events should ever be needed in future works, few efforts will be required to add the second half of the technique to the current procedure.

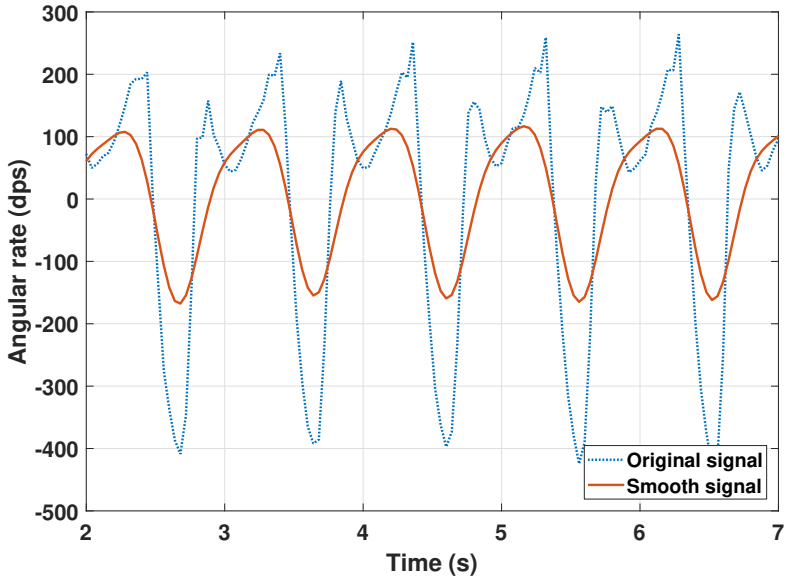


Figure 3.7 – Left shank's angular rate smoothed by consecutive integration and CWT differentiation. The selected wavelet $\vartheta(t)$ is the first derivative of a Gaussian curve, and the scale parameter $a = 5$ (selected empirically).

Since after the smoothing each local maximum of the signal is coupled with an IC event (the latter represented by acceleration's minima points), the user-defined peak detection algorithm described in the previous paragraph can finally be used to identify such maxima and therefore steps.

Accelerations and angular velocities are analyzed using the same procedure. However, while the gyroscope provides data containing information (steps) related only to the leg on which it is mounted, it is impossible to distinguish left steps from right ones by only analyzing trunk vertical accelerations, because the device mounted on the chest is affected in the same way by accelerations due to the impacts of the left foot and the right foot on the ground. Moreover, acceleration signals measured by the devices mounted on the shanks usually present two peaks per step: a higher one due to the foot impact, and a smaller one due to the swing phase. This makes it possible to use a single inertial device to detect both left and right steps, thus reducing the number of nodes of the network, but in such way their distinction becomes a hard task and it is not always achievable. Figure 3.7 and Figure 3.8 clearly show what is discussed here. The first one depicts the dominant angular speed measured by the device mounted on the left shank, in its original and smoothed versions: a total number of 5 steps can be observed by counting the signal's peaks. The second one, on the other hand, represents trunk and left shank vertical accelerations: for both data sources, a total number of 10 peaks can be counted, which correspond to the actual number of steps taken by the subject.

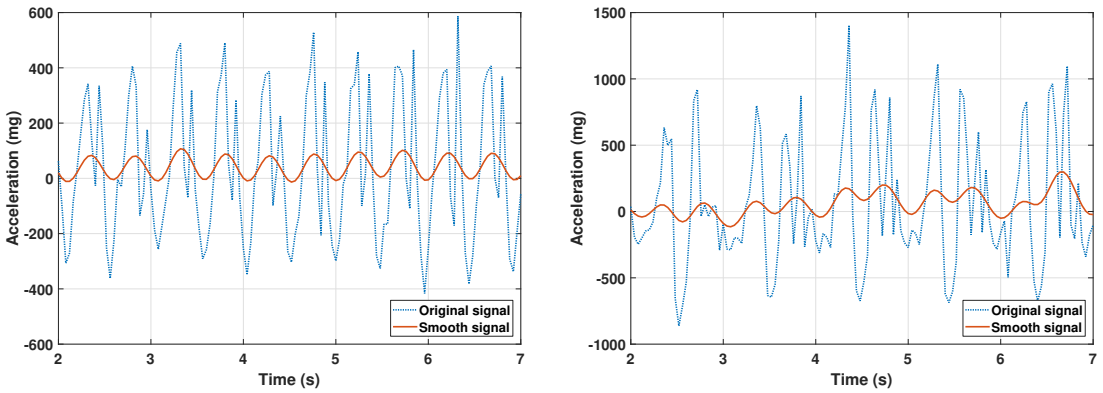


Figure 3.8 – Trunk vertical accelerations (on the left) and left shank vertical accelerations (on the right), as well as their smoothed versions performed as in Figure 3.7.

In conclusion, the inertial-based step detection procedure can be summarized by the following steps:

1. Select the desired data source among:
    a) Gyroscopes of devices mounted on the shanks (both left and right);

   b) Accelerometer of devices mounted on the shanks (either left or right);
   c) Accelerometer of the device mounted on the trunk.

2. Align the device's vertical axis with the subject's one by means of the acceleration-based alignment procedure.

3. Get the relevant axis on the basis of the selected data source:
   a) Vertical (z) component, when dealing with accelerations;
   b) Axis showing the component with the highest standard deviation, when dealing with angular rates.

4. Detrend the signal by removing its mean value.

5. Compute the approximate cumulative integral of the zero-mean signal by using the trapezoidal method.

6. Differentiate the integral by means of the Gaussian CWT.

7. Find steps (that is, local maxima) in the obtained smooth signal by using the user-defined peak detection algorithm.

3.3.2 Turns

The second main event that can occur during a walking test is the turning, in particular the so-called *U-turns*. Many of these tests in fact involve walking straight on a path for a certain amount of meters, then the subject has to turn and walk back. This operation can occur many times during the same assessment dependently on the specific test. Moreover, in many occasions the number of meters to walk in both directions is fixed and known. For instance, the Six-Minute Walk Test (6MWT) is based on a straight path of 30 meters-length which the subject has to walk back and forth as many times as possible for a total amount of time of 6 minutes; another example is the Timed Up and Go (TUG) test, a version of which involves to walk for 3 meters on a straight line, turn around a cone and walk back to the starting point: both of these tests will be deeper described in sections 3.4 and 3.5. In these contexts, the identification of turns becomes very important since it allows for the segmentation of the exercise into straight and turn walking bouts; moreover, if the length of the straight bouts is known, a raw estimation of the walked distance can be easily obtained.

Although turns occur in many walking test, there is no globally-accepted definition describing what a "turn" actually is. This is mainly due to the fact that a turn is a

complex movement that involves both legs and chest: some authors such as [25] identify it as the change in the direction of progression determined on a step-by-step basis; others define it as the rotation of the trunk around the vertical axis, of a angle greater than a minimum threshold [26]. These definitions mostly depend on the mounting position of the available devices, either legs or trunk. Unfortunately, they do not always overlap, because in many occasions one part of the body starts the turn movement before the others. In this thesis, different definition of U-turn will be adopted on the basis of the available data and the used algorithm: this approach limits the numeric comparison among them and with other methods in literature, but as far they are robust and accurate enough they will be still useful for longitudinal studies.

Quaternion-based Approach

The identification (beginning and end) of U-turns is a simple task if the body sensor network is made up of magneto-inertial devices. In this case in fact, the orientation of each node is either already available thanks to an embedded sensor fusion algorithm, or can be computed off-line by using raw data from accelerometer, gyroscope and magnetometer.

This procedure relies on the orientation of the device mounted on the chest. Once the provided quaternions are expressed with respect to the subject's reference system (by means of the alignment procedure) and converted into Euler angles, the trunk yaw angle describes the rotation of the chest around the vertical axis. In general, the subject faces the walking path at the beginning of the exercise, therefore during the outward route the yaw angle equals 0 degrees; on the contrary, during the backward route the orientation of the subject's trunk is represented by a yaw angle of ± 180 degrees depending on the direction of the turn (see Figure 3.9).

U-turns can thus be intended as a change of the trunk's orientation from 0 to ± 180 degrees, and vice versa. In order to account for the turning direction (and thus the sign of the angle), the absolute value of the yaw angle is first computed. Then, the signal is smoothed by a moving average filter with 5 samples-span. Finally, a 90 degrees threshold is used to identify turning midpoints: for each of them, the turn duration is defined as the interval from the first point greater than 45 degrees right before the midpoint and the last point lower than 135 degrees right after it (see Figure 3.10). The two thresholds may be switched depending on the starting angle of the current turn (either 0 or 180 degrees).

This method holds for turns performed at any speed, since it relies on the trunk yaw angle with respect to its starting orientation. However, it is highly dependent

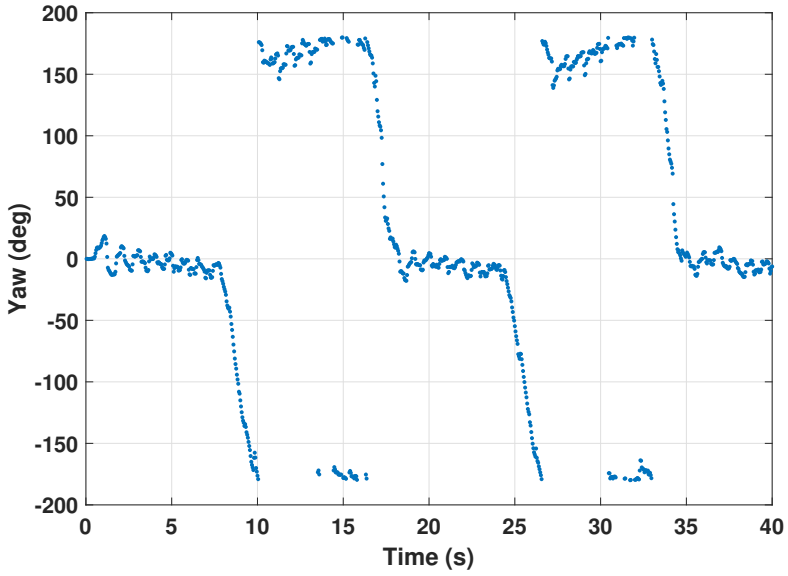


Figure 3.9 – Yaw angle of the trunk during the execution of the 6MWT. Outward routes (points at 0 degrees) and backward routes (points at ±180 degrees) are clearly visible.

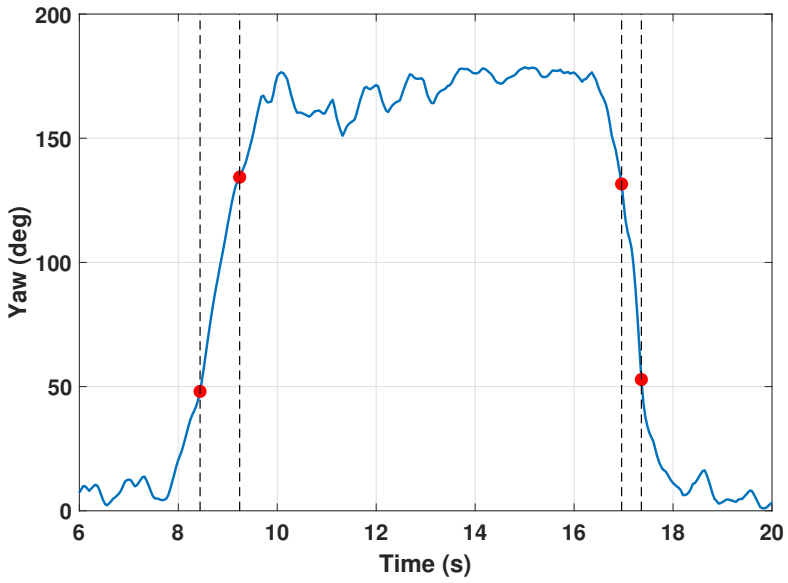


Figure 3.10 – Detection of starting and ending points of the U-turns by thresholding the yaw angle. Turning times are defined as the time interval between dashed black lines.

on the data quality, which in turn strongly depends on the presence of magnetic interferences and the ability of the calibration procedure to compensate for them.

Inertial Data-based Approach

Since a turn is always characterized by a rotation of the trunk, the gyroscope integrated on the device mounted on the subject's chest provides useful information.

The turn involves the rotation a the body around the vertical axis, therefore by using the acceleration-based vertical alignment procedure with angular rate measured by the gyroscope, the final z axis will be the one of interest.

During a walk, the trunk follows the movements of the legs to favor the gait. In this sense, small short rotations around the vertical axis due to steps are measured during the overall execution of the exercise. However, provided that the turn is performed at high-enough speed, the angular rate peak associated with a U-turn is dominant with respect to the smaller ones, in terms of amplitude and duration (see Figure 3.11).

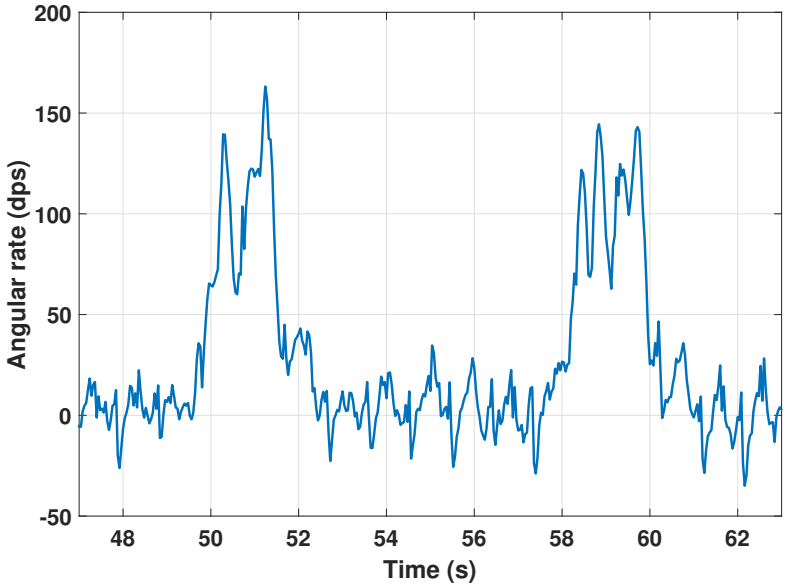


Figure 3.11 – Vertical angular rate of the trunk device. The small short peaks represent movements of the trunk associated with steps, whereas the big large peaks represent the U-turns.

By properly filtering the signal, it is possible to smooth small peaks and leave unaltered the bigger ones, each of which represents a U-turn: the selected smoothing technique is based on the *discrete wavelet transform* (DWT).

The DWT of a sampled signal x is calculated by passing it through a series of filters. Firstly, the samples are passed through a low-pass filter with impulse response g, resulting in a convolution of the two:

$$y[n] = (x * g)\,[n] = \sum_{k=-\infty}^{\infty} x[k]\, g[n-k]. \tag{3.18}$$

At the same time, the signal is also decomposed using a high-pass filter h. The outputs of the low-pass and the high-pass filter are called *approximation coefficients* and *detail coefficients*, respectively. The two filters are generated by using the same

mother wavelet, and each of them has to be the quadrature mirror filter of the other one (i.e. one filter's response has to be the mirror image about $\pi/2$ of that of the second one). Since half the frequencies of the signal have been removed by the filters, half of the samples of each coefficients set can be discarded according to the Nyquist's rule by using dyadic decimation (downsampling). A block diagram of the described signal decomposition is depicted in Figure 3.12.

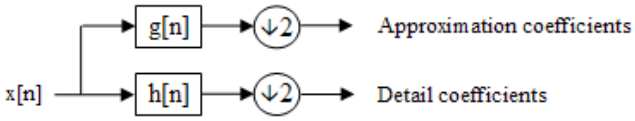


Figure 3.12 – Block diagram of the DWT decomposition.
Source: *Wikimedia Commons*, by Johnteslade

This decomposition can be repeated: the approximation coefficients are decomposed with high- and low-pass filters and then down-sampled. This process is represented as a binary tree known as a *filter bank*. At each level of the filter bank, the signal is decomposed into low and high frequencies: for a 2 level filter bank, for instance, level 1 detail coefficients include frequencies from $\frac{f_s}{2}$ to f_s, the level 2 detail coefficients from $\frac{f_s}{4}$ to $\frac{f_s}{2}$, and the level 2 approximation coefficients from 0 to $\frac{f_s}{4}$.

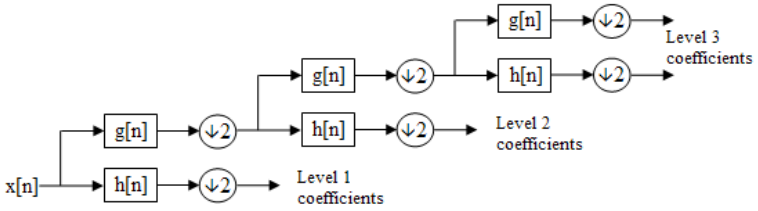


Figure 3.13 – A 3 level DWT filter bank.
Source: *Wikimedia Commons*, by Johnteslade

The original signal x can be reconstructed from the detail and approximation coefficients. But, if some of the coefficient sets are thresholded, components of the reconstructed signal that falls in the frequency band related to those sets will be attenuated: this feature reflects the behavior of a standard filter.

The angular rate smooth is performed by decomposing the signal by means of a 4 level filter bank based on the Daubachies 4 mother wavelet, resetting all sets but the level 4 approximation coefficients, and then reconstructing the signal: the effect of this smoothing action is depicted in Figure 3.14.

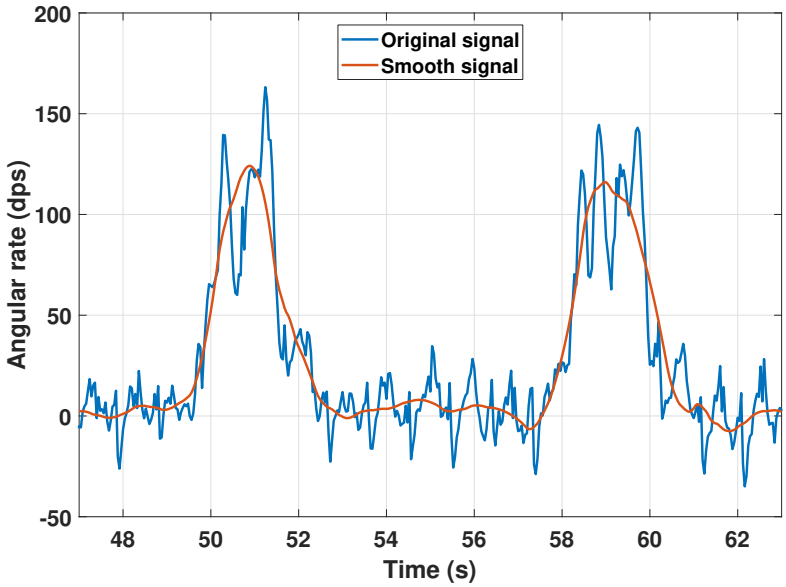


Figure 3.14 – Angular rate smoothed by using DWT filtering. The selected mother wavelet is a Daubachies 4.

After the signal smoothing, U-turns are clearly visible, each of which appearing as a quite symmetric curves with a peak in correspondence to the maximum angular velocity: this shape suggests the fit of the original signal with a series of Gaussian curves, which makes it feasible to formally describe the turns. In some cases, however, the number of U-turns is not known a priori, thus it is not possible to determine the number of Gaussian curves to be fit to the signal. In these cases, the following procedure can be adopted:

- The smooth angular rate signal is analyzed by using the user-defined peak detection algorithm described in section 3.3.1, in order to determine the number n of peaks (and thus, the number of U-turns) in the signal.

- The original, non-smoothed signal is then segmented into n parts, each of which starts and ends at midpoints between two consecutive peaks. In this way, each segment includes only one peak. For peaks close to the ends of the signal, the starting (ending) segment point is represented by the first (last) point of the signal.

- Each segment is separately fitted with a Gaussian curve in the form $f(x) = ae^{-(\frac{x-b}{c})^2}$, and the parameters a, b and c are extracted.

The described procedure also holds for signal characterized only by one U-turn. For each determined Gaussian curve, the position of the maximum represents the turn midpoint. In this approach, it was decided to define the turn duration td as the

curve's *full width at half maximum* (FWHM), which can be easily computed from the parameter c as

$$FWHM = 2 \cdot \sqrt{\ln(2)} \cdot c. \tag{3.19}$$

Figure 3.15 shows the fit of two Gaussian curves to the original angular rate signal. The dashed lines represent the starting and ending turn instants, defined by the value of the curve's FWHM.

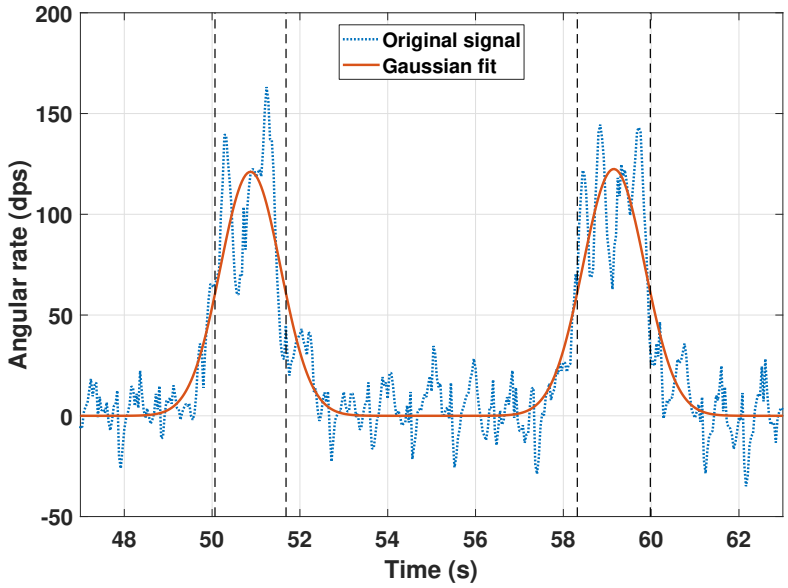


Figure 3.15 – Angular rate fit with Gaussian curves. The vertical dashed lines represent points at which the FWHM were evaluated.

3.4 Six-Minute Walk Test

Functional endurance is necessary for individuals to live independently without accommodation in community settings. The Six-Minute Walk Test (6MWT) is a simple and inexpensive method of indirectly assessing physical capability that is widely available and commonly used. This test measures the distance that a patient can quickly walk on a flat, hard surface in a period of 6 minutes. Although it was originally developed to assess cardiorespiratory and cardiovascular endurance, the 6MWT has been applied for assess walking disability in different neurological diseases such as stroke, multiple sclerosis, Alzheimer disease, and Parkinson's disease. In particular, the 6MWT is a validated motor outcome for PD [27], and the distance traveled during test (6MWD) may be indicative for evaluating the risk of falls [28]. In addition, according to the New York Heart Association (NYHA) functional classification the 6MWD has been shown to be a sensitive measure of the severity of heart disease.

According to the official statement of the American Thoracic Society (ATS) [29], practical guidelines for the 6MWT are as follows.

- The test should be performed indoors, along a long, flat, straight, enclosed corridor with a hard surface.
- The patient has to walk back and forth in such corridor as far as possible for 6 minutes.
- The walking course must be 30 meters in length, and turnaround points should be marked with cones.
- The patient is permitted to slow down, to stop and to rest if necessary, but they have to resume walking as soon as they can.

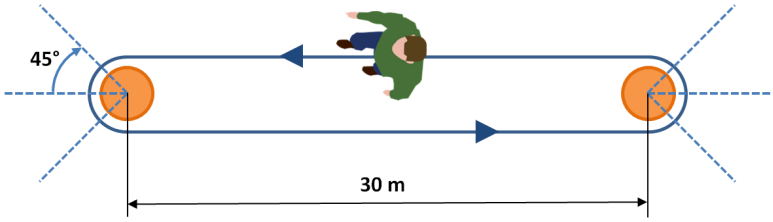


Figure 3.16 – Top view representation of the Six-Minute Walk Test.

The self-paced 6MWT assesses the submaximal level of functional capacity. Most patients do not achieve maximal exercise capacity during the 6MWT: instead, they choose their own intensity of exercise and are allowed to stop and rest during the test. However, because most activities of daily living are performed at submaximal levels of exertion, the 6MWD may better reflect the functional exercise level for daily physical activities. In recent decades, the 6MWD has been validated as a general indicator of overall physical performance and mobility for older people [30]. Optimal reference equations from healthy population-based samples using standardized 6MWT methods are not yet available. In one study, the median 6MWD was approximately 580 m for 117 healthy men and 500 m for 173 healthy women [31]. A mean 6MWD of 630 m was reported by another study of 51 healthy older adults [32]. Camarri et al. [33] demonstrated an average distance of 659 m in 70 healthy subjects aged 55–75 years. Stronger correlations were observed by Roul et al. [34] when individuals with advanced heart failure walk a distance less than 300 m.

3.4.1 Experimental protocol and Methods

A sample of 16 healthy subjects with an average age of 55 years (SD = 17) was enrolled to perform a reduced version of the 6MWT, namely the 2-Minutes Walk

Test. The body sensor network used in this test was composed of 2 MuSe platforms provided with adjustable straps: one device was mounted on the chest of the subjects, the second one on the right shank.

Data were collected in two different sessions and locations, since the available subjects lived in two different cities. Due to the low space availability in one of the two locations, 9 subjects performed the test walking on a 10 meters-long corridor, whereas the remaining 7 subjects could perform the test according the practical guidelines (that is, on a 30 meter-long corridor): this fact was taken into account during the analysis in order to handle the different number of turns and the total walked distance. Two cones delimited the walking path: subjects were instructed to turn around them as closely as possible.

Before the start of each acquisition session, devices were calibrated in the middle of the corridor, at the chest level or at the shank level dependently on their mounting positions. Unfortunately, both the locations were characterized by magnetic interferences that became more intense when approaching the ground: the calibration of the sensor on the lower leg could not be performed, and therefore quaternions provided by that device were discarded. Finally, the calibrated chest device was used to measure the quaternion $q^0_{0,1}$, which describes the direction of the corridor with respect to the magnetic north (see section 3.2.1 for more information).

Both orientation (quaternions) and inertial raw data (accelerations and angular rates) were collected from the two devices at a frequency rate of 25 Hz, by means of a C#-based software developed ad hoc. This software associates each module of the network to the limb on which they are mounted, establishes and tests connections and finally starts simultaneous data log.

Test Parameters

The step detection, the turns identification and the consequent turning time estimation were performed by using methods described in section 3.3 on the basis of the available data. In particular:

- Turns were identified by using both the quaternion-based and the inertial data-based approaches.
- The inertial data-based approach was selected as the step detection procedure. Among the available data sources, the analysis was performed by using the shank vertical acceleration, although it could have been substituted with the others (either shank angular rate or trunk vertical acceleration). The quaternion-based approach was not usable due to unavailability of the leg orientation.

The outputs of these procedures were used to compute the test parameters. The following list reports the description of each parameter and the methods used for their evaluation.

Avg Turn Time The average turning time in seconds. It was computed as the mean value of each turn duration, which in turn depended on the method used for the turns detection: time interval between 45 and 135 degrees for the quaternion-based one, FWHM of the fit Gaussian curve for the gyroscope-based one.

Steps The number of steps walked in straight segments, the ends of which were determined on the basis of the related turns duration.

Distance The total walked distance in meters. By knowing the corridor length and the number of fully-walked straight segments (derived form the turn detection), the distance was first estimated as the multiplication of these two quantities. However, the resulted value lacked the meters between the last turning point and the position the subject ended the exercise. To determine such additional distance, the mean stride length was computed on the basis of the number of steps performed in the last fully-walked straight segment; then, such length was multiplied by the number of steps performed after the last turning point and added to the distance estimated before.

Avg Step Length The average step length in meters, computed as the total walked distance divided by the number of steps performed.

Avg/Max Step Frequency The average and the maximum step cadence in step/min. For each straight segment except for the last one, the cadence is determined by dividing the number of steps performed in the current segment by the time taken from the end of the previous turn to the start of the next one, in minutes. Then, the average and maximum values are computed.

Avg/Max Speed The average and the maximum walking speed in m/s. Similar to the *Avg Step Frequency* and the *Max Step Frequency* parameters, these features are determined by first dividing the corridor length by the time taken to walk a segment, and then computing the mean and the maximum values.

Corrected Distance The distance corrected to account for the different corridor lengths. Because of the shorter path, subjects who executed the test on the 10 meters-long corridor performed more turns in 2 minutes with respect to the others. The distance walked during the turn is negligible due to the close

movement around the cones, but the walking speed in their nearby decreases to allow the body to turn. Moreover, if the time required for the extra turns had been spent in walking the straight segment, the total distance would have been greater. For these reasons, the mean number of extra turns was computed for subjects who walked the shorter path, with respect to the mean number of turns performed by others; then, the distance was updated with a quantity given by the product of the average speed and the average turning time for each extra turn.

3.4.2 Results and Discussion

Results of the analysis are summarized in the tables below. In particular, Tables 3.1 and 3.2 report the 2MWT parameters estimated by detecting U-turns with both the quaternion-based approach and the inertial data-based one.

Table 3.1 – Parameters (part I) of the 2MWT, ordered by subject age. Results were obtained either by adopting the quaternion-based approach ("Q" columns) or the inertial-based approach ("Inrt" columns) for turns identification.

Subj. #	**Age**	**CL (m)**	**CS (#)**	**Distance (m)**		**Corr. Dist. (m)**		**TT_{avg} (s)**	
				Q	**Inrt**	**Q**	**Inrt**	**Q**	**Inrt**
14	28	30	7	233.23	234.19	233.23	234.19	0.42	0.79
15	29	30	6	205.83	207.43	205.83	207.43	0.72	1.21
16	35	30	7	238.06	239.03	238.06	239.03	0.80	1.29
13	39	30	7	222.86	224.55	222.86	224.55	0.77	1.34
9	41	10	13	132.50	134.62	142.09	152.84	0.98	1.69
8	42	10	19	192.00	193.00	198.98	206.09	0.50	0.88
7	47	10	14	148.46	150.00	155.92	165.84	0.70	1.35
6	61	10	15	155.71	158.46	165.00	180.06	0.80	1.64
12	62	30	5	180.00	182.43	180.00	182.43	0.83	1.36
5	67	10	14	149.29	150.00	157.29	168.47	0.74	1.52
10	68	30	6	184.19	184.29	184.19	184.29	0.82	1.42
1	69	10	14	143.13	144.00	150.49	162.21	0.71	1.57
2	70	10	14	141.43	143.08	151.19	161.40	0.94	1.61
4	71	10	14	147.50	150.00	154.98	167.20	0.70	1.45
3	73	10	13	133.33	134.29	141.38	151.02	0.83	1.58
11	76	30	5	170.00	171.75	170.00	171.75	0.82	1.50
M	55	-	-	173.59	175.07	178.22	184.92	0.76	1.39
SD	17	-	-	35.76	35.59	32.36	28.61	0.14	0.25

CL: Corridor Length; CS: Completed Segments; TT: Turn Time.

The definition of the turning time differs between the quaternion-based and the inertial data-based approach as discussed in section 3.3.2, thus the estimated average turn times resulted to be consistently different: 0.76 s in average for the

Table 3.2 – Parameters (part II) of the 2MWT, ordered by subject age. Results were obtained either by adopting the quaternion-based approach ("Q" columns) or the inertial-based approach ("Inrt" columns) for turns identification.

Subj. #	Steps (#)		SL_{avg} (m)		SF_{avg} (step/min)		SF_{max} (step/min)		$Speed_{avg}$ (m/s)		$Speed_{max}$ (m/s)	
	Q	**Inrt**	**Q**	**Inrt**	**Q**	**Inrt**	**Q**	**Inrt**	**Q**	**Inrt**	**Q**	**Inrt**
14	243	246	0.96	0.95	122.76	123.16	128.28	126.76	1.99	2.03	2.07	2.11
15	248	252	0.83	0.82	125.28	125.48	127.51	127.82	1.78	1.83	1.86	1.88
16	259	259	0.92	0.92	128.99	129.41	131.69	131.20	2.06	2.14	2.09	2.19
13	263	263	0.85	0.85	132.62	132.37	137.47	136.36	1.94	2.01	2.02	2.08
9	215	214	0.62	0.63	109.29	107.79	113.74	114.75	1.23	1.35	1.40	1.41
8	222	226	0.86	0.85	112.23	112.91	125.00	122.95	1.73	1.85	1.92	2.05
7	212	212	0.70	0.71	107.44	107.28	117.32	116.13	1.38	1.46	1.48	1.61
6	223	252	0.70	0.63	113.86	128.14	135.59	135.48	1.45	1.65	1.56	1.80
12	241	241	0.75	0.76	120.87	120.00	124.60	122.77	1.54	1.58	1.60	1.64
5	216	211	0.69	0.71	109.95	108.43	116.67	114.04	1.36	1.51	1.46	1.59
10	267	268	0.69	0.69	135.23	136.03	137.40	138.20	1.60	1.65	1.64	1.69
1	248	252	0.58	0.57	126.43	126.45	133.66	132.35	1.29	1.45	1.37	1.53
2	217	216	0.65	0.66	109.88	109.78	112.90	114.04	1.30	1.42	1.34	1.49
4	233	236	0.63	0.64	117.31	119.06	125.00	125.00	1.34	1.49	1.49	1.64
3	221	224	0.60	0.60	111.82	111.49	119.72	117.32	1.21	1.33	1.28	1.40
11	247	246	0.69	0.70	123.47	122.83	125.25	124.87	1.46	1.50	1.53	1.56
M	236	238	0.73	0.73	119.21	120.04	125.74	125.00	1.54	1.64	1.63	1.73
SD	18	19	0.12	0.12	8.94	9.41	8.08	8.20	0.28	0.26	0.27	0.26

SL: Step Length; SF: Step Frequency.

former approach, 1.39 s in average for the latter. Moreover, since the ends of each turn (defined by the turn time) represent the end of the previous straight segment and the start of the next one, some steps that in one method were considered to be performed during the turn, in the other method they were considered as part of the straight segment: therefore, all the other parameters were affected by the selected approach too. Nevertheless, Pearson's linear correlation coefficients between same parameters computed with different methods (e.g. quaternion-based *Avg Speed* and inertial data-based *Avg Speed*) highlighted a strong correlation for all pairs of parameters ($r > 0.86$, p-values < 0.0002).

In both cases, the estimated 2-Minutes Walked Distances (2MWDs) agreed with what reported in literature: in fact, they ranged from 150 m to 240 m, corresponding approximately to a range of 450–720 m for an equivalent 6MWD if a linear relation between time and walked distance were considered. In order to check the existence of a possible relation with age, the Pearson's linear correlation coefficients r between parameters of both methods and age were computed. Moreover, two-tailed significance tests were performed under the null hypothesis of correlation being equal to zero, against the alternative hypothesis of non-zero correlation. The significance

level α was set to 0.05. Coefficients and related p-values are reported in Table 3.3.

Table 3.3 – Pearson's linear correlation coefficients between 6MWT parameters and age, for both quaternion-based and inertial data-based approaches. Values in brackets represent the p-values obtained from the significance tests, under the null hypothesis of correlation being equal to zero ($\alpha = 0.05$).

Parameter	**Quaternion-based r (p-value)**	**Inertial-based r (p-value)**
CorrDistance	-0.72 (0.002)	-0.74 (0.001)
Steps	-0.17 (0.536)	-0.15 (0.570)
AvgStepLen	-0.78 (< 0.001)	-0.77 (< 0.001)
AvgStepFreq	-0.16 (0.556)	-0.12 (0.656)
MaxStepFreq	-0.14 (0.609)	-0.14 (0.607)
AvgSpeed	-0.72 (0.002)	-0.72 (0.002)
MaxSpeed	-0.77 (< 0.001)	-0.72 (0.002)
AvgTurnTime	0.42 (0.108)	0.65 (0.007)

Results indicated that the walked distance had a statistically significant negative correlation with age, meaning that the older the subject, the shorter the walked path, as one can expect among healthy subjects. Also, the average step length as well the average and maximum speed resulted to be negatively correlated with the age, thus reinforcing the natural idea of a decrease of the overall physical performance associated with aging. In a similar way, the average turn time seems to slightly increase with age, indicating reduced velocities while older subjects were turning around the cones: this may indicate that those subjects proceeded with higher caution, maybe due to a self-consciousness of their own instability. Regarding this parameter, the only statistically significant version was the one determined by using the inertial data-based approach.

Finally, no correlation was obtained neither between age and steps number, nor between age and step frequencies (average and maximum), as expected. As a matter of fact, these results correctly indicate that the number of steps taken is independent of age (and as such, the step frequency too): in fact, although an elder subject may perform shorter strides due to their greater instability and their reduced motor skills, it is likely that the number of steps will be similar to those taken by a younger subject.

To conclude, in this section results of preliminary analysis about the Six-Minute Walk Test were reported and discussed. A sample of healthy subjects of different ages were monitored during the execution of a shorter version of the test, by means of two wearable inertial modules mounted on the chest and on the lower leg. The overall walked distances, i.e. the primary outcome of the test, agreed with literature

results concerning this category of subjects. Two different definitions (and as such, two different methods) of turn detection were adopted, thus affecting the estimation of other parameters directly depending on such feature: in order to establish the most appropriate definition, results of the two analyses need to be assessed through the usage of a visual motion tracking system in the next future. Moreover, pathological subjects need to be included in the sample to validate the used analysis procedures and to check the compliance of the so-obtained results with literature ones.

3.5 Extended Timed Up and Go Test

The Timed-Up and Go (also known as TUG) test is a simple exercise able to assess mobility, balance and fall risk of an individual. This test is easy to run and it is acknowledged as a reference practice in the rehabilitation field for several diseases, like Alzheimer's disease or Parkinson's disease. The exercise consists of the following steps (refer to Figure 3.17 for phases numbers):

- The patient starts from a seated position, with the spine leaning on the seatback of a chair;
- When the operator gives the start signal, the patient stands up (1), walks three meters (2), turns around (3) and walks back to the chair (4);
- The patient sits down (5).

The equipment required to carry out this test consists only in an armchair, a stopwatch and a marker to locate the turning point. A variation of the exercise exists, namely the Expanded TUG (ETUG), which differs from the standard test in the following elements:

- The patient has to cover 10 m;
- The turning trajectory has a larger radius;
- Intermediate times are registered in order to consider the time taken by elementary actions (standing, walk time, turn duration, sitting).

The TUG test is used frequently in the elderly population, as it is easy to administer and can generally be completed by most older adults. Moreover, in the last decade several studies revealed that results from this test correlates with many diseases scales and progression, such as in Parkinson's Disease [35, 36] and

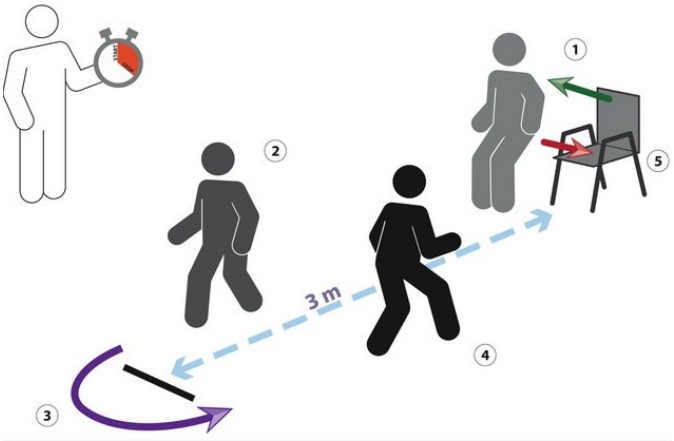


Figure 3.17 – Principle of the Timed-Up and Go (TUG) test.
Source: Marie-Cécile Nierat et al, "When Breathing Interferes with Cognition: Experimental Inspiratory Loading Alters Timed Up-and-Go Test in Normal Humans", Figure 1.

Multiple Sclerosis [37, 38]. The gold standard for TUG test are motion capture systems (e.g. ultrasound system, optical system, and/or force plates), but recent studies also support the use of inertial devices to assess objectively the movement of people with PD by demonstrating the validity of inertial platforms in comparison to motion capture systems [39, 40]. The use of such devices would in turn support the monitoring of subjects in real-life conditions and the spread of telemedicine and home rehabilitation: the test could be prescribed by clinicians and performed by subjects themselves at their own home. In order to achieve such goal, monitoring systems must take into account several aspects:

- The use of a large number of sensors typically makes it possible to capture a wide range of movement patterns and features adopted by people with Parkinson's Disease, but, to address acceptability and usability requirements, a reduced number of sensors is more appropriate;
- In order to account for non-idealities of an unsupervised environment, some sensors such as magnetometers require a precise calibration that would need to be performed by the user;
- Large dimension and weight of wearable devices should be avoided because they are not comfortable;
- It is crucial that a feedback to the user about results of the gait analysis is given immediately after execution of the test, for instance through the development of semi-automated operations. The algorithms should automatically detect all the test features and show differences between healthy subject's performances

and the current ones, even if large variations in results are common due to different walking styles [41].

A huge amount of work has been done in the recent decade to *sensorize* the TUG test and to automate the feature extraction. Among the several publications, Salarian et al. [42] developed an innovative system called iTUG: it relies on inertial platforms to monitor the execution of the TUG test and to determine many different metrics which can not be estimated by means of the standard evaluation procedure. Although the publication is not so recent (2010), their methods are still used in today's literature. With respect to such analysis algorithms, the author of this thesis wants to provide alternative methods to compute TUG's features. However, it won't be possible to compare the performances of Salarian's procedures with those of the methods presented hereby, as long as data from patients are not available.

3.5.1 Experimental protocol and Methods

A body sensor network composed of 5 MuSe nodes was used to monitor the execution of the ETUG test performed by 9 healthy subjects with an average age of 60 years (SD = 13). Devices were provided with elastic adjustable straps and mounted on the shanks, the wrists and the chest of each subject. Before the start of the acquisition session, device were calibrated in the middle of the walking path, at a height depending on their mounting positions. Moreover, the calibrated chest device was used to measure the quaternion $q^0_{0,1}$, which describes the direction of the exercise with respect to the magnetic north (see section 3.2.1 for more information).

Both orientation (quaternions) and inertial raw data (accelerations and angular rates) were collected from the MuSe devices at a frequency rate of 25 Hz, by means of the same C#-based software developed for the 6MWT: it handles the connections with the devices, allowing the user to start and stop the data log of the devices simultaneously. Different procedures based either on orientation or raw data were adopted, in order to provide results obtainable feature extraction algorithms that might be used according to the specific use cases (depending on the environment type and the number of nodes available).

Execution Time and Reaction Time

The most important parameter to consider when assessing the performance results of the TUG test is the *execution time*, that is, the time elapsed from the start command given by the operator and the instant the subject sits down at the end of the exercise: high values of this parameter are correlated with gait disabilities and risk of falls.

While it is possible to make sure that the start instant coincides with the start of the acquisition (by coordinating the vocal command with the press of the start button), this is not always feasible for the end of the test. According to the guidelines, the timing should be stopped when the subject is seated again correctly with their back resting on the back of the chair. However, some subjects may have difficulties in seating in a correct position, as the case of patients affected by dyskinesia. For this reason, the definition of execution time might be slightly different from user to user: for instance, in some occasions the timing is stopped as soon as the subject sits down the chair; in others, when the back of the subject is perpendicular to the ground.

The execution time was estimated by using two similar approaches, different only because of the used data source: the choice of implementing two algorithms was made in order to provide feature extraction algorithms that might be used both in supervised and unsupervised test environments. Both methods are based on the following aspect: in order to perform a sit-to-stand (and stand-to-sit) movement in a natural way, it is necessary to move the projection of the body center of mass between the feet, that is, to lean forward. This event can be detected by using both quaternions and raw inertial data collected by the device mounted on the trunk.

Quaternion-based After the alignment of reference systems, the pitch angle is computed by using the provided quaternions. A moving average filter with 3 samples-span is used to smooth the signal. Finally, the peak detection algorithm described in section 3.3.1 identifies the highest peaks, corresponding to the sit-to-stand event E_1 and the stand-to-sit event E_2.

Inertial data-based Similarly to the quaternion-based approach, events E_1 and E_2 are identified as peaks in the acceleration signal referred to the subject's reference system. When leaning forward, part of the gravity acceleration can be measured on the x axis. This signal is analyzed by means of the very same procedure described in section 3.3.1 for the inertial data-based step detection: mean removal; integration and CWT-based differentiation; and peak detection.

For both procedures, the peak detection algorithm is set to find the two highest peaks with a minimum distance of 2 seconds among them: as a result, the events corresponding to instants where the maximum incline occurs are found (see Figure 3.18). A first easy computation of the execution time can then be achieved by measuring the time interval between the start of the exercise and the last event detected. Another parameter named *reaction time* can be easily estimated from

the results of the described procedures: it is defined as the time required by the subject to start moving after the start vocal command given by the operator, and can be obtained by measuring the time interval between the start of the acquisition and the first detected event.

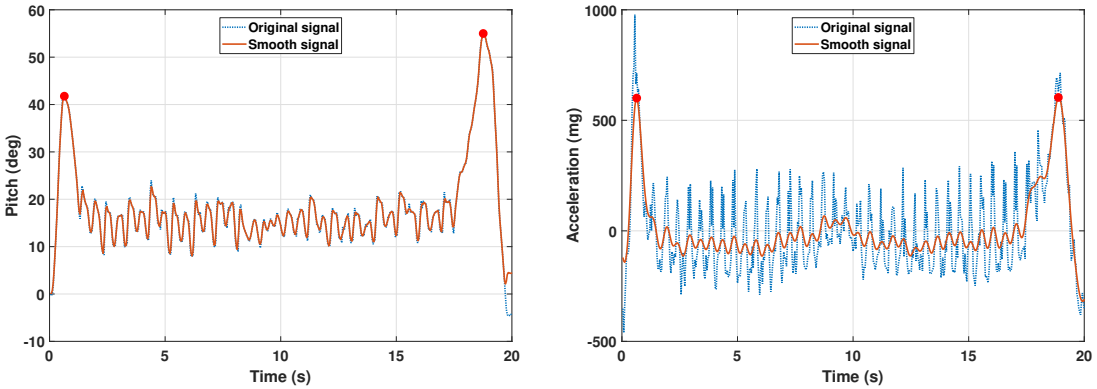


Figure 3.18 – Results of the quaternion-based (left) and inertial data-based (right) sit-to-stand and stand-to-sit events detection. The red points identify their locations.

The stand-to-sit event E_2 identified by the described procedures, however, does not coincide with the instant at which timing should be stopped according to the guidelines, since it lacks the final sitting movement (when the subject lays their back on the back of the chair). In order to account for this extra time, an alternative can be found by fitting the signal — either smoothed pitch angle or smoothed acceleration's x component — with two Gaussian curves (one for each event) and considering their width, σ_1 and σ_2: the execution time can be approximated by the time between the first sample and $E_2 + 2 \cdot \sigma_2$, whereas the reaction time can be defined as the time between the first sample and $E_1 - 2 \cdot \sigma_1$ (see Figure 3.19).

Swing Identification

When used to monitor people with Parkinson's Disease performing the ETUG test, a standard BSN composed by 5 nodes (like the one considered in this chapter) can be used to estimate the mean oscillation amplitude of limbs. This parameter can encode a significant information: in fact, since PD typically affects only one side of the body, the severity of the disease can be associated with asymmetries in arms oscillation amplitudes.

Once again, two methods have been developed for the estimation of limb's swing amplitude, one based on quaternions and one based on raw inertial data. While the former simply relies on the pitch angles computed from the estimated quaternions, the latter involves the integration of the angular velocities to determine the angular

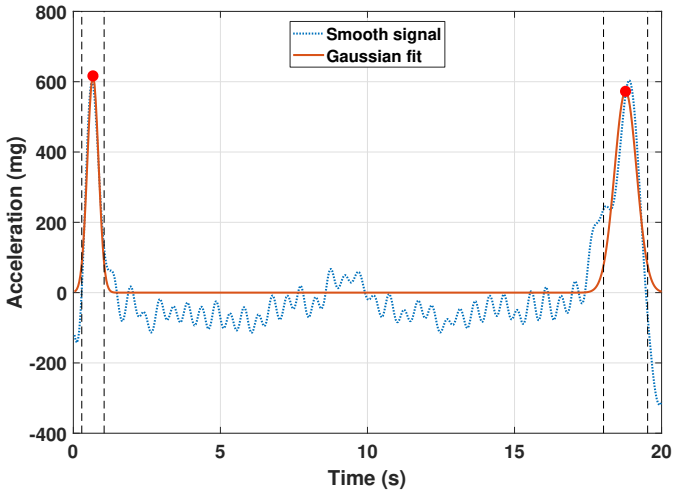


Figure 3.19 – Gaussian curves fitted to the acceleration signal at the sit-to-stand and the stand-to-sit events. The red points identify their locations, while the vertical dashed lines represent the $E_x \pm 2 \cdot \sigma_x$ locations.

displacement of the limb. In this case, when dealing with legs oscillations the relevant axis is the one providing for data with the highest standard deviation, as in the identification of steps based on data from the gyroscope (see section 3.3.1); instead, when dealing with arms oscillations the relevant axis is always the subject's y axis, because the mounting position of the wrist devices is known and can not be different as the leg case. Numerical integration is performed by means of the trapezoidal rule.

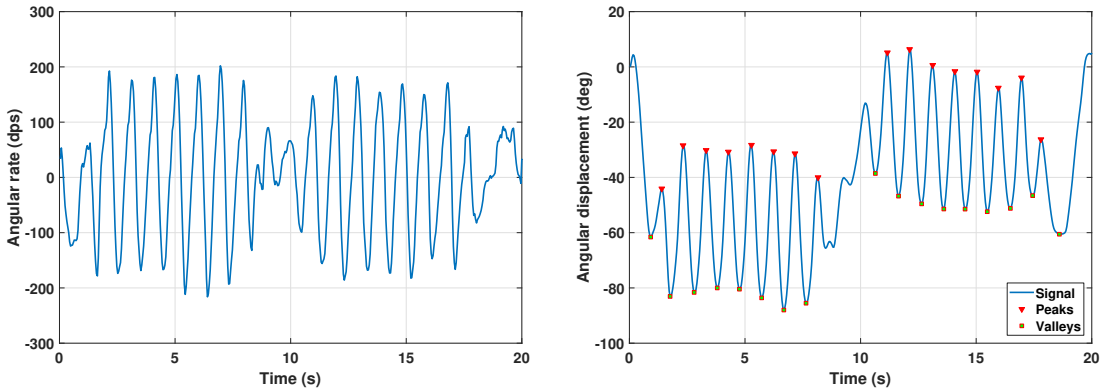


Figure 3.20 – Lateral component of the left arm's angular velocity on the left, and results of integration and peak detection on the right. The initial displacement drop is due to the arm starting position (laid on the armrest of the chair).

Given the limb signal — either pitch angle or angular displacement — both algorithms then look for peaks in the signal and its reverse with a minimum prominence of 15 degrees, and concatenate them alternately to define the signal's

peaks and valleys.

Test Parameters

The step detection and the turns identification were performed by using methods described in section 3.3. In particular:

- Turns were identified by using both the quaternion-based and the inertial data-based approaches.
- Steps were identified by using both the quaternion-based and the inertial data-based approaches. Moreover, all the three possible data sources (shank angular rate, shank vertical acceleration and trunk vertical acceleration) were used for the inertial data-based approach, in order to compare the results.

The outputs of these procedures were used to compute the test parameters. The following list reports the description of each parameter and the methods used for their evaluation.

Execution Time - Reaction Time The execution time and the reaction time in seconds, computed as described in the previous section. Regarding the approach that make use of inertial data, both the peaks-based and the fit-based procedures were adopted.

Turn Time The turning time in seconds. Its value depended on the method used for the turn detection: time interval between 45 and 135 degrees for the quaternion-based one, FWHM of the fit Gaussian curve for the gyroscope-based one.

Steps The number of steps walked in total (*tot*) or in straight segments only (*str*). The ends of such segments were determined on the basis of the turns duration.

Step Frequency The average step cadence in step/min, either relative to the total path (*tot*) or to the straight segments only (*str*). The former is determined by dividing the total number of steps by the time taken from the end of the sit-to-stand event and the start of the stand-to-sit one (in minutes), whereas the latter excludes from this period the time taken to perform the turn.

Step Length The average step length in meters, relative to the gait performed in straight segments only. It is defined as the distance covered from a left

foot stance to the next right foot stance (or vice versa). It is computed as the total path length (20 m) divided by the number of steps taken in straight segments.

Stride Length The average foot stride length in meters, relative to the gait performed in straight segments only. It is defined as the distance covered from a left (right) foot stance to the next left (right) foot stance. This parameter is computable only when steps are identified by using quaternions or angular rates, since any other data source wouldn't allow to distinguish a left from a right step.

Speed The average walking speed in m/s, relative to the straight segments only. It is determined by dividing the total path length (20 m) by the time taken from the end of the sit-to-stand event and the start of the stand-to-sit one (in minutes), excluding the time taken to perform the turn.

Swing The mean absolute difference between consecutive peaks and valleys of the limb's pitch angle (quaternion-based) or angular displacement (inertial data-based). Only peaks and valleys occurred when walking the straight segments were considered. LA, RA, LL and RL stand for left arm, right arm, left leg and right leg respectively.

3.5.2 Results and Discussion

The following tables report the results of the data analysis, obtained by using methods described in the previous sections. In particular, Table 3.4 shows the temporal parameters — i.e. the execution time, the reaction time and the turn time — estimated by means of quaternion-based and inertial data-based approaches. Moreover, the latter was performed considering both the events detection methods: the one based on peaks in the acceleration signal, and the one based on the fitted Gaussian curve.

Regarding the algorithms using the peak-based events detection (i.e. "Quat" and "Inrt$_{pks}$"), the maximum differences among average values are 0.16 s for the execution time and 0.04 s for the reaction time. These values slightly increase up to 0.84 s and 0.44 s respectively when considering the algorithm based on the fitted curve (i.e. "Inrt$_{fit}$"), due to the different definition of those parameters: nevertheless, results are still comparable. In a similar way, the difference between quaternion-based and inertial data-based estimates of the turn time (0.57 s) is due to the specific definition of "turn" adopted by each method. Unfortunately, subjects were not timed manually during the execution of the test, thus it was not possible

Table 3.4 – Temporal parameters of the ETUG test, computed by using both quaternion-based approaches ("Q") and inertial data-based approaches ("Inrt").

Subj. #	Age	Execution Time (s)			Reaction Time (s)			Turn Time (s)	
		Q	**Inrt$_{pks}$**	**Inrt$_{fit}$**	**Q**	**Inrt$_{pks}$**	**Inrt$_{fit}$**	**Q**	**Inrt**
9	41	16.76	16.84	17.32	1.12	1.12	0.76	0.92	1.29
8	42	21.28	21.64	22.20	1.24	1.24	0.80	0.88	1.58
7	47	17.36	17.44	18.00	1.32	1.32	0.92	0.84	1.14
6	61	19.40	19.68	20.28	0.48	0.44	0.04	1.00	1.93
5	67	18.32	18.48	19.20	1.40	1.36	1.00	1.16	1.62
1	69	18.76	18.88	19.52	0.64	0.64	0.32	1.00	1.79
2	70	17.60	17.72	18.36	1.12	1.00	0.48	1.40	1.78
4	71	20.00	20.04	21.00	1.68	1.56	1.16	1.00	1.81
3	73	21.48	21.76	22.68	0.92	0.84	0.44	1.12	1.55
M	60	19.00	19.16	19.84	1.10	1.06	0.66	1.04	1.61
SD	13	1.68	1.76	1.86	0.37	0.36	0.36	0.17	0.26

pks: events refer to signal peaks. *fit*: events refer to the fitted Gaussian curve.

to evaluate accuracies of the estimated temporal parameters: further acquisition sessions are needed to investigate which of the presented methods has a better performance in estimating the real parameters.

The computation of Pearson's linear correlation coefficients revealed that in the current dataset there was no correlation neither between age and execution time, nor between age and reaction time. Instead, weak correlations were observed between age and turning times estimated by the two methods, the lowest of which being 0.67 (p-value = 0.048). In any case, a larger sample will be needed in the next future to assess current correlations.

Tables 3.5 to 3.7 report summaries of the parameters related to gait. While steps in the first table were determined by using the quaternion-based procedure, the second and the third tables present results of the inertial data-based procedure for all of the three available data sources: angular rates of legs (AR_{legs}), vertical acceleration of legs (VA_{legs}), and vertical acceleration of the trunk (VA_{trunk}). Moreover, for each table, data were analyzed according to the sit-to-stand and stand-to-sit events detected with the three methods previously described: peaks in the pitch angle of the trunk, peaks in the trunk acceleration, and specific points of the fitted Gaussian curves.

Considering the same event identification method, the step count maximum error was 3 for methods based on peaks (Tables 3.5 and 3.6) and 1 for the fit-based method (Table 3.7). The lower error (and thus, the better performance) of the fit-based method with respect to peak-based ones may be ascribed to the fact that

Table 3.5 – Mean values and standard deviations (in brackets) of quaternion-based gait parameters. The stand-to-sit and the sit-to-stand events were computed as peaks among the trunk's pitch angles.

Parameter	**Quat**
Steps_{tot}	32 (4)
Steps_{str}	30 (4)
StepFreq_{tot}	118.65 (20.07)
StepFreq_{str}	117.98 (20.44)
StepLen	0.68 (0.08)
StrideLen_{L}	1.36 (0.16)
StrideLen_{R}	1.35 (0.18)
Speed	1.31 (0.13)

Table 3.6 – Mean values and standard deviations (in brackets) of inertial data-based gait parameters. The stand-to-sit and the sit-to-stand events are related to acceleration peaks.

Parameter	$\mathbf{AR}_{\boldsymbol{legs}}$	$\mathbf{VA}_{\boldsymbol{legs}}$	$\mathbf{VA}_{\boldsymbol{trunk}}$
Steps_{tot}	34 (3)	32 (4)	31 (3)
Steps_{str}	31 (3)	29 (4)	28 (3)
StepFreq_{tot}	112.39 (4.89)	106.13 (8.81)	103.10 (8.20)
StepFreq_{str}	112.08 (5.32)	106.50 (10.30)	101.89 (8.74)
StepLen	0.66 (0.07)	0.69 (0.09)	0.72 (0.07)
StrideLen_{L}	1.40 (0.14)	-	-
StrideLen_{R}	1.24 (0.15)	-	-
Speed	1.22 (0.12)	1.22 (0.12)	1.22 (0.12)

AR: Angular Rate. VA: Vertical Acceleration.

Table 3.7 – Mean values and standard deviations (in parenthesis) of inertial data-based gait parameters. The stand-to-sit and the sit-to-stand events were computed from the fitted Gaussian curves.

Parameter	$\mathbf{AR}_{\boldsymbol{legs}}$	$\mathbf{VA}_{\boldsymbol{legs}}$	$\mathbf{VA}_{\boldsymbol{trunk}}$
Steps_{tot}	31 (3)	30 (4)	30 (3)
Steps_{str}	28 (3)	27 (3)	27 (2)
StepFreq_{tot}	113.56 (4.04)	106.99 (8.78)	110.16 (7.84)
StepFreq_{str}	113.37 (4.94)	107.46 (10.26)	109.62 (8.78)
StepLen	0.71 (0.07)	0.76 (0.10)	0.74 (0.06)
StrideLen_{L}	1.50 (0.13)	-	-
StrideLen_{R}	1.37 (0.16)	-	-
Speed	1.35 (0.13)	1.35 (0.13)	1.35 (0.13)

AR: Angular Rate. VA: Vertical Acceleration.

the latter consider movements immediately after the sit-to-stand event (and before the stand-to-sit event) as gait data, although they are instead associated to the end (or the start) of the sitting phase: in some cases, such movements generated signals that were erroneously considered as steps. Differences in the other parameters depend only on the different number of steps detected. Thus, the specific definition of the sit events directly affects the number of steps identified, which in turn varies the related parameters such as step cadence and step length. No significant correlation was obtained among age and parameters ($r < 0.6$, p-value > 0.07).

Finally, Table 3.8 summarizes the mean oscillation amplitude of limbs computed by methods described previously. Even in this case, the detection of valid oscillations depended on the sit-to-stand and the stand-to-sit events identification.

Table 3.8 – Mean values and standard deviations (in parenthesis) of quaternion-based and angular rate-based limbs swings. One subject was considered as an outlier, thus it was removed.

Parameter	**Quat**	**AR$_{legs}$**	
		Inrt$_{pks}$	**Inrt$_{fit}$**
Swing$_{LA}$	40.71 (8.63)	47.31 (11.35)	47.41 (11.41)
Swing$_{RA}$	33.56 (8.08)	40.32 (9.83)	40.43 (10.18)
Swing$_{LL}$	66.84 (13.04)	66.80 (10.53)	67.58 (10.70)
Swing$_{RL}$	66.90 (13.00)	68.35 (5.05)	68.78 (4.78)

AR: Angular Rate.

As far as the legs are concerned, the mean swing amplitudes were very similar among different methods, with a maximum difference of less than 2 degrees; in addition, the estimation of the left leg's oscillation amplitudes are comparable to the right leg's ones. On the contrary, arms swings were different both between quaternion-based and inertial data-based methods, and between left and right amplitudes. Two hypothesis can explain these differences:

1. Some subjects in the sample set are characterized by skewness in arm movements during the gait.

2. Since the arm swing is composed of angular rates on all of the three axis (due to the higher freedom of movement with respect to legs), the analysis of one component only may lead to a much smaller amplitude the more the subject rotates the arm around axes of non-interest for the algorithm.

Both hypothesis require a comparison with results from a visual motion tracking system in order to be confirmed.

To summarize, in this section methods to extract information about the execution of the ETUG test were discussed. Several procedures were developed on the basis of:

- The available nodes of the BSN (legs, arms, trunk);
- The available on-board sensors (accelerometer, gyroscope) and processing capabilities (quaternion computation);
- Whether the test is performed in a supervised environment (quaternions available thanks to the calibration of the magnetometer) or not (quaternions not available, inertial data only).

Generally speaking, results are slightly different depending on available nodes and sensors, as well as on the definitions adopted for events and parameters. Results of the presented methods need to be compared to those obtainable from a visual motion tracking system in the next future, in order to determine the estimation error for each procedure and thus to assess their accuracy. In this way, clinicians could configure their own BSN on the basis of the required accuracy, the level of burden that the network represents for a pathological subject, and the environment in which the system is going to be used — either in clinic or at home.

A larger sample is needed to confirm (or refuse) the correlation results; moreover, the enrollment should be enlarged to include patients affected by Parkinson's disease, in order (1) to determine which parameter is more sensitive to the disease scales, (2) to observe whether or not correlations among test parameters and disability level provided by literature will agree with those obtained by using the described methods, and (3) to compare the results of the described procedures with the state-of-the-art methods. Regarding the BSN, further investigations will be conducted to assess if changing the mounting position of the trunk node to the lower back makes it possible to extract more information about steps (for instance, to distinguish left steps from right ones).

3.6 Balance Evaluation

Unstable balance, falls, and loss of independence in the older population are a growing problem in the developed countries. The safety of walking and balance are indispensable requirements to prevent falls and loss of independence. The afferent visual, vestibular and proprioceptive systems represent the base of safe mobility as they offer the subject vital information for postural control: they determine, respectively, three behavioral strategies. Using posturography or stabilometry, it is

possible to distinguish and to evaluate the functional levels of the strategy used by each subject [14]. A posturographic system should be able to analyze the postural oscillations and the strategy used to maintain control of an upright stance, to offer the possibility to analyze several parameters and to provide comparable data (see Figure 3.21).

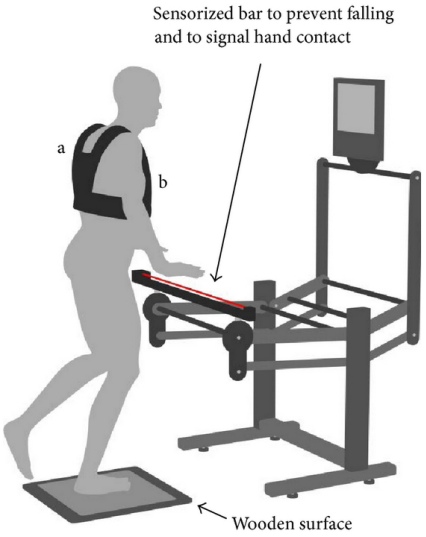


Figure 3.21 – Example of a posturographic station. The red line represents the infrared ray of a sensorized bar. Vest (a) to support the "postural reader", an accelerometer unit (b), in sternal position.
Source: [14], Figure 1.

The balance is evaluated considering the average value and the standard deviation of the center of gravity in different conditions. Among several balance tests, the static double-leg stance (*stabilometric*) and the static single-leg stance (*static propaedeutic*) tests provide a quantification of the postural control [43]. The former test assesses the stability in bipodalic stance during different sensory conditions (open eyes, closed eyes, hyperextended head and on rubber). The comparison among the data of the different tests underlines the presence of visual dependence, visual effect paradox and vestibular effect paradox. On the other hand, the static propaedeutic test has the goal to evaluate the risk of fall and if the walking stability of the subject depends on the visual information. Thanks to the comparison of a simple static test performed in monopodalic stance in two different sensory conditions (eyes open and eyes closed) it is possible to assess:

- The precaution strategy, that intervenes when the subject is not able to manage the static monopodalic stance without the help of an external support;
- The visual strategy that allows a quantification of the visual dependence that is how much the stability of the subject depends on the visual information;

- The intervention of the proprioceptive strategy that is the base of the postural control;
- The intervention of the vestibular strategy (ability to manage the emergency situations).

3.6.1 Experimental protocol and Methods

A cohort of 16 healthy people with an average age of 55 years (SD = 17) was enrolled to monitor their balance during different sensory conditions. The measured subjects were instructed before the execution of the test and all of them followed the protocol hereafter explained. Each subject wore a MuSe device mounted on the center of their chest by means of an elastic strap. Such mounting location was chosen due to the fact that the absolute position of the center of mass mainly affects the chest orientation: in fact, movements of its upper part help in counteracting body imbalances. After the start given by an operator, they were asked to stand still in double-leg stance with eyes open for 20 seconds, then after hearing an acoustic notification they had to close their eyes for other 20 seconds. Raw data were collected at a frequency rate of 50 Hz by using a C#-based software developed ad hoc, establishing the Bluetooth connection with the device and starting and stopping the data streaming.

When dealing with balance, the chest's movements of interest are those lying in the sagittal (or longitudinal) plane and in the coronal (or frontal) plane. These movements are known in anatomy as flexion/extension (movements within the sagittal plane) and adduction/abduction (movements within the coronal plane), and can be easily associated with pitch and roll rotations respectively. For a clear view of the association between anatomical planes and Euler angles, see Figure 3.22. Since the most informative contribution comes from the variation of the subject's vertical axis' orientation, the accelerometer seemed to be the most valuable sensor.

In order to easily extract information from the collected accelerations, data were first converted to the reference system of the subject wearing the MuSe device. Because the platform was mounted in the center of the subject's chest, there was no yaw offset between the device's reference system and the subject's one, but only a pitch offset of about 90 degrees and a small roll offset (the latter probably due to some imperfections in the elastic strap fabric). For this reason, the vertical alignment procedure described in section 3.2.2 was an effective way to change the accelerations reference system: it was performed based on an acceleration sample collected in stationary conditions, right before the execution of each test (see its

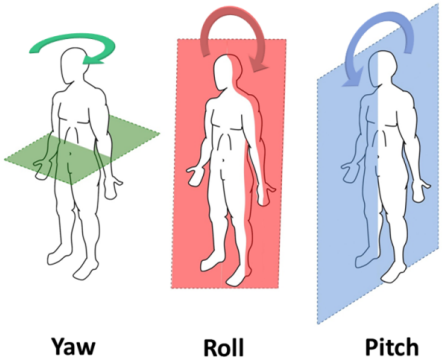


Figure 3.22 – Anatomical planes and Euler angles associated with them. From left to right, the transverse plane, the coronal plane and the sagittal plane are depicted.

effect in Figure 3.4).

Thanks to the vertical alignment, the measured accelerations were re-computed with respect to the subject's reference system: any chest's rotation within the sagittal plane would then have been represented by an increase of the acceleration's x component (and a consequent decrease of the z component), whereas any rotation within the coronal plane would then have resulted in an increase of the acceleration's y component (same as before for the z one). The so-called "aligned accelerations" were then used to determine the roll angles and the pitch angles — namely α and β — of the chest's rotations by means of eq. (3.9) and eq. (3.8) respectively.

The stability parameters were computed in accordance to the procedure described by Riva in [14]. These parameters are independent of the specific balance test (whether it is the stabilometric test or the static propaedeutic test) and can be easily used to assess the subject's stability. Firstly, the angular deviation of center of mass (COM) with respect to the vertical axis was determined by averaging all the roll and pitch angles:

$$COM = \left(\frac{1}{N} \sum_{i=1}^{N} \alpha_i, \frac{1}{N} \sum_{i=1}^{N} \beta_i \right). \tag{3.20}$$

A first measure of the subject's postural instability (PI) was given by the average absolute deviation of the measured COM components from the mean roll angle and the mean pitch angle:

$$\begin{aligned} PI_x &= \frac{1}{N} \sum_{i=1}^{N} |\alpha_i - \bar{\alpha}|, \\ PI_y &= \frac{1}{N} \sum_{i=1}^{N} \left|\beta_i - \bar{\beta}\right|. \end{aligned} \tag{3.21}$$

In order to get a combined measure of the subject's instability, the absolute instant

deviations AID were computed by means of the equation

$$AID_i = \sqrt{(\alpha_i - \bar{\alpha})^2 + \left(\beta_i - \bar{\beta}\right)^2}, \tag{3.22}$$

and averaged to obtain the average radius of the postural cone of instability PI_{xy}:

$$PI_{xy} = \frac{1}{N} \sum_{i=1}^{N} AID_i. \tag{3.23}$$

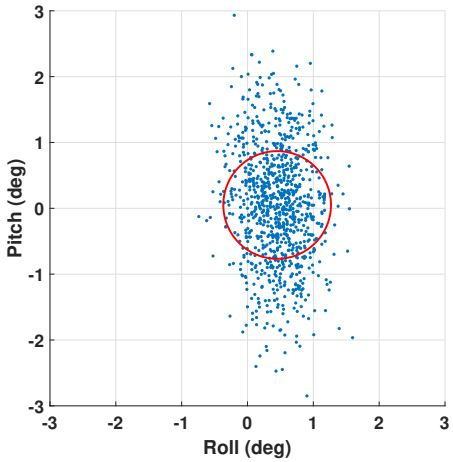


Figure 3.23 – Top view of the chest orientation during the execution of the stabilometric test. The red circle represents the top view of the postural cone, centered in the subject's COM.

In [14], Riva introduced a metric called *sample performance value* (*spv*) to consider the contribution of each sample to stability performances. This metric takes into account the value of the i-th sample as well as an attribute indicating whether or not a hand contact occurred in that instant: this last information was obtained by using an infrared sensorized bar in front of the subject, to prevent falling and to detect hand contacts. Since the adopted acquisition system did not integrate such a bar, and because no hand contact actually occurred, each *spv* was calculated only by means of the equation for no hand contact:

$$spv_i = \begin{cases} 10^{-\frac{AID_i}{25°}} & AID_i < 50° \\ 0.01 & AID_i \geq 50°. \end{cases} \tag{3.24}$$

From the previous formula it is possible to observe that the *spv* ranges from 0.01 to 1 depending on the dimension of the postural cone: the wider the cone, the smaller the *spv*. By averaging the computed *spv*, the stability index SI was determined, representing the relation (in %) between the *spv* sum of the test session and a

virtually perfect one:

$$SI = \frac{100}{N} \sum_{i=1}^{N} spv_i. \tag{3.25}$$

Finally, the visual dependency VD was evaluated by subtracting the stability index with eyes closed condition from the stability index with eyes open condition. This parameter gives a quantification of how much the subject relies on visual information to correct the balance.

3.6.2 Results and Discussion

Table 3.9 reports the estimated balance parameters during the execution of the stabilometric, double-leg stance test. As can be noticed, all of the subjects performed well both in eyes open and in eyes closed conditions, with average stability indexes of 95.03 % and 94.91 % respectively.

Table 3.9 – Balance parameters related to the double-leg stance test.

N.	**Age**	**Eyes Open**			**Eyes Closed**			**VD**(%)
		COM(°)	$\mathbf{PI_{xy}}$(°)	**SI**(%)	**COM**(°)	$\mathbf{PI_{xy}}$(°)	**SI**(%)	
1	28	$(-0.52; 1.03)$	0.35	96.86	$(-0.33; 0.00)$	0.28	97.44	-0.58
2	29	$(0.03; 0.82)$	0.54	95.18	$(0.73; 1.01)$	0.33	96.99	-1.81
3	35	$(-0.21; -0.70)$	0.22	98.03	$(-0.97; -0.94)$	0.19	98.31	-0.28
4	39	$(0.49; -1.61)$	0.31	97.20	$(3.16; -3.56)$	0.24	97.80	-0.60
5	41	$(-0.09; -0.45)$	0.41	96.32	$(0.05; -0.21)$	0.46	95.85	0.46
6	42	$(0.28; 0.15)$	0.35	96.86	$(-0.59; -0.10)$	0.36	96.75	0.10
7	47	$(0.16; 0.25)$	0.58	94.83	$(-0.54; 0.35)$	0.70	93.79	1.04
8	61	$(-0.20; -1.25)$	0.99	91.39	$(0.00; -0.48)$	1.01	91.25	0.13
9	62	$(0.08; -1.00)$	0.20	98.18	$(0.09; 1.10)$	0.35	96.82	1.36
10	67	$(-0.02; -2.55)$	0.86	92.50	$(0.42; 1.03)$	0.89	92.26	0.24
11	68	$(-0.84; 1.55)$	0.40	96.43	$(-1.47; 0.23)$	0.32	97.14	-0.71
12	69	$(0.45; 0.05)$	0.82	92.85	$(0.30; 0.17)$	0.87	92.39	0.45
13	70	$(0.21; 0.10)$	1.03	91.08	$(0.48; 0.06)$	1.26	89.26	1.82
14	71	$(-0.55; -2.52)$	0.86	92.55	$(-1.09; -0.52)$	0.85	92.53	0.02
15	73	$(-0.26; 0.65)$	0.75	93.37	$(0.22; 1.16)$	0.72	93.63	-0.26
16	76	$(0.32; 0.66)$	0.53	96.84	$(-0.36; 2.20)$	0.42	96.26	0.58

In order to observe greater differences, a subset of the sample composed by 7 subjects were asked to perform the static propaedeutic, single-leg stance test in addition to the stabilometric one: results of the acquisitions are shown in Table 3.10. As expected, the stability maintenance is slightly more difficult in a single-leg stance situation, which typically occurs during the gait: the estimated average radius of postural cone of instability is 0.75° in the static propaedeutic test, against 0.56° estimated in the stabilometric test. The balance gets even worse if the visual information is absent: the stability index drop from an average value of 93.45 % in

eyes open condition down to an average value of 77.27 % in eyes closed condition. Moreover, the visual dependence index seems to be positively correlated with the age of the subjects: a low index value indicates a better intervention of proprioceptive and vestibular strategy, as one can expect in younger people. However, due to the small size of the sample, further investigations need to be performed in order to confirm this hypothesis.

Table 3.10 – Balance parameters related to the single-leg stance test.

N.	**Age**	**Eyes Open**			**Eyes Closed**			**VD**(%)
		COM(°)	$\mathbf{PI_{xy}}$(°)	**SI**(%)	**COM**(°)	$\mathbf{PI_{xy}}$(°)	**SI**(%)	
1	28	$(-0.59; 1.71)$	0.68	93.98	$(-4.54; -1.22)$	1.69	86.05	7.93
2	29	$(-1.36; -1.00)$	1.12	90.38	$(-2.06; 1.21)$	2.14	82.55	7.83
3	35	$(-1.63; -0.51)$	0.48	95.74	$(-2.57; 0.95)$	0.68	94.04	1.70
4	39	$(9.50; 2.71)$	0.36	96.77	$(17.59; 2.24)$	5.13	64.07	32.70
9	62	$(-1.53; 5.48)$	1.05	90.94	$(1.34; 5.25)$	2.77	78.42	12.52
11	68	$(-3.11; -0.98)$	0.68	94.05	$(-3.51; -0.54)$	5.46	64.70	29.35
16	76	$(-0.59; -3.99)$	0.89	92.31	$(-2.60; -1.93)$	3.95	71.05	21.26

New acquisition sessions are foreseen in the next future, involving a larger sample as well as pathological subjects. Moreover, the use of data from gyroscope will be investigated, in order to monitor angular velocities and extract angular accelerations generated to correct the posture.

References

[1] V. L. Feigin *et al.*, "Global, regional, and national burden of neurological disorders during 1990–2015: A systematic analysis for the global burden of disease study 2015", *The Lancet Neurology*, 2017. DOI: 10.1016/S1474-4422(17)30299-5.

[2] M. Caldara, D. Comotti, M. Galizzi, P. Locatelli, V. Re, D. Alimonti, M. Poloni, and M. C. Rizzetti, "A Novel Body Sensor Network for Parkinson's Disease Patients Rehabilitation Assessment", in *2014 11th International Conference on Wearable and Implantable Body Sensor Networks*, 2014, pp. 81–86. DOI: 10.1109/BSN.2014.28.

[3] M. Caldara, P. Locatelli, D. Comotti, M. Galizzi, V. Re, N. Dellerma, A. Corenzi, and M. Pessione, "Application of a wireless BSN for gait and balance assessment in the elderly", in *2015 IEEE 12th International Conference on Wearable*
10.1109/BSN.2015.7299386.

[4] A. Hadjidj, M. Souil, A. Bouabdallah, Y. Challal, and H. Owen, "Wireless sensor networks for rehabilitation applications: Challenges and opportunities", *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 1–15, 2013. DOI: 10.1016/j.jnca.2012.10.002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804512002196.

[5] I. Paraskevopoulos and E. Tsekleves, "Use of gaming sensors and customised exergames for parkinson's disease rehabilitation: A proposed virtual reality framework", in *2013 5th International Conference on Games and Virtual Worlds for Serious Applications (VS-GAMES)*, 2013, pp. 1–5. DOI: 10.1109/VS-GAMES.2013.6624247.

[6] M. Munneke, S. Kens, M. Nijkrake, G. Kwakkel, H. Berendse, R. Roos, and et al., "Efficiency of evidence-based physiotherapy for Parkinson's disease: the ParkinsonNet Trial", *The Lancet Neurology*, 2010. DOI: 10.1016/S1474-4422(09)70327-8.

[7] G. Frazzitta, P. Balbi, R. Maestri, G. Bertotti, N. Boveri, and G. Pezzoli, "The beneficial role of intensive exercise on Parkinson disease progression", *American Journal of Physical Medicine & Rehabilitation*, 2013. DOI: 10.1097/PHM.0b013e31828cd254.

[8] M. Giuberti, G. Ferrari, L. Contin, V. Cimolin, C. Azzaro, G. Albani, and A. Mauro, "Assigning updrs scores in the leg agility task of parkinsonians: Can it be done through bsn-based kinematic variables?", *IEEE Internet of Things Journal*, vol. 2, no. 1, pp. 41–51, 2015. DOI: 10.1109/JIOT.2015.2390075.

[9] M. F. Gago, V. Fernandes, J. Ferreira, H. Silva, M. L. Rodrigues, L. Rocha, E. Bicho, and N. Sousa, "The effect of levodopa on postural stability evaluated by wearable inertial measurement units for idiopathic and vascular Parkinson's disease", *Gait & Posture*, vol. 41, no. 2, pp. 459–464, 2015. DOI: 10.1016/j.gaitpost.2014.11.008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0966636214007759.

[10] P. Pierleoni, L. Palma, A. Belli, and L. Pernini, "A real-time system to aid clinical classification and quantification of tremor in Parkinson's disease", in *IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*, 2014, pp. 113–116. DOI: 10.1109/BHI.2014.6864317.

[11] World Health Organization, "Global health risks: Mortality and burden of diseases attributable to selected major risks", WHO, 2009. [Online]. Available: http://www.who.int/healthinfo/global_burden_disease/GlobalHealthRisks_report_full.pdf.

[12] ——, "What are the main risk factors for falls amongst older people and what are the most effective interventions to prevent these falls?", WHO, 2004. [Online]. Available: http://www.euro.who.int/___data/assets/pdf_file/0018/74700/E82552.pdf.

[13] S. Heinrich, K. Rapp, U. Rissmann, C. Becker, and H. H. König, "Cost of falls in old age: A systematic review", *Osteoporosis International*, 2010. DOI: 10.1007/s00198-009-1100-1.

[14] D. Riva, C. Mamo, M. Fanì, *et al.*, "Single Stance Stability and Proprioceptive Control in Older Adults Living at Home: Gender and Age Differences", *Journal of Aging Research*, 2013. DOI: 10.1155/2013/561695.

[15] World Health Organization. (2010). Global recommendations on physical activity for health, [Online]. Available: http://whqlibdoc.who.int/publications/2010/9789241599979_eng.pdf.

[16] M. Pahor, J. M. Guralnik, W. T. Ambrosius, *et al.*, "Effect of structured physical activity on prevention of major mobility disability in older adults: The life study randomized clinical trial", *JAMA*, vol. 311, no. 23, pp. 2387–2396, 2014. DOI: 10.1001/jama.2014.5616.

[17] G.-Z. Yang, *Body Sensor Networks*. Berlin, Heidelberg: Springer-Verlag, 2006, ISBN: 1846282721.

[18] B. S. Todd and D. C. Andrews, "The identification of peaks in physiological signals", *Computer and biomedical research*, 1999.

[19] W. Zijlstra and A. L. Hof, "Assessment of spatio-temporal gait parameters from trunk accelerations during human walking.", *Gait & posture*, vol. 18 2, pp. 1–10, 2003.

[20] R. C. González, A. M. López, J. Rodriguez-Uría, D. Álvarez, and J. C. Alvarez, "Real-time gait event detection for normal subjects from lower trunk accelerations", *Gait & Posture*, vol. 31, no. 3, pp. 322–325, 2010. DOI: 10.1016/j.gaitpost.2009.11.014.

[21] D. Trojaniello, A. Cereatti, E. Pelosin, L. Avanzino, A. Mirelman, J. M. Hausdorff, and U. Della Croce, "Estimation of step-by-step spatio-temporal parameters of normal and impaired gait using shank-mounted magneto-inertial sensors: Application to elderly, hemiparetic, parkinsonian and choreic gait", *Journal of NeuroEngineering and Rehabilitation*, vol. 11, no. 1, p. 152, 2014. DOI: 10.1186/1743-0003-11-152.

[22] J. McCamley, M. Donati, E. Grimpampi, and C. Mazzà, "An enhanced estimate of initial contact and final contact instants of time using lower trunk inertial sensor data.", *Gait & Posture*, vol. 36, no. 2, pp. 316–8, 2012.

[23] F. A. Storm, C. J. Buckley, and C. Mazzà, "Gait event detection in laboratory and real life settings: Accuracy of ankle and waist sensor based methods", *Gait & Posture*, vol. 50, pp. 42–46, 2016. DOI: 10.1016/j.gaitpost.2016.08.012.

[24] J. Luo, J. Bai, and J. Shao, "Application of the wavelet transforms on axial strain calculation in ultrasound elastography", *Progress in Natural Science*, vol. 16, no. 9, pp. 942–947, 2006. DOI: 10.1080/10020070612330093. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1080/10020070612330093.

[25] B. Mariani, C. Hoskovec, S. Rochat, C. Büla, J. Penders, and K. Aminian, "3D gait assessment in young and elderly subjects using foot-worn inertial sensors", *Journal of biomechanics*, vol. 43, pp. 2999–3006, Nov. 2010. DOI: 10.1016/j.jbiomech.2010.07.003.

[26] H. P. Nguyen, F. Ayachi, C. Lavigne–Pelletier, M. Blamoutier, F. Rahimi, P. Boissy, M. Jog, and C. Duval, "Auto detection and segmentation of physical activities during a Timed-Up-and-Go (TUG) task in healthy older adults using multiple inertial sensors", *Journal of NeuroEngineering and Rehabilitation*, Apr. 2015. DOI: 10.1186/s12984-015-0026-4.

[27] C. L. Tomlinson, C. P. Herd, C. E. Clarke, *et al.*, "Physiotherapy for Parkinson's disease: a comparison of techniques", *Cochrane Database of Systematic Reviews 2014*, 2014. DOI: 10.1002/14651858.CD002815.pub2.

[28] M. J. Falvo and G. M. Earthart, "Six-minute walk distance in persons with Parkinson disease: a hierarchical regression model", *Archives of Physical Medicine and Rehabilitation*, 2009. DOI: 10.1016/j.apmr.2008.12.018.

[29] D. Brooks, S. Solway, and W. J. Gibbons. (2003). ATS statement on six-minute walk test, [Online]. Available: https://www.thoracic.org/statements/resources/pfet/sixminute.pdf.

[30] S. S. Ng, P. C. Yu, F. P. To, J. S. Chung, and T. H. Cheung, "Effect of walkway length and turning direction on the distance covered in the 6-minute walk test among adults over 50 years of age: A cross-sectional study", *Physiotherapy*, vol. 99, no. 1, pp. 63–70, 2013. DOI: 10.1016/j.physio.2011.11.005. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0031940612000259.

[31] S. Miyamoto, N. Nagaya, T. Satoh, S. Kyotani, F. Sakamaki, M. Fujita, N. Nakanishi, and K. Miyatake, "Clinical Correlates and Prognostic Significance of Six-minute Walk Test in Patients with Primary Pulmonary Hypertension", *American Journal of Respiratory and Critical Care Medicine*, vol. 161, no. 2, 2000. DOI: 10.1164/ajrccm.161.2.9906015. [Online]. Available: https://www.atsjournals.org/doi/full/10.1164/ajrccm.161.2.9906015.

[32] D. Stevens, E. Elpern, K. Sharma, P. Szidon, M. Ankin, and S. Kesten, "Comparison of Hallway and Treadmill Six-minute Walk Tests", *American Journal of Respiratory and Critical Care Medicine*, vol. 160, no. 5, 1999. DOI: 10.1164/ajrccm.160.5.9808139. [Online]. Available: https://www.atsjournals.org/doi/full/10.1164/ajrccm.160.5.9808139.

[33] B. Camarri, P. R. Eastwood, N. M. Cecins, P. J. Thompson, and S. Jenkins, "Six minute walk distance in healthy subjects aged 55–75 years", *Respiratory Medicine*, vol. 100, no. 4, pp. 658–665, 2006. DOI: 10.1016/j.rmed.2005.08.003. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0954611105003264.

[34] G. Roul, P. Germain, and P. Bareiss, "Does the 6-minute walk test predict the prognosis in patients with nyha class ii or iii chronic heart failure?", *American Heart Journal*, vol. 13, no. 3, pp. 449–57, 1998.

[35] B. A. Silva, C. D. M. C. Faria, M. P. Santos, and A. Swarowsky, "Assessing Timed Up and Go in Parkinson's disease: Reliability and validity of Timed Up and Go Assessment of biomechanical strategies", *Journal of Rehabilitation Medicine*, 2017. DOI: 10.2340/16501977-2254.

[36] J. R. Nocera, E. L. Stegemöller, I. A. Malaty, M. S. Okun, M. Marsiske, C. J. Hass, and National Parkinson Foundation Quality Improvement Initiative Investigators, "Using the Timed Up & Go Test in a Clinical Setting to Predict Falling in Parkinson's Disease", *Archives of Physical Medicine and Rehabilitation*, 2013. DOI: 10.1016/j.apmr.2013.02.020.

[37] A. Kalron, M. Dolev, and U. Givon, "Further construct validity of the Timed Up-and-Go Test as a measure of ambulation in multiple sclerosis patients", *European Journal of Physical and Rehabilitation Medicine*, 2017. DOI: 10.23736/S1973-9087.17.04599-3.

[38] E. Sebastião, B. M. Sandroff, Y. C. Learmonth, and R. W. Motl, "Validity of the Timed Up and Go Test as a Measure of Functional Mobility in Persons With Multiple Sclerosis", *Archives of Physical Medicine and Rehabilitation*, 2016. DOI: 10.1016/j.apmr.2015.12.031.

[39] S. D. Din, A. Godfrey, and L. Rochester, "Validation of an Accelerometer to Quantify a Comprehensive Battery of Gait Characteristics in Healthy Older Adults and Parkinson's Disease: Toward Clinical and at Home Use", *IEEE Journal of Biomedical and Health Informatics*, vol. 20, no. 3, pp. 838–847, 2016. DOI: 10.1109/JBHI.2015.2419317.

[40] A. Ferrari, P. Ginis, M. Hardegger, F. Casamassima, L. Rocchi, and L. Chiari, "A Mobile Kalman-Filter Based Solution for the Real-Time Estimation of Spatio-Temporal Gait Parameters", *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 24, no. 7, pp. 764–773, 2016. DOI: 10.1109/TNSRE.2015.2457511.

[41] E. Rovini, C. Maremmani, and F. Cavallo, "How wearable sensors can support parkinson's disease diagnosis and treatment: A systematic review", *Frontiers in Neuroscience*, 2017. DOI: 10.3389/fnins.2017.00555.

[42] A. Salarian, F. Horak, C. Zampieri, P. Carlson-Kuhta, J. Nutt, and K. Aminian, "iTUG, a sensitive and reliable measure of mobility", *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 18, no. 3, pp. 303–10, 2010. DOI: 10.1109/TNSRE.2010.2047606.

[43] P. De Carli, M. Patrizi, L. Pepe, G. Cavaniglia, D. Riva, and L. R. D'Ottavi, "Postural control and risk of falling in bipodalic and monopodalic stabilometric tests of healthy subjects before, after visuo-proprioceptive vestibulo-postural rehabilitation and at 3 months thereafter: role of the proprioceptive system", *Acta Otorhinolaryngologica Italica*, 2010.

Chapter 4

The RADAR-CNS Project

Remote Assessment of Disease and Relapse – Central Nervous System (RADAR-CNS) is a new collaborative research program that will explore the potential of wearable devices to help prevent and treat depression, multiple sclerosis and epilepsy. It is funded by the Innovative Medicines Initiative — a Public Private Partnership set up between the European Federation of Pharmaceutical Industries and Associations (EFPIA) and the European Union. The project is jointly led by King's College London and Janssen Pharmaceutica NV and includes 23 organizations from across Europe and the US, including the University of Bergamo. It brings together an international consortium of academic and EFPIA members who are leaders in the fields of depression, multiple sclerosis and epilepsy, with clinical expertise and access to patient cohorts in each disease area, combined with leading technical and methodological experts in the disciplines required for RMT development and implementation.

Within the RADAR-CNS project, the contribution of the author of this thesis involves the analysis of physical activity data related to MS patients, collected by wearable devices. As such, this chapter reports the project description, the MS study design and the analysis plan (composed of methods and algorithms) which will be used to extract physical activity features.

Figure 4.1 – The RADAR-CNS project logo.
Source: RADAR-CNS website (*www.radar-cns.org/*).

4.1 Project Rationale

Healthcare systems face an unprecedented challenge in addressing the need posed by central nervous system (CNS) disorders. As costs of care rise, models of payment will have to shift from those based on clinical activity to those based on clinical outcomes. There is therefore a pressing need for healthcare providers to demonstrate value — defined as the per euro outcome benefit. Reactive models of healthcare provision typically fail to reward providers for acting early to prevent adverse health outcomes among patients with CNS disorders. If market conditions rewarded effective early intervention that prevented adverse outcomes then patients, carers and the wider health economy would benefit.

The last decade has seen an explosion in the capability of monitoring individuals via sensors in smartphones or wearable devices, and the range of parameters which can be measured by such technologies will continue to grow. The development of remote measurement technologies (RMT) is an innovation which could, in the foreseeable future, be used to predict and avert negative clinical outcomes by providing real time information on the patient's current clinical state, as well as providing predictive information indicative of a future deterioration [1]. In common with other "big data" projects, RMT has been hailed as a potential "paradigm shift" in the way in which clinical services can be delivered, but will require healthcare adaptation for its delivery. RMT is therefore a disruptive innovation which requires careful evaluation, with an eye always kept on implementation in real world settings. Without such evaluation the promise of RMT will not be accomplished [2]. CNS disorders are an excellent test bed for such evaluation because they are chronic diseases whose course is dynamic with multiple relapses in parameters that could be measured remotely and passively via unobtrusive on-body biosensors and smartphones [3, 4].

In this context, Remote Assessment of Disease and Relapse in Central Nervous System Disorders (RADAR-CNS) is a major international research project whose overall ambition is to transform patient care through remote assessment. To do so, the work group is building an infrastructure to identify clinically useful RMT-measured biosignatures and assist in the early identification of relapse or deterioration, which can be translated into health services within the next 5 years. The team will develop a platform to identify these biosignatures from well characterized patient cohorts and will anticipate potential barriers to translation by initiating a dialog with key stakeholders (patients, clinicians, regulators and healthcare providers) from the outset.

4.2 RADAR-base

Nowadays, a huge amount of data is constantly collected by smartphones, smartwatches and other wearable devices: from localization and activity, to heart rate and sleep pattern. Careful analysis of data gathered by more and more patients gives scientists more insight into their health status; more importantly, it enables early detection of relapses, allowing patients and clinicians to act quickly and adequately.

Since 2016, The Hyve (an innovative and fast-growing IT company) has been actively involved in RADAR-CNS, developing and integrating the required technologies and making the data accessible. In collaboration with other project partners, RADAR-base was launched in April 2018. The platform builds on experience gained with the RADAR-CNS project and enables scientific research with data from Android smartphones and wearable devices. Although RADAR-CNS focuses exclusively on three conditions of the CNS (epilepsy, multiple sclerosis, and major depressive disorder), RADAR-base is intended to incorporate studies on a wider variety of diseases, such as Alzheimer's Disease and different types of heart diseases.

4.2.1 RADAR-CNS Platform

The RADAR-CNS platform is a healthcare pipeline to obtain clinical and health data from various active and passive sources like questionnaires, wearable devices and phone sensors [5]. The platform is open source: it is based on top of Apache Kafka and is built as such to accommodate easy integration of future devices and applications to the platform. The infrastructure (Figure 4.2) includes data gathering, management and processing.

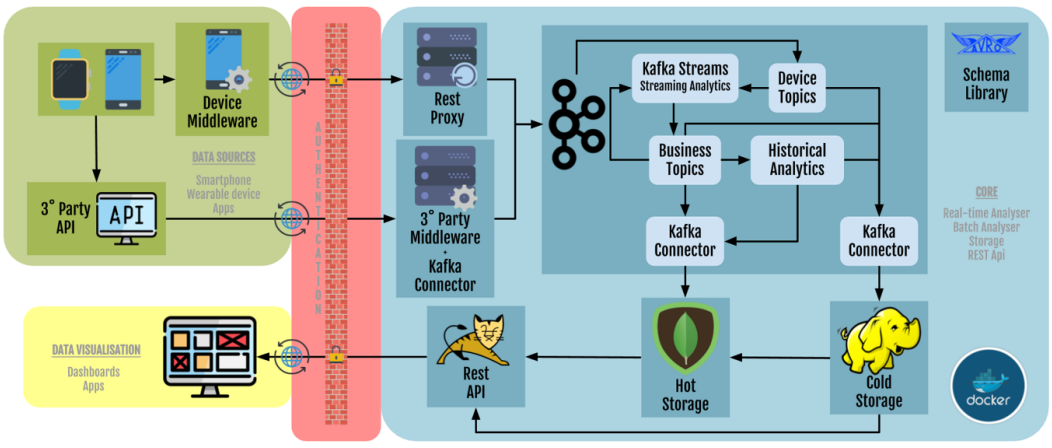


Figure 4.2 – Detailed schema of the RADAR platform.
Source: Amos Folarin and others, *The RADAR platform: an open source generalised mHealth infrastructure.*

Data from Bluetooth-enabled devices and phone data (such as accelerations) are gathered by means of the developed mobile Android applications (see section 4.2.4 below for more details). Then, applications send data payloads to the backend server over a secure HTTPS connection. Moreover, they store the data on the device to ensure a reliable data stream [6]. All data are schematized before sending, for a consistent data view across data sources.

RADAR-Backend is a Java application to standardize, analyze and persist data collected by RADAR-CNS data sources. The data are produced and consumed in Apache Avro format using the schema stored inside the RADAR-CNS schema repository. Firstly, the backend checks conformance of data to the schema. Also, according to the authorization scheme used, the backend can verify whether the data originates from the user that submitted them. The backend immediately rejects data that do not conform to our standards so the application can notify the user and erroneous data do not pollute our data streams.

RADAR-Backend provides an abstract layer to monitor and analyze streams of wearable data and write data to two different data warehouse layers (hot and cold storage). Once the data arrive at the Apache Kafka cluster, the backend monitors their contents and immediately processes the data with Kafka Streams: the results of these processing steps are stored in a MongoDB (i.e. the hot storage) for fast access. All raw data is stored to HDFS (the cold storage) for offline processing, like machine-learning with Apache Spark, or for exporting to external processing tools. The code to all components is available on GitHub under the open source Apache 2 License.

4.2.2 RADAR-CNS Management Portal

One of the main components of the RADAR-base infrastructure is the Management Portal. This is the central management service of the RADAR Platform, offering services like:

- Authorization of different components;
- Creating and storing information of *Subjects*, devices (aka *Sources*), studies (aka *Projects*), etc.;
- Creating relationship between different components (like registering a source to a subject).

A platform administrator can use the portal to create a new Project by providing the name, description and selecting the device family associated to it. Then, the

platform administrator can assign a user to the project and give them one or more roles, such as project administrator or participant. Once the project administrator account is activated, they are allowed to edit the project details, add subjects to the study and assign specific devices to the new users. However, they can only see the users associated with the projects that they administer.

4.2.3 Patient Enrollment

The patient enrollment process begins with REDCap (Research Electronic Data Capture): it is a secure web application for building and managing online surveys and databases. While REDCap can be used to collect virtually any type of data, it is specifically geared to support online or offline data capture for research studies and operations [7].

Before data can be collected, projects must be set up, including adding subjects to the platform and assigning devices to them [8]. The recruiter starts by entering information about the patient into REDCap, including Personally Identifiable Information (PII) and Protected Health Information (PHI). The recruiter also interacts with the RADAR-CNS platform and creates a new user by providing the token generated by REDCap as login credentials. The REDcap integration web app then integrates a REDcap Project with a Project on Management Portal linking corresponding subjects in both systems: this ensures that personal identifiable data (Human Readable Ids and UUIDs) is kept securely in REDcap (out of RADAR-base platform) and linked via pseudonymized data only (see Figure 4.3). The non-identifiable sensor data is stored in the RADAR-base platform [5].

Once enrolled, the patient is provided with one or more wearable devices, each of which will send data to the platform via the smartphone apps.

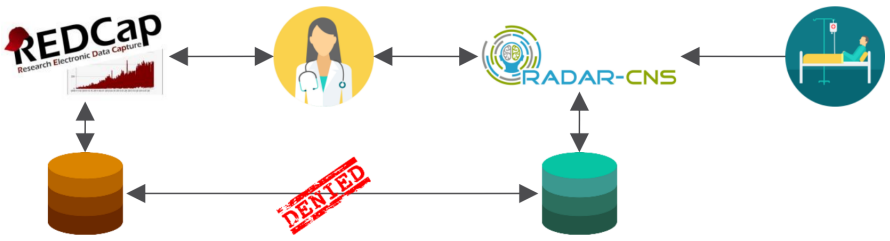


Figure 4.3 – Interaction between REDCap and RADAR-CNS Platform. The access to patient's PII and PHI is granted to clinicians only. The two systems only share a common key for each enrolled patient.
Source: *RADAR Platform: Large scale, open source wearables data processing for disease and relapse assessment.* From *The Hyve.*

4.2.4 Smartphone Apps

The smartphone collects data both passively and actively via two Android apps developed by The Hyve. The active remote monitoring (aRMT) app collects data from patients by using questionnaires that might ask them about their mood, medication intake, or the severity of symptoms (see Figure 4.4). Moreover, it allows users to perform timed tasks. The aRMT app gives feedback to subjects regarding their participation and can be configured according the the subject's preferences.

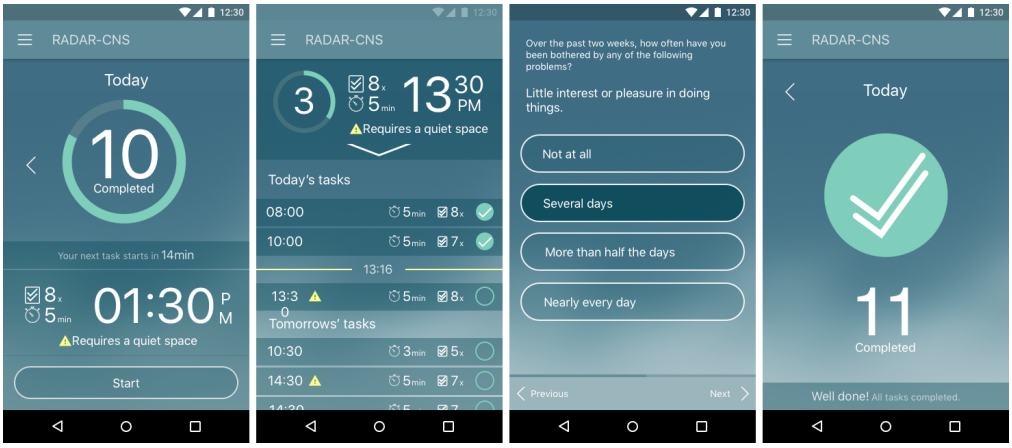


Figure 4.4 – The RADAR active RMT app.
Source: *RADAR active RMT*. From Google Play.

On the other hand, the passive remote monitoring (pRMT) app includes real time monitoring of movement, location, audio, calls and texts, and app usage. The app also collects data from wearable devices, such as the wrist-worn Empatica E4, Fitbit Charge 2 and Biovotion. In addition, the app provides information about server status and data upload, as well as information about remote changes to the app configuration (see Figure 4.5).

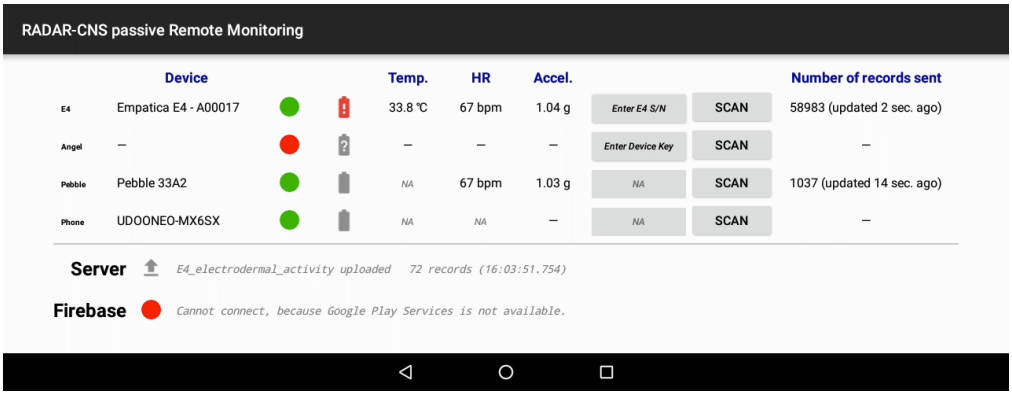


Figure 4.5 – The RADAR passive RMT app.
Source: *RADAR passive RMT*. From Google Play.

4.2.5 Data Visualization

Besides collecting and processing data, it is crucial to have tools to monitor data collection and visualize subject-specific data. At the time of writing of this thesis, a dashboard for this purpose was under development. According to the available mockups, the dashboard should provide summary statistics and the composition of the study population, as well as an overview of the infrastructure status. A patient overview should give timely information about the status of data collection across subjects, and should visualize compliance.

Figure 4.6 – RADAR dashboard mockup.
Source: *RADAR Platform: Large scale, open source wearables data processing for disease and relapse assessment.* From *The Hyve.*

The dashboard will enable the zoom into specific subjects with a summary of available data. The data will also be visualized at several resolutions, allowing study administrators to zoom in and examine data in a specific time period.

4.3 RADAR-MS Disability Study

The National Multiple Sclerosis Society (NMSS) define MS as "a chronic, unpredictable disease of the central nervous system (CNS), which is made up of the brain, spinal cord and optic nerves" (see section 1.3.1 for more details about the disease). MS typically begins between the ages of 20 and 40 years and it is the leading cause of non-traumatic disability in young adults [9]. This disease also represents a high economic burden with an estimated annual cost in Europe of € 14 billions [10].

MS patients can suffer from several impairments including: difficulties in breathing, fatigue, heat sensitivity and possible disordered body temperature, lower muscle strength, leading to lower physical activity, loss of balance, fatigue and problems with memory and concentration. These symptoms have an effect on the daily life as well as the quality of life of the afflicted individuals and vary between early and late stages of the disease.

4.3.1 Standard Assessment

Conventional outcome assessment in MS relies upon self-report, structured questionnaires and rating scales. The gold standard for disability assessment in MS is the Expanded Disability Status Scale (EDSS), which is performed by the neurologist at the clinical office [11, 12]. It is a quantitative rating scale, with scores ranging from 0 to 10, which relies heavily on ambulation.

Gait is one of the most important and valued functions for people with multiple sclerosis (PwMS). In fact, gait dysfunction is one of the main contributors to quality of life deterioration in PwMS [13]. Monitoring gait impairment has become a common outcome in both clinical trials and clinical practice in MS [12, 14]. Specific tests assessing gait performance in MS clinical settings include timed tests for walking speed (timed 25-Foot Walk Test, 25FWT) or distance (2MWT and 6MWT). In addition, there are MS specific patient-reported outcomes like the Patient Determined Disease Steps (PDDS) and the Multiple Sclerosis Walking (MSWS-12), which assess the impact of walking impairment in people with MS [15].

Besides gait dysfunction, MS-related fatigue is defined as "a subjective lack of physical and/or mental energy that is perceived by the individual or caregiver to interfere with usual or desired activities" [16]. Fatigue is not only the most prevalent symptom in MS, being reported by more than 90% of PwMS, but also it is considered their most disabling symptom by more than half of PwMS [17]. Assessment of fatigue in PwMS represents a challenge due to its subjective and multifaceted nature. Self-report questionnaires are the most used way for evaluating

fatigue in clinical research settings. There are numerous fatigue self-report scales, although the most commonly used in MS is the Fatigue Severity Scale (FSS). The FSS is a 9-item scale that measures the severity of fatigue and its effect on a person's activities and lifestyle. Each item is assessed by the patient using a 7-point Likert scale ranging from strongly disagrees (score 1) to strongly agree (score 7). Scores higher than 36 indicate severe fatigue [18].

Most of the mentioned measurements are only clinical, requiring patients to leave their homes and go to a hospital. This can be very uncomfortable for the patients, suffering from progressing motor disability. Furthermore, special trained examiners are needed to conduct these tests. In addition, these tests neither allow for an objective assessment of the degree of the disease nor do they give information about the daily routines. As such, it is desirable for MS assessment and treatment to have a broad range of non-invasive, easy-to-use and cheap measurements of disease progression. Wearable devices could give one of such additional possibilities. An important advantage of using wearables is that patients do not have to leave their home as they can perform the measurements in their known environment.

4.3.2 Study Design

"*RADAR-CNS: disability and fatigue study in MS*" is an observational non-randomized, non-interventional study, using commercially available wearable technology and smartphone's sensors, representing no change to the usual care or treatments of participants due to participation. The study will have a duration of 24 months, with no control comparison group nor randomized group allocation.

Participants will be recruited from three international Multiple Sclerosis Centre: Hospital San Raffaele (Milan, Italy), Fundació Hospital Universitari Vall d'Hebron - Institut de Recerca, VHIR, (Barcelona, Spain) and Region Hovedstaden (RegionH), Danish Multiple Sclerosis Center (Copenhagen, Denmark). Unfortunately, although the recruitment was due to start in October 2017 according to the project initial schedule, no participant was yet enrolled at the time of writing (autumn 2018) because of delays in the devices purchase.

Study Definition

Four-hundred people with MS will be included in the study: 300 with a relapsing-remitting MS and 100 with a secondary progressive MS. Relapse is defined as new or recurrent neurological symptoms not associated with fever, lasting at least 24 hours. New or recurrent neurological symptoms that evolve gradually over months

are to be considered disease progression, not an acute relapse. New or recurrent neurological symptoms that occur fewer than 30 days following the onset of a relapse as defined above, are to be considered part of the same relapse. Sustained disability progression is defined as: (1) an increase of more than 1 point from baseline EDSS, if the baseline EDSS is less than or equal to 5.5 points; (2) an increase of more than 0.5 points from baseline EDSS, if the baseline EDSS is greater than 5.5 points. This change has not to be attributable to another etiology (i.e. fever, concurrent illness, MS relapse or concomitant medication) and has to be confirmed at a regularly scheduled visit after at least 12 weeks after the initial disease progression.

Outcomes of Interest

Outcomes will be measured during clinical visits or remotely using either smartphone app or web-based platform. These outcomes are designed to provide a sensitive measure of disability status and can detect changes of disability over time.

The primary outcome of interest will be to assess the reliability of RMT in monitoring the disability status of the patients and its changes over time. For this purpose, data deriving from active and passive RMT will be correlated with clinical measures of disability. The primary outcome will be the EDSS and will be assessed face to face during the clinical visits every 3 months.

Secondary outcomes will be collected every 3 months through a web-based platform (REDCap) at home and/or during the clinical visits, or face to face during the clinical visits. They include: Symbol Digit Modality Test (SDMT); timed 25-Foot Walk (25FW) test; 9 Hole Peg test (9HPT); 6 Minute Walk Test (6MWT); Fatigue Severity Scale (FSS-9); Work and Social Adjustment Scale (WSAS); Short Form Health Survey (SF-36); and Brief Illness Perceptions Questionnaire (BIPQ).

Procedure

Clinical visits will be performed at baseline, and then every 3 months. Participants will also undergo remote periodic evaluation with self-reported questionnaires to evaluate different aspects of disability, fatigue and mood changes. These evaluations will be performed either through the aRMT smartphone app or through the web-based platform (REDCap). During the first visit (month 0), participants will receive a 60-minute training session on the use of the wearable devices, mobile sensors, and smartphone app. At the end of this training session, they will be given the devices to take home, and active recording of data will begin. Baseline assessments will be conducted face-to-face during recruitment as part of the introductory training session. All the clinical visits will include the following measurements:

- Questionnaires about socio-demographics, social environment, medical history, MS history, treatment history and technology usage (at month 0 only);
- Self-reported questionnaires for primary and secondary outcomes assessment;
- Assessment of gait skills (2MWT, Two-Minute Walk Test) and standing balance (Romberg's test and Tandem Walking test) based on a commercial chest-worn wearable device (eMotion Faros 180).

In addition to the clinical usage, the patient will be asked to wear the chest-worn device for seven consecutive days starting from the day of the clinical visit (every 3 months), in order to collect ECG and accelerometer data during daily life activities. The device will be worn using disposable, single-use ECG patches along the participant's sternum. The participants will only be asked to wear the device during the day. Moreover, the day after the clinical visit participants will be asked to perform the 2MWT, the Romberg's test and the Tandem Walking test at home, as performed during the visit. They will receive an advice through the RADAR-CNS app with instructions about how to perform those tests.

Lastly, participants will be asked to wear a commercial wrist-worn device (Fitbit Charge 2) for the entire study period, during which longitudinal heart rate and accelerometer data will be prospectively collected. During the same period, standard built-in smartphone sensors will collect data on call activity, location, ambient noise and ambient light by means of the pRMT smartphone app.

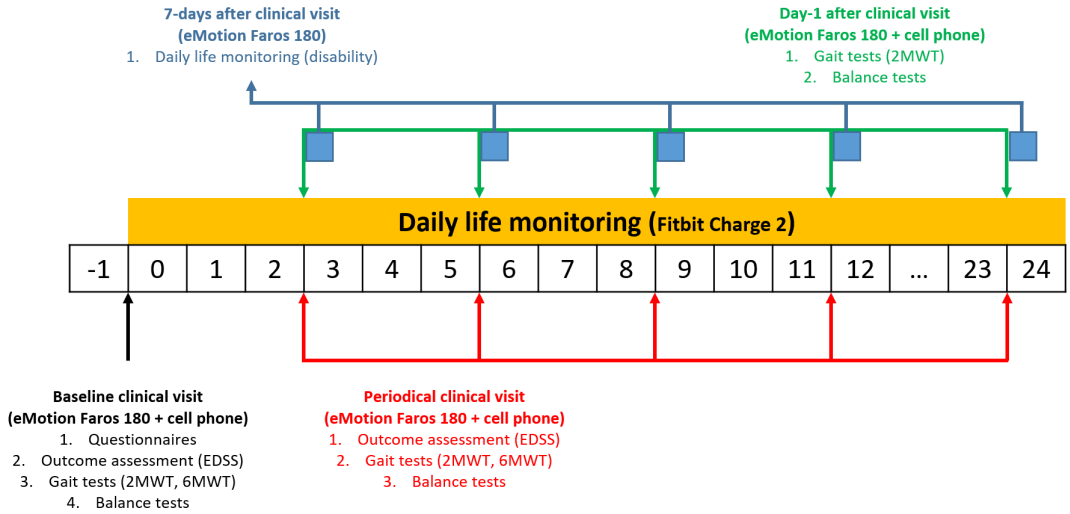


Figure 4.7 – Overview of the RADAR-MS Disability study procedure.

4.4 Gait and Balance Assessment

4.4.1 Gait Analysis in MS: Literature Review

The gait of MS patients is usually measured in clinical settings and through subjective assessments by the medical personnel. Gholami et al. [19] stated that with this setting and this kind of assessment, examiners were unable to detect subtle changes in a patient's gait. A reason for this statement is that patients could not be observed during their daily life.

A standard test in this field is the Six-Minute Walk Test (6MWT). Qureshi et al. [20] used a 3-axis accelerometer and a minute-separated version of the 6MWT to enable the analysis of changes in gait features for a cohort of 28 MS patients. Every minute was analyzed separately through monitoring of gait speed and cycle length. The z-plane signal of the accelerometer was filtered and analyzed by a peak-to-peak detection. The range between two peaks was considered as one gait cycle. They showed that MS patients, which suffer from fatigue, have particularly reduced gait speed and a high cycle length variance. Engelhard et al. [21] conducted a minute-separated 6MWT as a mean to detect subtle, progressive changes on a cohort of 86 MS patients and 29 healthy controls. By applying the Dynamic Time Warping (DTW) to their acceleration data, they were able to detect small changes over time during the 6MWT that were especially caused by fatigue.

A second standard test is the Timed Up and Go test (TUG): participants, who are seated in a chair in the beginning of the test, are timed as they stand up from a chair, walk a specified distance, turn around and sit down again [22]. This test was used by Spain et al. [23] in a series of experiments to observe the sway acceleration amplitude, medio-lateral flexion, axial rotation and turning duration in the study of 31 MS patients and 28 healthy controls. The participants wore six sensors, each consisting of a gyroscope and accelerometer. They were able to distinguish MS patients from healthy persons by means of above mentioned parameters (p-value = 0.02, p-value = 0.017, p-value = 0.026, and p-value = 0.031 respectively). In contrast to Spain et al., Greene et al. [24, 25] used the TUG to derive mobility parameters for automated classification of MS patients. In their study 38 MS patients and 33 healthy controls wore shank mounted accelerometers. Greene et al. were able to discriminate between MS participants and healthy controls with an accuracy of 96.90 %.

In contrast to the previous works, Grenez et al. [26] developed a system, which is based on the bending and pressure sensors placed on the same positions as in the previous study, except of the one inside a shoe. An additional bending sensor was

also mounted on an ankle. By analyzing these two types of sensor signals, Grenez et al. were able to differentiate between healthy and pathological gaits. The accuracy for the right pressure sensor was 86 %, for the left 93 %, for the heel 100 %, and for both bending sensors 100 %. Nevertheless, it must be noted that this study did not include any MS patients, but was conducted on 7 healthy participants simulating gait problems.

4.4.2 Two-Minute Walk Test

The Two-Minute Walk Test (2MWT) is a shorter version of the 6MWT, which is the simplest and cheapest method of indirectly assessing physical capacity, that is widely available and commonly used (see section 3.4). Unlike the 6MWT, in which the standard corridor length is 30 meters, there are no indications about how long the corridor has to be for the 2MWT. However, considering that this test will be performed also at home, it was decided to reduce the standard 6MWT corridor length from 30 to 10 meters.

Methods

In this version, the participant is asked to walk back and forth in a flat, straight corridor as longest distance as possible during 2 minutes, and the overall walked distance is measured. The participant is asked to walk as fast as possible during the execution of the test: they are permitted to slow down, to stop and to rest if necessary, but they have to resume walking as soon as they are able to do so. The test can be performed either indoor or outdoor, provided that the requirements about the walking course are satisfactory. The turning points can be marked with cones or any other objects that do not interfere with the walk, and the turning regions should be walked as close as possible to the markers.

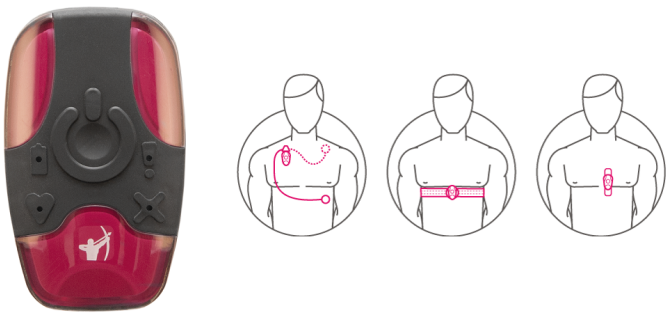


Figure 4.8 – eMotion Faros 180 and mounting positions.
Source: Biomation, *eMotion Faros Cardiac Monitors*.

Data will be collected by using the eMotion Faros 180 (Figure 4.8): it is a chest-worn device integrating a 3-axial accelerometer and an ECG sensor, which can be fixed along the user's sternum by means of ECG patches. Raw data are stored in the device memory and will be downloaded by the Android aRMT app at the end of each test, then it will be uploaded to the RADAR-base platform.

The integrated accelerometer will allow to detect steps, whereas the ECG sensor will be used to monitor the fatigue during the execution of the test (data from the latter sensor will be analyzed by other RADAR-CNS partners). However, the selected device lacks of the gyroscope, which is a very important sensor when dealing with the 2MWT (or with its longer version) as already discussed. To cope with this, a smartphone integrating such sensor will be mounted on the chest of the participant by means of a harness: along with the commercial device, it will collect information about the angular speed of the subject during the exercise, allowing the re-use of some of the analysis methods described in the previous chapter.

Test Parameters

Gait features related to the 2MWT will be extracted by using the same techniques described in section 3.4. Considering the available sensors and that the test will be performed both at home (unsupervised environment) and in clinics (supervised environment), the only available solution is to adopt methods based on inertial data.

The list of the test parameters is reported below:

- Number of steps (data source: trunk's vertical acceleration).
- Average turning time, in seconds (data source: trunk's angular rate).
- Total walked distance, in meters.
- Average step length, in meters
- Average and maximum step cadence, in steps/minute.
- Average and maximum walking speed, in meter/second.

4.4.3 Romberg's test

The Romberg's Test is used for the evaluation of balance in static conditions, similar to the stabilometric test descibed in section 3.6.

Methods

The participant is asked to stand still with feet together and hands along the sides. The first half of the test is performed with eyes open, whereas the second half is performed with eyes closed. The test duration may vary depending on the age of the participant and their ability to maintain balance. It was decided to perform an acquisition of 20 seconds: the first 10 seconds with eyes open, the last 10 seconds with eyes closed. Data will be collected by using the eMotion Faros 180 and transmitted to the RADAR-base platform by the Android aRMT app.

Test Parameters

Balance features will be extracted by using the same techniques described in section 3.6. In particular, only the center of mass (COM) and the average radius of postural cone will be estimated.

4.4.4 Tandem Walking test

The Tandem Walking Test is used to exacerbate all gait problems, especially those related to balance. During the execution of the test, many aspects are monitored (i.e. shift of the pelvis, flexion of the knee). However, within the RADAR-CNS project, this test will just be used to evaluate the capability of the participants to maintain balance during the gait.

Methods

The participant is asked to walk in a straight line with the front foot placed such that its heel touches the toe of the already standing foot (*tandem gait*). There are no indications about how many steps the patient has to walk: a number of steps equals to 10 was chosen. Data will be collected by using the eMotion Faros 180 and transmitted to the RADAR-base platform by the Android aRMT app.

Test Parameters

Balance features will be extracted by using the same techniques described in section 3.6. In particular, only the center of mass (COM) and the average radius of postural cone will be estimated.

4.5 ADL Monitoring

RADAR-MS Disability study involves the monitoring of patients in real-life conditions, that is, during activities of daily living (ADLs or ADL): for this purpose, two commercial wearable devices will be used.

The first device will be the Fitbit Charge 2 (Figure 4.9). It is a consumer-grade device that uses a 3D accelerometer and photoplethysmography sensor to measure activity and heart rate, respectively. Activity and heart rate data collected by the Fitbit are processed automatically to report step count and active minutes, whereas sleep detection algorithms are applied once data has been synced to the Fitbit cloud. However, since these algorithms are proprietary, neither the processing methods nor the raw data can be accessed for the RADAR-MS study.

Processed data are aggregated into minute-long epochs and are made available via Fitbit's intra-day API. The API reports:

- Total step count per minute, across the entire day;
- Activity level per minute, across the entire day (calculated by applying thresholds to "accelerometer activations" and categorizing minutes as sedentary, lightly active, fairly active, or very active);
- Heart rate, reported in 1-second or 1-minute intervals across the entire day, as calculated by Fitbit's detection algorithm;
- Sleep stages (deep, light, REM, and wake) with 30-second granularity.

Through automatic API calls, the output of the proprietary algorithms will be periodically uploaded from the Fitbit cloud onto the RADAR-base platform, where automated procedures will aggregate such information into a higher granularity level in order to resume the total performed activity day by day.

Figure 4.9 – Fitbit Charge 2.
Source: Fitbit website (*https://www.fitbit.com/it/charge2*).

In addition to the Fitbit device, participants will be asked to wear the eMotion Faros 180 device (described in section 4.4) during the seven days starting from the day of each clinical visit. Besides its usage for monitoring of gait and balance tests at home during the first day after each visit, this wearable device will also be used to collect accelerometer and ECG data during the 7-day monitoring period. The participants will only be asked to wear the device during the day. Raw data will be transmitted via Bluetooth to the smartphone, and then uploaded onto the RADAR-base platform by means of the Android aRMT app. By analyzing such data, several information about day-by-day physical activity will be extracted:

- Steps count;
- Total count of movement;
- Sedentary time, in minutes;
- Moderate-to-vigorous physical activity (MVPA), in minutes;
- Average and maximum walking cadence, in steps/minute;

Methods to extract such parameters from data collected by the chest-worn device will be widely discussed in the current section.

A third device collecting raw accelerometer data from a wrist-worn position was undergoing tests to verify that it meets other study inclusion criteria at the time of writing this thesis. If this device is implemented, a small sample of participants will worn it on their wrist along with the Fitbit, allowing side-by-side comparison of the utility between the Fitbit's processed data and raw accelerometer data: in order to account for this possible scenario, algorithms related to physical activity monitoring and step detection were developed as if data were collected by a wrist-worn device. In fact, if such algorithms will be able to deal with the wide range of movements related to that part of the body, their usage on data from the chest-worn device (eMotion Faros 180) will not represent a problem, since the trunk is typically subject to less movements than the wrist.

Although these devices are the ones that will be used within the ADL monitoring study, none of them was actually available for preliminary data acquisition at the time of writing (autumn 2018) due to delays in the purchase of the devices: therefore, no data from real patients were available to implement any algorithm. In order to obtain some initial data and develop the first version of the ADL analysis algorithms, the MuSe platform was used in place of the selected devices: such

platform, in fact, is able to provide the same raw inertial information of both the eMotion Faros 180 and the device under test. Moreover, it can be used as an activity tracker (thus emulating the unavailable Fitbit device) if combined with the proper ADL analysis algorithms, as shown in the next paragraphs. Once the devices will be purchased and ready to be used within the project, the developed algorithms will be adapted to work with data collected from the commercial devices, and fine-tuned to cope with data from actual patients.

4.5.1 Physical Activity

The primary goal of the activity monitoring within the RADAR-MS study is to distinguish low-intensity activities from high-intensity ones, so that clinicians can have a general insight about the level of movement of their patients when they are at home. In addition to intensity, a secondary analysis that might enrich such activity monitoring is related to the identification of the particular task the subject performs during a predefined time window. At these purposes, an algorithm based on raw data provided by the accelerometer (either the one integrated in the Faros device, or the one integrated in the third device under test in case this will be approved) was developed. The algorithm is mainly based on a classification procedure that is able to determine which activity, among a predefined series, is associated with data collected during a specific time window. Although nowadays commercial activity trackers are able to distinguish a wide range of activities, this study focused on four simple tasks only: resting, working at a personal computer, walking and running. In fact, considering the primary interest of the RADAR-MS study, such small subset of activities was considered sufficiently significant.

Methods

Ten healthy subjects with an average age of 29 years (SD = 4) were recruited to collect data during the execution of daily life activities. Participants were monitored by means of one MuSe inertial platform (see section 2.2), mounted on the wrist of their non-dominant hand by means of an elastic strip: the mounting position was specifically selected to emulate Fitbit's behavior, as well as to collect data similar to those potentially collected by the device under test (in case this will be approved). Each of the participants was asked to perform only one specific activity per acquisition, in order to keep the data separated. In addition to users' activities, some acquisitions were performed when the device was not worn: the identification of such state, in fact, will allow the platform to automatically determine when the device needs to run the algorithms, in the next future. Log sessions were performed

at a frequency rate of 25 Hz for a minimum time of one hour: Table 4.1 reports the recording times for each activity type.

Table 4.1 – List of activities and related recording time.

Activity	Description	Recording Time (min)
Not worn	-	285
Resting	Watching TV, reading a book	456
Working at a PC	Typing, Internet search	592
Walking	Walking at different paces	416
Running	Running at different paces	304

Analysis Procedure

The objective of the analysis was to determine the type of activity the user performed within fixed-size periods named *epochs*. The selected epoch size was 60 seconds: this duration was chosen because it makes it possible to include enough data to overcome the high variability of the wrist movements. In fact, by choosing a narrow window, brief time intervals of high-intensity wrist movements (e.g. due to gesture while talking) would result in a high-level activity, even if the user is in resting conditions; on the other hand, a wider window is more likely to include activities of different type.

In order to represent the quantity (or intensity) of movement associated with the recorded accelerations, a virtual elementary unit called *count* was defined as

$$counts = \sum_{i=1}^{w \cdot f_s} acc_i \cdot \Delta t = \sum_{i=1}^{w \cdot f_s} \frac{acc_i}{f_s}, \tag{4.1}$$

Thus, given a sampling period Δt (or equivalently, a sampling frequency f_s), the counts represent thus the area under the acceleration signal (computed by numerical integration) within a defined time window of length w. Although their measurement unit resemble a speed, such definition of counts originates from the way the first pedometers (i.e. step counters) "counted" the activity level, that is, by integrating the electric charges generated by the bending of a piezoelectric cantilever beam due to an external acceleration.

The conversion from acceleration to counts started with the computation of the magnitudes of the measured 3D acceleration vectors. The gravity component (1000 mg) was then removed from such magnitude values, and their measurement unit was converted from mg to m/s^2. Finally, the squared of the resulting magnitudes were computed, so that the successive numerical integration would only lead to

positive values of counts (movement intensity level).

By means of this simple conversion procedure, the first important information required by the RADAR-MS study will be easily obtainable: the total amount of movement per day, computed as the overall sum of the counts in a day. Moreover, by selecting proper threshold levels on the amount of counts per epoch, the differentiation between sedentary, moderate and vigorous physical activities will be easily performed, allowing the rapid computation of the other two pieces of information needed within the study (i.e. sedentary time and MVPA time).

However, in order to take a deeper insight, several features were extracted from both the acceleration signal and the counts level to build a classifier able to distinguish the different activity types. As a result, the total sedentary time per day would be computed as the number of epochs labeled as "Resting" or "Working at PC", whereas the total MVPA time per day would be computed as the number of epochs labeled as "Walking" or "Running". The feature extraction process was performed by means of two different methods: the first one included some domain knowledge by taking into account how activities typically change from a short time window to the other; the second one followed a more generic approach and involves the computation of some standard temporal and spectral features.

The first feature set FS_1 relied on the intensity variability of the physical activities. Each epoch was split into four time sub-windows of the same duration w: since the epoch size selected for this analysis procedure was one minute, each sub-window represented an acquisition of 15 seconds. The sum of the counts was then performed for each sub-window, leading to four different summation values per epoch.

In general, activities such as walking and running are characterized by a high-intensity physical activity level, almost constant within a short time window (e.g. one minute). On the other hand, activities such as typewriting and reading show a higher variability, with brief high-intensity movements alternating to long periods of very low-intensity ones (as in a steady state). For this reason, the overall sum of counts and the coefficient of variation

$$CV = \frac{\sigma}{\mu} \tag{4.2}$$

of the four summation values were computed for each epoch, representing the total movements intensity level and their variability, respectively. Figures 4.10 and 4.11 depict the scatter plot of the coefficients of variation as a function of total counts for each activity class.

On the other hand, the second feature set FS_2 was computed on the basis of the

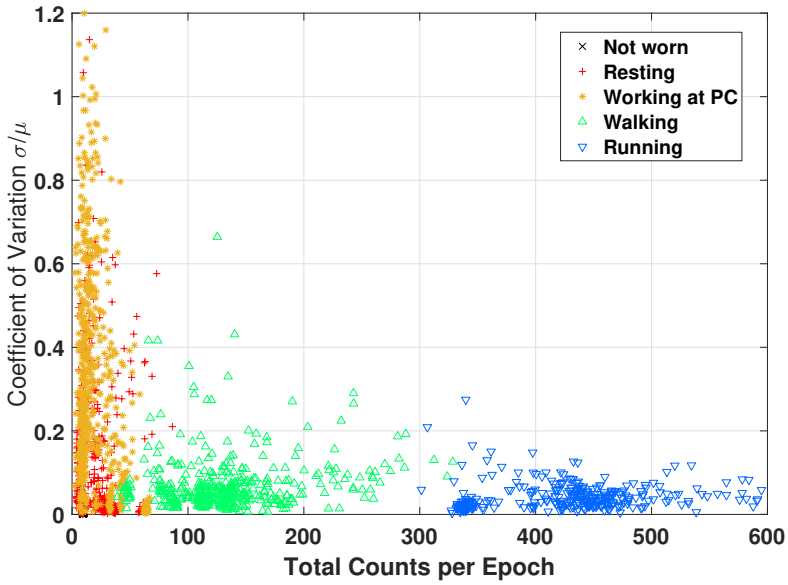


Figure 4.10 – Coefficients of variation as a function of total counts. Clusters associated to each activity class are clearly visible.

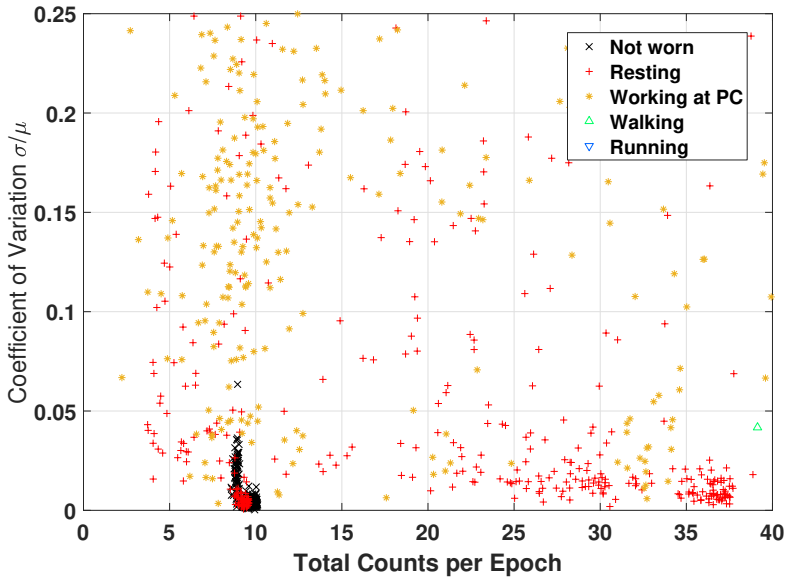


Figure 4.11 – Zoom of the "CV vs Total Counts" scatter plot in low intensity/low variability region. A cluster of "Not worn" points is visible between (10; 0) and (10; 0.05) coordinates.

raw signals, specifically the magnitude of the acceleration vectors. In fact, since the wearable tracking device was mounted on the wrist, it was not possible to determine any useful information from data on the single axes due the freedom of movement of such body part. On the contrary, if the device had been mounted on the chest of the subject, the direction of the gravity would have been used to detect the orientation of the device, thus increasing the available information: for instance, the detection of the gravity component on any axis different from the

vertical one, would exclude walking-related activities from the classification, in favor to the sedentary ones. This latter use case, however, could potentially prevent from distinguishing between activities in which the chest movements are similar, as in the case of "Resting" and "Working at PC".

Starting from the magnitude of the measured accelerations, the following features were extracted for each epoch:

- Mean value and standard deviation of the signal;
- Height (P) and location (L) of the first three peaks in the frequency spectrum;
- Height of the main peak (P1), as well as location and height of the second peak (P2, L2), in the signal's autocorrelation;
- Total signal power in five adjacent, predefined frequency bands (PB): 0.5–1.5 Hz, 1.5–2.5 Hz, 2.5–3.5 Hz, 3.5–4.5 Hz, and 4.5–10 Hz.

Finally, two different groups of classifiers were built based on FS_1 and FS_2 feature sets, respectively. The building procedure followed the one proposed in section 2.6 for the tremor classification: for each learning algorithm, a first, loosely optimized model was trained with the entire feature set; then, a small subset of features was selected and used to train the final model. However, contrary to that problem, the differentiation of physical activity types is a multiclass classification problem: as such, Support Vector Machines were not used since they are inherent to two-classes classifiers only (the most common technique to do multiclass classification with SVMs in practice would require to build several one-versus-rest classifiers).

Results and Discussion

Tables 4.2 and 4.3 list the out-of-sample accuracies of classifiers built from the FS_1 and FS_2 feature sets, respectively. In addition, for each learner the results of the feature selection and hyperparameters optimization procedures are reported.

Out-of-sample accuracies ranged from 77 to 90 %, with best results obtained with the k-Nearest Neighbors and the decision tree learning algorithms. Nevertheless, such good achievement is counterbalanced by the complexity of the trained models, either in terms of numerosity of selected features or in terms of hyperparameters optimization. For instance, the 90.21 % accuracy of the kNN model built from FS_2 was obtained by considering a neighborhood composed of 99 samples: this would require computing the weighted Euclidean distance from lots of points anytime a new sample is to be classified, with high computational costs. The lowest accuracies

Table 4.2 – Classification models trained on feature set FS_1. For each model, the selected feature subset, the optimized hyperparameters and the out-of-sample accuracy are reported.

Learner	Feature set	Hyperparameters	Accuracy (%)
k-Nearest Neighbors	Counts CV	Neighbors Number: 15 Distance Metric: Minkowski (exponent: 0.74236) Distance Weight: squared inverse Standardize: yes	83.97
Naïve Bayes	Counts CV	Distribution: kernel Kernel Type: normal Width: 0.013948	82.03
Discriminant Analysis	Counts CV	Type: quadratic Gamma: 0	77.40
Decision Tree	Counts CV	Min Leaf Size: 8 Max Number of Splits: 138 Split Criterion: twoing	84.22

Table 4.3 – Classification models trained on feature set FS_2. For each model, the selected feature subset, the optimized hyperparameters and the out-of-sample accuracy are reported.

Learner	Feature set	Hyperparameters	Accuracy (%)
k-Nearest Neighbors	Mean, STD	Neighbors Number: 99 Distance Metric: Euclidean Distance Weight: squared inverse Standardize: no	90.21
Naïve Bayes	Mean, STD Spect: L1, L2, L3 Autocorr: P2	Distribution: kernel Kernel Type: normal Width: 0.025848	85.05
Discriminant Analysis	Mean, STD Spect: P1 Spect: L2, L3 Autocorr: P1, P2 Autocorr: L2 PB: 2, 4	Type: quadratic Gamma: 0	79.59
Decision Tree	Mean, STD Spect: P2, P3 Spect: L2, L3 Autocorr: L2 PB: 1, 3, 4, 5	Min Leaf Size: 8 Max Number of Splits: 60 Split Criterion: twoing	89.19

were obtained from models based on Discriminant Analysis in both groups of classifiers, with 77.40 % and 79.59 % of accuracy, respectively.

In general, classifiers built from the second feature set (temporal and spectral features) performed better than those trained on the first one (total counts and coefficients of variation), probably due to the higher variety of features contained in FS_2 with respect to FS_1. Nevertheless, in some cases models built from FS_1 might still be preferred. Simple features, in fact, are often associated to low-complexity extraction methods, and thus to low computational costs: this is an important requirement in order for a classification algorithm to be embedded in a wearable device. In this sense, the need for a trade-off between simplicity and accuracy might favor the usage of less performing models based on simpler features, as in the case of FS_1, against more performing models based on more complex ones (as FS_2): the final choice strongly depends on the costs associated to the feature extraction algorithms needed, as well as the computational power of the target device.

The relatively low general accuracy obtained from the trained models mainly depends on the fact that two out of the five selected classes are similar. As can be noticed in Figures 4.10 and 4.11 in fact, "Resting" and "Working at PC" samples largely overlap in the CV vs Total Counts scatter plot: they are both characterized by brief, high-intensity movements, alternated to long, low-intensity ones. This assertion is supported by the confusion matrices of the trained model too, which clearly show how the major source of misclassification lies within the differentiation of those two classes (as depicted in Figure 4.12).

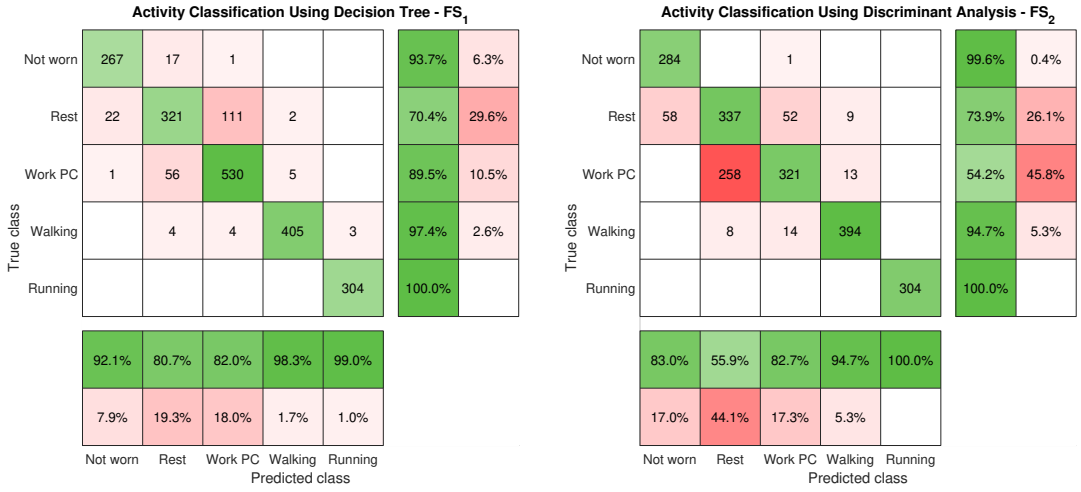


Figure 4.12 – Confusion matrices of the activity classification performed with two different learners, on both feature sets. Diagonal and off-diagonal cells are colored in green and red, respectively (gradation proportional to the cell value normalized to the largest one).

Although for the purpose of the RADAR-MS study such misclassification will not be very critical (both resting and working at PC tasks will be considered as sedentary activities), the common characteristics of such classes can preclude the possibility of distinguishing between them. Since, without a doubt, resting and working at PC are very similar activities, one workaround might be to merge such categories to create one class only. Table 4.4 reports the out-of-sample accuracies of the classification models, trained with dataset containing four labels only: "Resting" and "Working at PC" observations, in fact, were merged into one single class, and such resulting sample group was then randomly halved to preserve the overall dataset balance. As can be noticed, accuracies increased greatly, being above 95 % and 97 % for the FS_1- and FS_2-based models, respectively: such small accuracy difference would contribute to shift the model selection towards one among those based on simpler features. Nevertheless, it is important to remark that there is no such thing as a free lunch: the considerable accuracy increase was paid with a lower detail level in the classification task (only four classes against the original five).

Table 4.4 – Classification models trained on both feature sets with four classes only: "Not worn", "Resting" (obtained by merging the original resting label and the "Working at PC" one), "Walking" and "Running".

Learner	**Accuracy (%)**	
	FS_1	**FS_2**
k-Nearest Neighbors	95.80	98.45
Naïve Bayes	95.02	97.61
Discriminant Analysis	95.08	97.61
Decision Tree	95.67	98.77

4.5.2 Step Counter

Another outcome of interest for the RADAR-MS study is represented by the number of steps walked throughout a day. In recent years, in fact, step counters have been used to observe activity and as a motivational tool to support physical activity in clinical settings [27, 28]. Nowadays, a wide range of commercial step counters exists, with Fitbit, Jawbone, Nike and Garmin being among the famous ones. Due to the lack of evidence regarding their reliability and validity, however, there is a cautious use of commercial step counters in research and clinical settings, as discussed by Thorup et. al [29]: accuracy studies have reported counting errors in the step readings of such devices, particularly in cases of slow walking, high body mass index and individuals with gait disorders and functional limitations. Moreover, there is no agreement on the required accuracy level among researchers:

some studies recommend that an acceptable measurement error under controlled conditions or for research purposes is within ±3 % [30, 31], whereas other studies recommend that mean errors of less than 20 % have acceptable validity for clinical usage [30, 32]. Most of commercial pedometers are intended to be used for fitness purposes such as walking and running, therefore they are generally set to work in a conservative way: underestimating the number of steps does not represent a big issue, since such user category is already prone to the physical activity and at worse they will be more motivated to take more steps. This does not stand when using such devices in a clinical study: a wrong step number estimate could lead to a bad evaluation of the subjects' physical capability (if the estimated step number is lower than the actual one), or, even worse, to an underestimation of the real physical inability (if the estimated step number is greater than the actual one).

The high variability of accuracy in step counting strongly depends on the mounting position of the wearable pedometer. Clearly, devices mounted on the legs are more likely to estimate a correct number of steps with respect to wrist-worn devices, due to the wider range of possible arm movements. Also, waist-worn and chest-worn pedometers are characterized by high accuracies too: they implement the step identification on the basis of the measured vertical accelerations, produced by the feet impacts and spread throughout the entire body. Storm et al. [33] compared seven commercially available activity monitors in terms of step count detection accuracy, under different walking conditions and speeds. As expected best performances were obtained from devices mounted on the lower back (DynaPort Movemonitor) and at waist level (Fitbit One) at high walking speeds, with mean percentage errors below 3 %; the two wrist-worn devices (Jawbone Up and Nike+ Fuelband), instead, resulted in mean percentage errors varying from 4 % to 35 %, on the basis of the test conditions.

The goal of this study was to develop a step detection algorithm able to compete with those provided by actual commercial devices. The algorithm was purposely designed on the basis of simple subroutines in order to facilitating its embedding into research-grade wearable devices in the next future, but it is also usable as an off-line procedure as for the RADAR-MS study purposes. Moreover, the algorithm was developed to identify steps from data collected by a wrist-worn device, in order to cover all the possible scenarios within the RADAR study, as previously discussed. Once the number of steps will be available within predefined time interval, i.e. one minute, derived parameters such as mean and maximum step cadence will be easily computed.

Methods

Four young subjects with an average age of 26 years (SD = 2) were recruited and asked to walk while wearing a MuSe platform on the wrist of their non-dominant hand, in the very same way as in the monitoring of physical activity discussed above (see section 4.5.1).

Step counter accuracy has often been studied in healthy adults and under controlled conditions, for example, on a treadmill. The advantages in performing tests by using such instrumentation are the possibility of determining inaccuracies at different walking parameters values, such as speed, duration, cadence and distance. Nevertheless, treadmill walking is not representative of normal walking. For these reasons, two different test types were planned:

Structured tests Each of the four subjects was asked to walk on a treadmill for 10 minutes at two different speeds, i.e. 3 km/h and 6 km/h: such values were selected since they represent the extrema of typical walking speeds. Tests were performed twice per subject and speed, and each time the walked steps were manually counted. This type of test should highlight any possible relationship between the accuracy of the algorithm and the gait speed.

Unstructured tests One of the four subjects was asked to perform different walking sessions in a free-living environment (both indoor and outdoor) at desired paces and duration. The overall walked steps were manually counted. This type of test should highlight any possible relationship between the accuracy of the algorithm and the user's specific walking style.

Table 4.5 resumes the duration of the tests for each subject. Acquisitions were performed by using the MuSe device in log mode: acceleration data were collected at a frequency rate of 25 Hz, stored in the on-board memory and successively downloaded via Bluetooth by means of a C#-based software.

Table 4.5 – Summary of the duration of the tests performed during the step detection protocol.

Subj. ID	Walked minutes		
	3 km/h	**6 km/h**	**Free**
1	20	20	-
2	20	20	-
3	20	20	-
4	20	20	48.42
Total	80	80	48.42

Analysis Procedure

The step counting algorithm described hereafter is based on motion data, specifically on 3D acceleration collected by the wearable platform. Although complete acquisitions were available, the algorithm was designed to work in real time conditions as if it were embedded in a device's microcontroller: therefore, it was enclosed in a process which simulated the continuous stream of acceleration samples. Figure 4.13 depicts the flowchart of the developed algorithm: the bold blocks represent the three subroutines, which will be described in further details below.

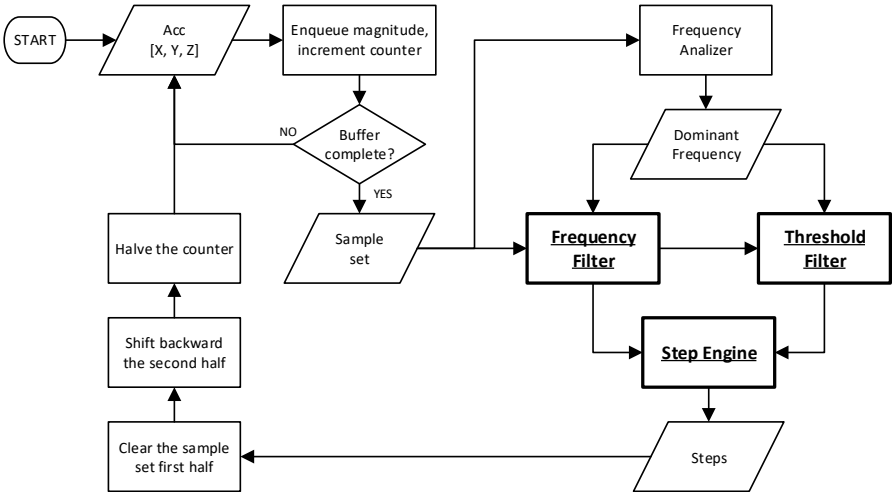


Figure 4.13 – Flowchart of the step detection algorithm.

In order to be independent of the device orientation, data from the three axis is used to compute the acceleration magnitude (or *modulus*). In a real-time usage, the user might want to know at any time of the day how many steps were taken that far. Since the need of a periodic update, the step identification procedure is based on the analysis of samples within a sliding window: in this sense, new samples coming from the real-time simulation process are stocked in a temporary buffer, and the resulting sample set is analyzed at once as soon as it is complete. The size of the buffer was set to 256 samples, in order (1) to frequently provide the user with an updated number of steps, and (2) to optimize the next analysis phase. Each sample set is built so to include a 50 % overlap of the previous one (i.e. the last 128 samples of the previous sample set), in order not to lose any dynamics between consecutive sets. As such, since a new sample set is available at each new 128 samples (after the initial 256), each analysis is performed on the old 128 samples and the new 128 ones, providing a step update every 5.12 s.

Frequency Filter The first subroutine involves an adaptive filtering of the acceleration modulus. The usage of a fixed low threshold frequency, such as 2.5 Hz (i.e.

2.5 steps per second), might enable the successful detection of most walking steps, but it might prevent the detection of running steps which generally occur at a higher frequency (e.g. 2.5–4.5 Hz). For this reason, the filter passband is adaptively selected according to the characteristics of the current sample set.

Firstly, an FFT algorithm is used to compute the frequency representation of the 0 g-mean acceleration modulus. Then, the peak detection procedure described in section 3.3.1 is used to determine the dominant frequency of the current sample set, based on the two highest peaks exceeding an empiric threshold of 0.01 g:

- The dominant frequency is set to 0 if no peaks satisfying the minimum height requirement are found, or alternatively, if only one peak is identified but its frequency location is below 0.2 Hz (i.e. one step every 5 seconds);
- In case two peaks are correctly identified, the dominant frequency is set as the location of the highest peak, or to the lowest of the two locations if the ratio between peaks exceeds a threshold of 75 %. Such verification should account for cases in which the first harmonic is higher than the fundamental frequency due to noise summation.

For a non-zero dominant frequency, a second verification is performed to ensure that the detected value is associated with an actual acceleration signal. At this purpose, the ratio of the signal power in a ±0.2 Hz frequency band around the dominant frequency to the overall signal power is computed and compared to a threshold determined empirically (0.2 %): in case the power of the dominant frequency band is only a small fraction of the signal total power, such frequency is discarded and set to zero.

Finally, once the dominant frequency is successfully assessed, it is used to select one among five predefined, overlapping frequency bands: 0.2–2 Hz, 2–3 Hz, 2.5–3.5 Hz, 3–4 Hz and 3.5–4.5 Hz. During the initialization stage, an elliptic band-pass filter is built for each of the previous bands: in this sense, the dominant frequency is used to adaptively select the most suitable filter, in order to highlight the walking signal within the acceleration modulus (as depicted in Figure 4.14).

Threshold Filter After the frequency analysis and filtering, the second subroutine composing the algorithm involves a threshold filter to be applied to the output of the previous phase. Threshold-cross based step detection techniques, in fact, conventionally involve detecting a step based on acceleration swing about a fixed threshold (e.g. 1 g). However, by using a fixed threshold, some steps may not be detected due to shifts in the acceleration modulus: for instance, when the user is

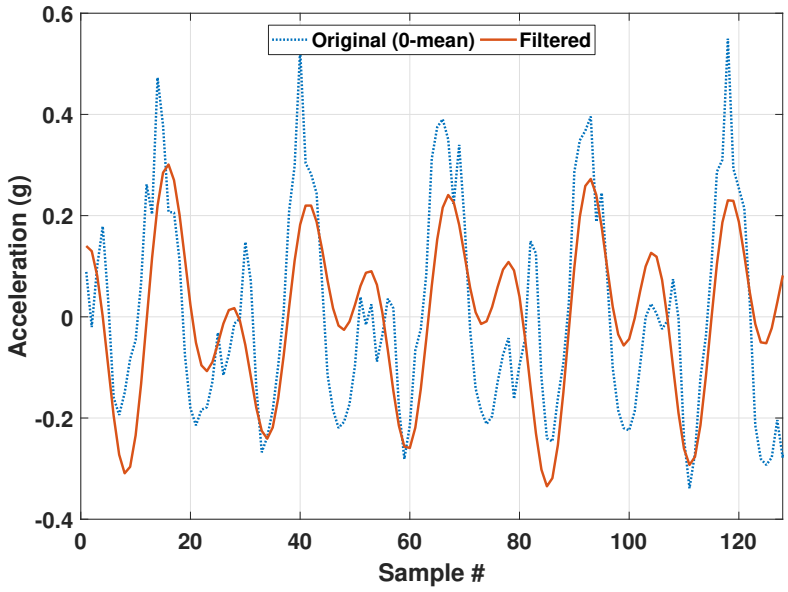


Figure 4.14 – Filtering action of the elliptic band-pass filter.

running the device is subject to higher accelerations. For this reason, an adaptive threshold is computed as the moving average of the filtered acceleration modulus. The size of the moving average window is big enough to include an entire period of the signal, according to the dominant frequency previously determined.

Step Engine The third and last subroutine of the step detection algorithm is called *Step Engine*: it was realized as a decision tree, able to identify valid steps on the basis of several rules. The input variables are the filtered acceleration modulus and the adaptive threshold (moving average), whereas the output is the number of steps associated with the current sample set. Due to the overlapping windows, however, only steps identified in the second half of the set are actually considered as such, since those falling in the first half should have already been counted during the analysis of the previous sample set. An exception is made for the very first window (for obvious reasons), and for windows subsequent to sample set characterized by a null dominant frequency: in the latter case, in fact, no step identification was actually conducted on such window, and thus the first half of the current one may include steps that were never counted before.

A peak detection algorithm is firstly used to find positive and negative peaks in the filtered acceleration signal: this information is successively used by the Step Engine to improve the step identification. Although the peak detection algorithm described in section 3.3.1 is efficient, it is only able to detect positive peaks: negative ones (valleys) can be determined by reversing the signal and re-running the algorithm, but this would come with an overall increase in computation time and/or resources

due to the need of two runs. Therefore, the so-called Todd-Andrews peak detection algorithm was selected instead, since it detects both peaks and valleys in one run only. The algorithm accepts a threshold as an input parameter, representing how much a local maximum (minimum) has to exceed the previous samples to be considered an true peak (valley): regarding its usage in the Step Engine, such parameter was set to 0.05 g.

As already anticipated, common step detection techniques count a new step each time the signal crosses a fixed threshold. The Step Engine was designed to behave in similar way: the filtered signal crossing the adaptive threshold in a specific direction (in this case, downward) is considered a step. For this reason, the input signals are scanned to find threshold crossings, and a set of rules are applied as soon as a new crossing is detected in order to account for any lost step. Figure 4.15 depicts an example of Step Engine's input signals, as well as the detected peaks, valleys and crossings: the author encourages the reader to refer to this figure to better understand how the algorithm works, since it embeds all of the possible scenarios subsequently described.

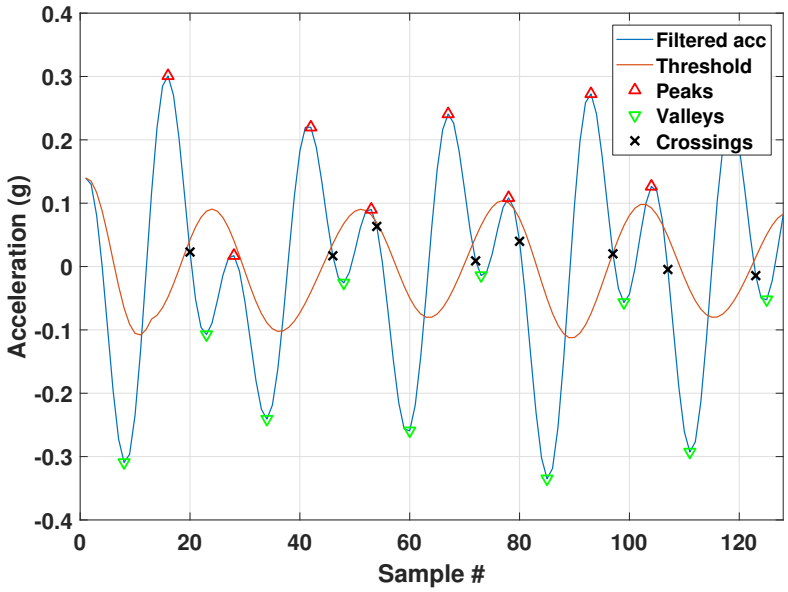


Figure 4.15 – Step Engine's input signals, where peaks, valleys and threshold crossings are highlighted.

To avoid low, noise-related acceleration swings to be considered as steps, a crossing is labeled as "valid" or "invalid" on the basis of whether the previous peak-to-peak amplitude exceeds a threshold of 0.2 g. While the detection of an invalid crossing causes the Step Engine to proceed with the search of the next one, a valid crossing immediately counts as a step and activates the check of finer rules, so to determine whether or not any invalid crossing identified before the current one

represents one (or more) lost step. The first rule to be assessed is the time elapsed between the just-labeled valid crossing and the last valid one. If such interval is less than the fastest walking rate for two steps (in this study, considered as 0.6 s), it is determined that the user is running and thus no other steps are counted. Instead, if the amount of time is greater than such threshold, new steps may be inferred on the basis of subsequent rules outcomes. Three different sequences may occur while considering the last three crossings (one of them being the newest one), on the basis of their label: the following list reports the possible sequences and how the algorithm deals with them.

Invalid-invalid-valid sequence The newest crossing is valid, but the preceding two are not. The last valid crossing prior to the current one is too distant, thus no extra steps can be inferred.

Valid-invalid-valid sequence An invalid crossing lays between two valid crossings. For instance, this may occur if an acceleration swing is slightly lower than the peak-to-peak amplitude threshold of 0.2 g. In this case, the threshold is temporarily halved and the peak-to-peak amplitude is compared to the new value: if the rule now applies, an additional step can be counted, the crossing is marked as valid and the Step Engine can proceed and search the next one; otherwise, the *X-valid-valid sequence* is triggered, as described below.

X-valid-valid sequence Two consecutive valid crossings are found (the oldest may be either valid or not). In this case, since the time elapsed between them is greater than 0.6 s, one crossing (and thus, the associated step) might have been lost. For instance, this may occur when an acceleration swing is affected by a temporary offset which prevents the threshold crossing. In this case, the number of peaks between the two valid crossings is counted:

- If there are two peaks between the valid crossings, i.e. the expected one (which generates the current crossing) and an additional one, then an acceleration swing did not cross the adaptive moving average: as such, a new step is added to the counter.
- If only one peak (the expected one) lays between the valid crossings, a new step may be inferred based on the time elapsed between them. In fact, during a walk the step rate is approximately constant, and within each sample set it is represented by the dominant frequency determined previously. Thus, according to it, if one step was expected between two steps (valid crossings) but it was not identified, a step can be inferred if the elapsed time between the two consecutive crossings approximates

the step period (inverse of the dominant frequency), with a margin of ±20 %. For instance, if the walking rate of a sample set is 1 Hz (one step per second) and two consecutive steps are distant 1.9 s, it is likely that a step may have been missed, thus the counter is incremented.

Results and Discussion

More than 16 000 steps were walked during the execution of structured tests, according to the previously described protocol (two different speeds, fixed duration): Table 4.6 reports the results of the steps detection algorithm run on accelerations collected by the wrist-worn device. For each step number estimation, the Relative Error and the Absolute Relative Error with respect to visually measurements were computed as follows:

$$\text{Relative Error} = \frac{\tilde{N} - N}{N} \cdot 100, \tag{4.3}$$

$$\text{Absolute Relative Error} = \frac{\left|\tilde{N} - N\right|}{N} \cdot 100, \tag{4.4}$$

where N and $\tilde{N}$ represent the actual number of walked steps and its estimated value, respectively.

Table 4.6 – Measured and estimated steps in structured tests. For each test, the Relative Error (RE) in percentage is reported.

Subj. ID	3 km/h			6 km/h		
	Meas.	Est.	RE %	Meas.	Est.	RE %
1	788	771	−2.16	1150	918	−20.17
	831	863	3.85	1169	1046	−10.52
2	749	751	0.27	1168	1163	−0.43
	780	780	0	1169	1164	−0.43
3	814	762	−6.39	1127	1125	−0.18
	821	905	10.23	1129	1130	0.09
4	894	943	5.48	1256	1234	−1.75
	913	1020	−11.72	1260	1093	−13.25

Results of the structured tests reveals that the developed algorithm performs well both during slow and fast walking, with Absolute Relative Errors largely below 20 % in most cases. The accuracy of the step detection seems to be dependent on the specific user walking style: subject 1, for instance, is characterized by lower absolute relative errors when walking at low speed, with respect to absolute errors associated with a faster pace; subject 3, on the contrary, exhibits the opposite behavior.

The average Absolute Relative Error is 5.01 % (SD: 4.33 %) and 5.85 % (SD: 7.77 %) for low and high walking rates, respectively. There seems to be a tendency to slightly overestimate steps at slow speeds; on the contrary, at faster speeds the algorithm is more inclined to underestimate steps: such behavior can be clearly observed in Figure 4.16, which depicts the distributions of the relative errors, grouped by speed.

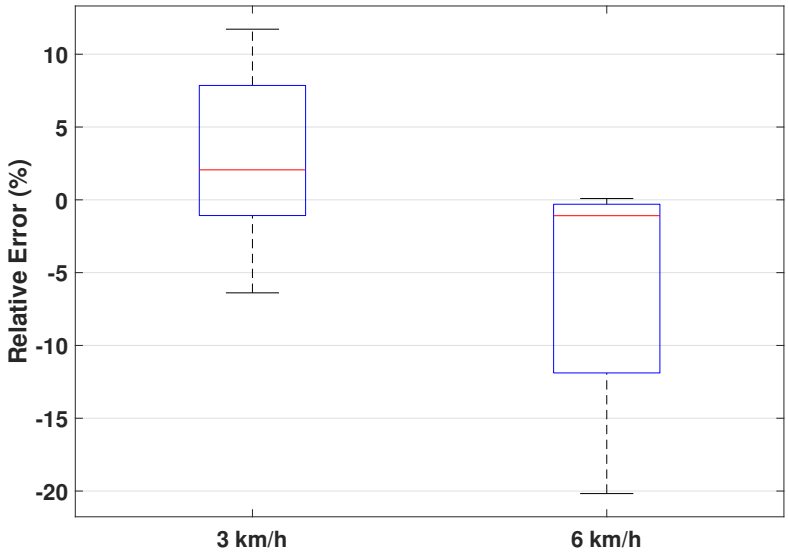


Figure 4.16 – Box plots of the relative errors in structured tests, grouped by speed.

Unstructured tests were carried out by the user for which the algorithm performed worse at both speeds in structured tests (i.e. subject 4). Table 4.7 lists the estimated steps and the relative errors for each of the performed acquisitions. Results are characterized by a higher variability with respect to the structured ones, nevertheless the Absolute Relative Errors still fall below 20 %: the mean value is 6.80 % (SD: 6.34 %), which is comparable to the results of structured tests for this subject (8.60 % ARE for low speed, 7.50 % ARE for high speed). In this sense, the algorithm seems to perform in similar way both in supervised and unsupervised conditions.

Finally, the algorithm was tested on accelerations related to "not worn", "resting" and "working at PC" activities, collected for the physical activity monitoring described in section 4.5.1. Since no steps were performed during such activities, the goal of this test was to determine the amount of false positives identified by the algorithm in absence of a real walk. This kind of evaluation is of fundamental importance if considering that the current system (composed of a wearable device and the developed step detection algorithm) is not going to be used on fitness users, but rather on patients affected by diseases impairing their physical conditions: it is more likely, thus, that they are not prone to a very active lifestyle. In this

Table 4.7 – Measured and estimated steps in unstructured tests. For each test, the Relative Error (RE) in percentage is reported.

Session #	Measured	Estimated	RE %
1	108	109	0.93
2	120	112	−6.67
3	120	122	1.67
4	123	119	−3.25
5	131	127	−3.05
6	140	115	−17.86
7	140	136	−2.86
8	140	149	6.43
9	148	125	−15.54
10	150	158	5.33
11	282	240	−14.89
12	500	411	−17.80
13	1000	973	−2.70
14	1000	1015	1.50
15	1000	1015	1.50

sense, about 1300 minutes of the previously cited activities (divided in different acquisitions) underwent the developed step detection algorithm, and its false positive rate was expressed in terms of number of estimated steps per minute against a measured value of zero. Surprisingly, results of such analysis reveals that the algorithm is robust enough to detect 0 steps in every acquisition of such activity categories: this result needs to be further investigated, in order to confirm or reject the hypothesis of 100 % specificity of such algorithm when used in no-walking conditions.

In conclusion, the preliminary validation demonstrated that the performance of the developed step detection algorithm can potentially resemble that of the current commercial pedometers. Having a maximum estimation error always below 20 %, the algorithm could be ranked between the Jawbone Up and the Nike+ Fuelband devices according to Storm et al. [33]; nevertheless, no information about false positive rate is available for such commercial platforms: in this sense, the current algorithm might even outperform them.

Further investigations will be performed in the next future. The sample size is going to be enlarged by including new users of different sex and ages. More importantly, the algorithm needs to be tested on patients affected by multiple sclerosis in order to be considered as a valid instrument for physical assessment, as required by the RADAR-MS study. Once a first fine-tuned version of the algorithm will be released, it may be embedded in a new wearable platform purposely designed for research usage, as will be discussed in the next chapter.

References

[1] J. A. Naslund, L. A. Marsch, G. J. McHugo, and S. J. Bartels, "Emerging mHealth and eHealth interventions for serious mental illness: a review of the literature", *Journal of Mental Health*, pp. 1–12, 2015. DOI: 10.3109/09638237.2015.1019054.

[2] A. Gaggioli and G. Riva, "From mobile mental health to mobile wellbeing: Opportunities and challenges", *Studies in Health Technology and Informatics*, vol. 184, pp. 141–7, 2013. DOI: 10.3233/978-1-61499-209-7-141.

[3] M. Olff, "Mobile mental health: A challenging research agenda", *European journal of psychotraumatology*, vol. 6, p. 27 882, May 2015. DOI: 10.3402/ejpt.v6.27882.

[4] Y.-G. Lee, W. Sig Jeong, and G. Yoon, "Smartphone-Based Mobile Health Monitoring", *Telemedicine journal and e-health : the official journal of the American Telemedicine Association*, vol. 18, pp. 585–90, Oct. 2012. DOI: 10.1089/tmj.2011.0245.

[5] RADAR-base community. (2018). RADAR-base Wiki, [Online]. Available: https://radar-base.atlassian.net/wiki/spaces/RAD/overview.

[6] Joris Borgdorff. (2017). Wearables on the radar, [Online]. Available: http://blog.thehyve.nl/blog/wearables-on-the-radar.

[7] (2018). REDCap project, [Online]. Available: https://www.project-redcap.org/.

[8] Caroline Langen. (2017). RADAR-CNS Annual Meeting, [Online]. Available: http://blog.thehyve.nl/blog/radar-cns-annual-meeting.

[9] National Multiple Sclerosis Society. (2018). What Is MS?, [Online]. Available: https://www.nationalmssociety.org/What-is-MS.

[10] J. Olesen, A. Gustavsson, M. Svensson, H. U. Wittchen, and B. Jönsson, "The economic cost of brain disorders in Europe", *European Journal of Neurology*, 2012. DOI: 10.1111/j.1468-1331.2011.03590.x.

[11] J. Kurtzke, "Rating neurologic impairment in multiple sclerosis: an expanded disability status scale (EDSS)", *Neurology*, 1983.

[12] M. D. Goldman, R. W. Motl, and R. A. Rudick, "Possible clinical outcome measures for clinical trials in patients with multiple sclerosis", *Therapeutic Advances in Neurological Disorders*, 2010. DOI: 10.1177/1756285610374117.

[13] C. Hessen, J. Bohm, and C. Reich, "Patient perception of bodily functions in multiple sclerosis: Gait and visual function are the most valuable", *Multiple Sclerosis Journal*, pp. 988–991, 2008. DOI: 10.1177/1352458508088916.

[14] J. A. Cohen, S. C. Reingold, C. Polman, *et al.*, "Disability outcome measures in multiple sclerosis clinical trials: Current status and future prospect", *The Lancet Neurology*, vol. 5, pp. 467–76, 2012. DOI: 10.1016/S1474-4422(12)70059-5.

[15] J. C. Hobart, A. Riazi, D. L. Lamping, *et al.*, "Measuring the impact of ms on walking ability: The 12-item ms walking scale (msws-12).", *American Academy of Neurology Journals*, 2003.

[16] Multiple Sclerosis Clinical Practice Guidelines, *Fatigue and multiple sclerosis: evidence-based management strategies for fatigue in multiple sclerosis*. Paralyzed Veterans of America, 1998.

[17] J. A. Cohen, S. C. Reingold, C. Polman, *et al.*, "The impact of fatigue on patients with multiple sclerosis", *Canadian Journal of Neurological Sciences*, 1994.

[18] L. B. Krupp, N. G. LaRocca, J. Muir-Nash, and A. D. Steinberg, "The fatigue severity scale: Application to patients with multiple sclerosis and systemic lupus erythematosus", *Archives of Neurology*, vol. 46, no. 10, pp. 1121–1123, 1989. DOI: 10.1001/archneur.1989.00520460115022.

[19] F. Gholami, D. A. Trojan, J. Kövecses, W. M. Haddad, and B. Gholami, "Gait Assessment for Multiple Sclerosis Patients Using Microsoft Kinect", *CoRR*, vol. abs/1508.02405, 2015.

[20] A. Qureshi, M. Brandt-Pearce, and M. D. Goldman, "Relationship between gait variables and domains of neurologic dysfunction in multiple sclerosis using six-minute walk test", *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 4959–4962, 2016.

[21] M. M. Engelhard, S. R. Dandu, S. D. Patek, J. C. Lach, and M. D. Goldman, "Quantifying six-minute walk induced gait deterioration with inertial sensors in multiple sclerosis subjects", *Gait & Posture*, vol. 49, pp. 340–345, 2016. DOI: 10.1016/j.gaitpost.2016.07.184.

[22] D. Podsiadlo and S. Richardson, "The timed "up & go": A test of basic functional mobility for frail elderly persons", *Journal of the American geriatrics Society*, vol. 39, no. 2, pp. 142–148, 1991.

[23] R. Spain, R. S. George, A. Salarian, M. Mancini, J. Wagner, F. Horak, and D. Bourdette, "Body-worn motion sensors detect balance and gait deficits in people with multiple sclerosis who have normal walking speed", *Gait & Posture*, vol. 35, no. 4, pp. 573–578, 2012. DOI: 10.1016/j.gaitpost.2011.11.026.

[24] B. R. Greene, M. Healy, S. Rutledge, B. Caulfield, and N. Tubridy, "Quantitative assessment of multiple sclerosis using inertial sensors and the tug test", in *2014 36th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2014, pp. 2977–2980. DOI: 10.1109/EMBC.2014.6944248.

[25] B. R. Greene, S. Rutledge, I. McGurgan, C. McGuigan, K. O'Connell, B. Caulfield, and N. Tubridy, "Assessment and classification of early-stage multiple sclerosis with inertial sensors: Comparison against clinical measures of disease state", *IEEE journal of biomedical and health informatics*, vol. 19, no. 4, pp. 1356–1361, 2015. DOI: 10.1109/JBHI.2015.2435057.

[26] F. Grenez, M. V. Villarejo, B. G. Zapirain, and A. M. Zorrilla, "Wireless prototype based on pressure and bending sensors for measuring gate quality", *Sensors*, vol. 13, no. 8, pp. 9679–9703, 2013. DOI: 10.3390/s130809679.

[27] A. N. Kurti and J. Dallery, "Internet-based contingency management increases walking in sedentary adults", *Journal of Applied Behavior Analysis*, vol. 46, pp. 568–581, 2013. DOI: 10.1002/jaba.58.

[28] B. DM, S.-S. C, S. V, and et al, "Using pedometers to increase physical activity and improve health: A systematic review", *JAMA*, vol. 298, no. 19, pp. 2296–2304, 2007. DOI: 10.1001/jama.298.19.2296.

[29] C. B. Thorup, J. J. Andreasen, E. E. Sørensen, M. Grønkjær, B. I. Dinesen, and J. Hansen, "Accuracy of a step counter during treadmill and daily life walking by healthy adults and patients with cardiac disease", *BMJ Open*, vol. 7, no. 3, 2017. DOI: 10.1136/bmjopen-2016-011742.

[30] M. G. Abel, N. Peritore, R. Shapiro, D. R. Mullineaux, K. Rodriguez, and J. C. Hannon, "A comprehensive evaluation of motion sensor step-counting error", *Applied Physiology, Nutrition, and Metabolism*, vol. 36, no. 1, pp. 166–170, 2011. DOI: 10.1139/H10-095.

[31] P. L. Schneider, S. E. Crouter, O. Lukajic, and D. R. J. Bassett, "Accuracy and Reliability of 10 Pedometers for Measuring Steps over a 400-m Walk", *Medicine & Science in Sports & Exercise*, vol. 35, no. 10, pp. 1779–1784, 2003. DOI: 10.1249/01.MSS.0000089342.96098.C4.

[32] P. L. Schneider, S. E. Crouter, and D. R. J. Bassett, "Pedometer Measures of Free-Living Physical Activity: Comparison of 13 Models", *Medicine & Science in Sports & Exercise*, vol. 36, no. 2, pp. 331–335, 2004. DOI: 10.1249/01.MSS.0000113486.60548.E9.

[33] F. A. Storm, B. W. Heller, and C. Mazzà, "Step Detection and Activity Recognition Accuracy of Seven Physical Activity Monitors", *PLoS One*, vol. 10, no. 3, 2015. DOI: 10.1371/journal.pone.0118723.

Chapter 5

The Winter Platform

The studies reported in the previous chapters of this thesis relied on the use of different inertial platforms to monitor several aspects of motor disabilities caused by CNS diseases. Those platforms were either consumer-grade devices, such as the Fitbit Charge 2 and the eMotion Faros 180 used in chapter 4, or research-grade devices, such as the MuSe platform used in chapters 2 and 3. With regard to the tasks for which they were selected, such platforms were among the best choices and proved to be functional to the objective of their usage (i.e. vibration measurements, physical activity monitoring). Nevertheless, the selected devices are affected by some major limitations that may prevent their usage in other contexts.

- The Fitbit device makes use of proprietary algorithms and provides only the results of the analysis process, such as the step number or the mean heart rate per time window. Since neither the processing methods, nor (more importantly) the raw data are accessible, it is impossible to develop and test any new custom algorithms based on data collected by the device.

- Unlike Fitbit, the eMotion Faros provides access to raw data, but lacks an important inertial sensor such as the gyroscope. As such, this device may not be the optimal choice when the goal is to monitor the movement of a subject.

- The MuSe platform is not affected by the previous limitations: not only it integrates the most important tracking sensors (including the gyroscope), but also it provides both raw and processed data. Nevertheless, the platform was mainly designed for a usage in supervised contexts: its storage capacity and battery lifetime allow for short acquisition sessions only (few hours at most).

This chapter describes the development of a new inertial platform named *Winter*, designed specifically to account for these limitations and to potentially replace the used platforms.

5.1 Design Strategies

The experience gained from the studies described in the previous chapters led to the design of a new platform, able to overcome the limitations observed during the use of the selected wearable devices. *Winter*, acronym of Wearable INertial TrackER, is the result of a design aimed to provide a wireless system-on-board with inertial-based processing capabilities, low power consumption and high storage capacity, all in a small form factor. This wearable platform is primarily thought to be used as an activity tracker, similar to current consumer-grade commercial platforms (Fitbit, Jawbone, Polar, etc.). However, the chosen integrated inertial sensors and the possibility to perform long-term acquisitions and to access raw data, make it a perfect solution for research purposes too: to this respect, it can be considered as an alternative to the MuSe platform adopted in previous studies.

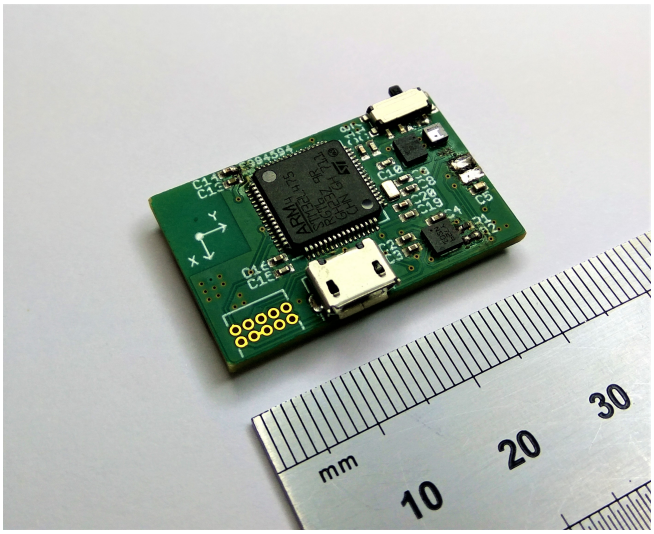


Figure 5.1 – The Winter platform. The scale is reported in millimeters.

Raw data can be stored into a MicroSD card mounted on-board: with respect to the MuSe integrated flash memory, this storage technology allows the acquisition of raw inertial data for multiple days consecutively. Data can then be transferred directly through the removable SD card, without the need of an active radio connection. In addition, Winter is provided with a Bluetooth Low Energy v4.1 module, characterized by low power consumption and designed mainly for sending and receiving short pieces of data (known as attributes). This wireless communication technology will be only used to configure the platform, and to provide results of the analysis algorithms embedded within the platform's firmware and run by the microcontroller: in this way, only short active radio connections will be needed, and thus the supply power required by the module will be minimized.

Winter is designed to be an *always-on platform*, reflecting the standard working conditions of the consumer-grade activity trackers. This means that even when the device is not worn, the system is always powered on and keeps draining current from the battery: it will be up to the hardware-embedded, sensing functionalities to recognize whether the device is being used or not, and thus to switch from a low-power consumption state (either *stop* or *sleep* mode) to the *run* mode (data collection, analysis and storage). Alternatively, the platform can be awakened by an active Bluetooth connection and set to operate as a data logger until an external stop input. However, to preserve the battery life even with this feature implemented, the system power consumption needs to be as low as possible: to this purpose, Winter is provided with ultra-low power components, whose current consumptions are summarized in Table 5.1. To support the choice of the hardware components, the firmware as well as the finite-state machine will need to be oriented to the minimization of the device's power requirement. No physical buttons are present on the board. Instead, the user is allowed to communicate with Winter either by an active Bluetooth connection, or by exploiting the event detection procedures provided by the inertial module (such as click and double-click functions).

Table 5.1 – Typical power consumption of the Winter active components. Data were provided by the respective datasheets.

Name	**Description**	**Current consumption**	**Operative conditions**
STM32L475R	MCU + FPU	0.33 µA	Shutdown w/ RTC
		Up to 0.65 µA	Standby w/ RTC
		1.4–108 µA	Stop mode (2, 1, 0)
		35–40 µA/MHz	Sleep, LPSleep
		100–136 µA/MHz	Run, LPRun
LSM6DSL	Inertial module	3 µA	Power-down
		0.29 mA	Low-power mode
		0.45 mA	Normal mode
		0.65 mA	High-performance mode
HTS221	Humidity and temperature sensors	0.5 µA	Power-down mode
		2 µA	ODR 1 Hz, 25 °C, 2.5 V
SPBTLE-RF	Bluetooth LE v4.1	2 µA	Power-down/Standby
		0.99 mA	Advertising (peripheral)
		0.12 mA	Connection (peripheral)
STNS01	Battery charger + LDO	400 µA	Charger disabled
		1.4 mA	Charging
MAX17048X+	Fuel Gauge	2 µA	Sleep mode
		5 µA	Hibernate mode
		40 µA	Run mode

Winter's battery can be charged through the Micro-USB 2.0 port mounted on the platform: however, this is not the only function of such component. Besides the on-board sensors, in fact, Winter is designed to potentially communicate and retrieve data from low-power, battery-less expansion boards. Such sensor-equipped peripherals would allow to enhance the sensing capability of the Winter device: regarding the tremor classification for instance, the parallel monitoring of wrist and fingers movements might provide useful additional information about some particular types of tremor, thus enabling the differentiation of other tremor-associated diseases. Because such peripherals would not come with a full power management system (no dedicated batteries to minimize the boards size), besides the data collection Winter will have to provide them the needed voltage supply. To this respect, the configuration of the Micro-USB port lines is purposely designed to enable data transfer and power sharing with an external device: in particular, the first feature is available thanks to the USB on-the-go (OTG) serial bus provided by the microcontroller.

5.2 Schematics

The circuit diagrams (or schematics) of Winter were designed by using the schematic editor provided by EAGLE 8.0 software.

5.2.1 Processing

The core processing unit of Winter is the STM32L475RG (Figure 5.2), an ultra-low-power microcontroller based on the high-performance Arm Cortex-M4 32-bit core operating at a frequency of up to 80 MHz. The Cortex-M4 core features a Floating point unit (FPU) single precision which supports all Arm single-precision data-processing instructions and data types. It implements a full set of DSP instructions and a memory protection unit (MPU) which enhances application security [1].

The STM32L475RG device embeds high-speed memories (1 Mbyte of Flash memory, 128 Kbyte of SRAM), a Quad SPI flash memories interface, a low-power RTC and an extensive range of enhanced I/Os and peripherals. It also features standard and advanced communication interfaces, among which: three I^2Cs, three SPIs, three USARTs, an SDMMC and a USB OTG full-speed.

Four specific I/O lines can be configured to work as wakeup pin. To this purpose, two of them (namely PA0 and PA2) are connected to a programmable interrupt line of the LSM6DSL and to the IRQ pin of the Bluetooth module, respectively: in this way, the microcontroller can be awakened either when an external physical

input occurs, or when a radio connection is active.

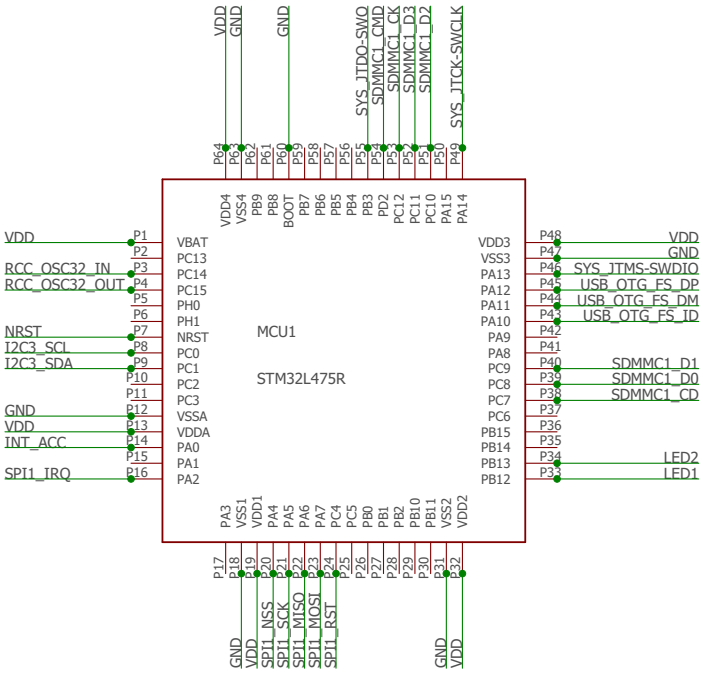


Figure 5.2 – STM32L475R microcontroller.

The system clock used by the embedded RTC is generated internally by the STM32L475RG. However, in order to obtain a more accurate and precise clock, an external crystal is connected to the microcontroller, specifically the ABS06-1-T SMD crystal provided by Abracon (Figure 5.3). This component generates a clock signal of 32.768 kHz with a frequency tolerance of ±10 ppm (parts per million) [2].

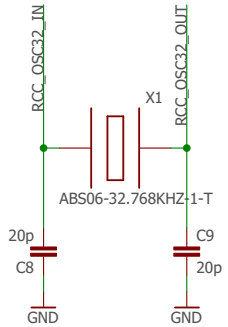


Figure 5.3 – ABS06-1-T crystal.

5.2.2 Sensing

The sensing capabilities of Winter are achieved by two modules, connected to the STM32L475RG microcontroller by means of an I^2C communication interface

(Figure 5.4). The first one is the LSM6DSL, a system-in-package provided by STMicroelectronics featuring a 3D digital accelerometer and a 3D digital gyroscope with full-scale ranges of ±2/±4/±8/±16 g and ±125/±245/±500/±1000/±2000 dps, respectively. The module works at 0.65 mA in high-performance mode and enables always-on low-power features [3]. The configurable event-detection interrupts enable efficient and reliable contextual awareness, implementing hardware recognition of free-fall events, 6D orientation, click and double-click sensing, activity or inactivity, and wakeup events: in particular, the click detection will be used to monitor external input and thus to generate a wakeup interrupt to the microcontroller.

The second sensing module is the HTS221 (STMicroelectronics), an ultra-compact sensor for relative humidity (rH) and temperature. It includes a sensing element and a mixed signal ASIC to provide the measurement information through digital serial interfaces [4]. The HTS221 has a relative humidity range of 0–100 % (accuracy: ±3.5 % in the 20–80 % rH interval), and a temperature range from −40 to 120 °C (accuracy: ±0.5 °C in the 15–40 °C interval). When the module operates at 1 Hz ODR, the power consumption is 2 µA.

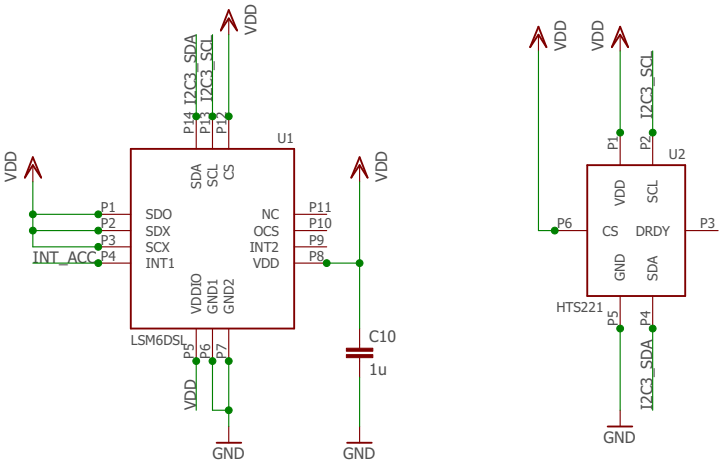


Figure 5.4 – LSM6DSL and HTS221 modules.

5.2.3 Storage

The Adafruit MicroSD card socket mounted on the Winter platform (Figure 5.5) allows for the usage of an external memory card to store raw data and processed results. The data transfer is performed by using the 4-line SDMMC communication interface provided by the microcontroller, at a rate up to 48 MHz (8-bit mode).

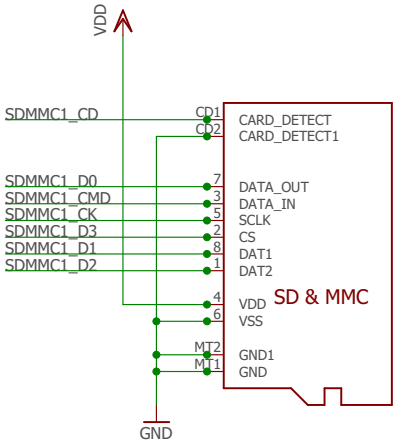


Figure 5.5 – MicroSD card socket.

5.2.4 Connectivity

The communication with external devices (such as smartphones and PCs) allows Winter (1) to transfer the output of the embedded data analysis algorithms, and (2) to be remotely configured to operate in one of the available working modes. To enable such communication, two different solutions were adopted.

The first solution is represented by the SPBTLE-RF (Figure 5.6), an easy-to-use Bluetooth low energy master/slave network processor module provided by STMicroelectronics, compliant with Bluetooth v4.1. It supports multiple roles simultaneously, and can act at the same time as Bluetooth low energy sensor and hub device. The entire Bluetooth low energy stack and protocols are embedded into SPBTLE-RF module. The microcontroller integrated on the Winter platform is connected to the SPBTLE-RF module through a standard SPI interface [5]. The dedicated SPI IRQ pin is connected to the PA2 microcontroller port to awake the processing unit when needed.

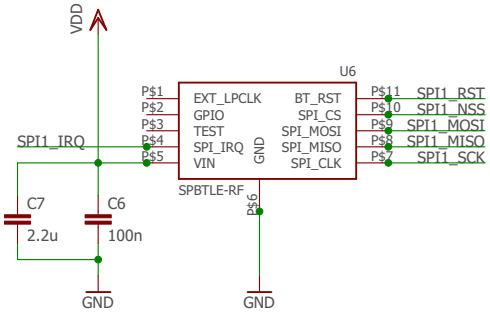


Figure 5.6 – SPBTLE-RF Bluetooth module.

On the other hand, Winter provides a Micro-USB 2.0 high-speed port (Figure 5.7) which allows data transfer with an external device in both directions. In order to prevent damages caused by possible electrostatic discharges, the HSP061-2 was

integrated: it is a 2-channel ESD array with a rail to rail architecture designed specifically for the protection of high speed differential lines [6].

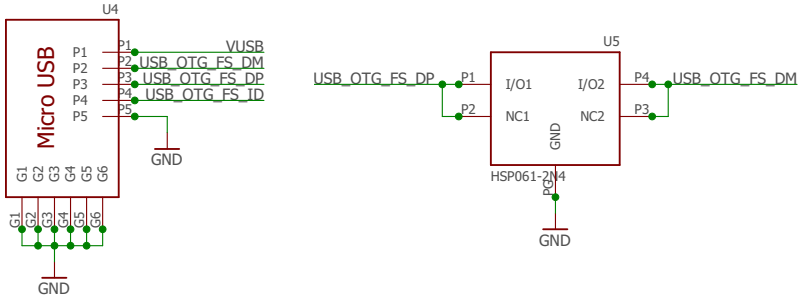


Figure 5.7 – Micro USB connector and HSP061 module.

5.2.5 Power Management

In addition to data, the USB port can be used to transfer power both into and out of the platform, according to the following possible working conditions:

- Directly supply the platform with an external power source;
- Charge the platform's battery;
- Supply a secondary external board with power from the platform's battery.

The first two operational modes are supported by the use of the STNS01 (STMicroelectronics), a linear charger for single-cell Li-Ion batteries integrating an LDO regulator and several battery protection functions (Figure 5.8). The input supply voltage is normally used to charge the battery and provide power to the LDO regulator. When a valid input voltage is not present and the battery is not empty, the device automatically switches to battery power [7].

The third operational mode can be achieved by using the Schottky diode BAT60J in the configuration depicted in Figure 5.8. This diode is characterized by a low forward voltage drop (0.28 V at $I_f = 1\,\text{mA}$, 0.35 V at $I_f = 100\,\text{mA}$) and a very fast switching action. When the device is connected to a power source via the USB port and a valid input voltage is present (VUSB = 5 V), the LDO regulator supply the platform with a voltage VDD = 3.1 V: in this condition, no current I_f flows through the diode since its forward voltage is negative (VDD < VUSB). Instead, when a secondary device is connected to the USB port (VUSB = 0 V) and the platform is supplied by the battery through the LDO regulator (VDD = 3.1 V), a forward current I_f flows through the diode and provides power to the external

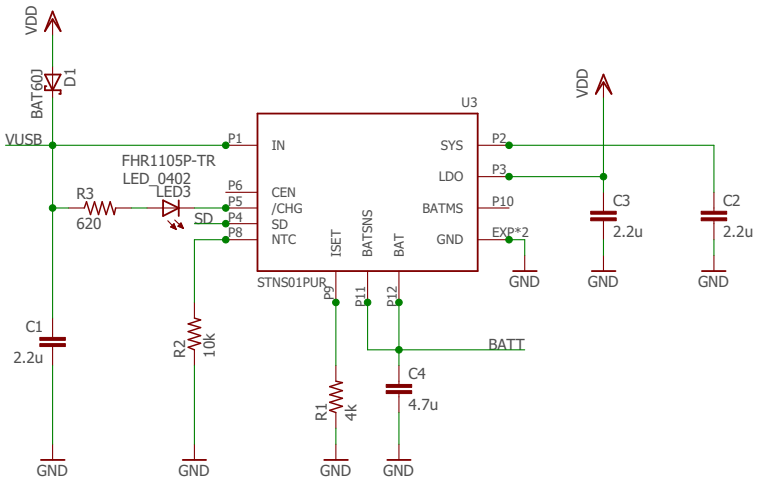


Figure 5.8 – STNS01 battery charger and LDO.

device: thanks to the low voltage drop of the Schottky diode, the resulting input voltage supply for the secondary device approximates VDD.

In order to monitor the battery charging level, the MAX17048X+ was integrated and connected to the microcontroller via I^2C connection interface (Figure 5.9). This IC is a tiny, micropower current fuel gauges for Li-ion batteries in handheld and portable equipment, operating with a single lithium cell. The IC use the sophisticated Li^+ battery-modeling algorithm ModelGauge to track the battery relative state-of-charge (SOC) continuously over widely varying charge and discharge conditions. The ModelGauge algorithm eliminates current-sense resistor and battery-learn cycles required in traditional fuel gauges [8].

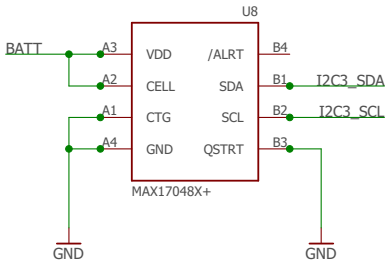


Figure 5.9 – MAX17048X+ fuel gauge module.

5.2.6 I/O

A 10-pin connector and a switch are mounted on the Winter platform for debug purposes, and can be removed at release. Moreover, the board is provided with 3 LEDs: one LED is directly connected to the STNS01 battery charger (see Figure 5.8)

and indicates the charging status of the device, whereas the remaining two are connected to the microcontroller and can be piloted programmatically (Figure 5.10).

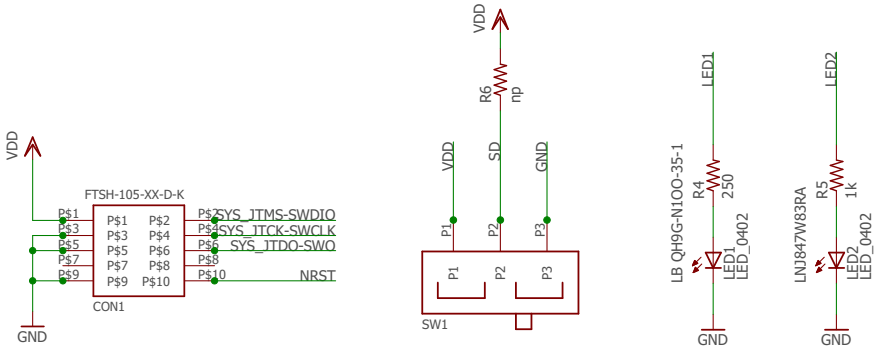


Figure 5.10 – Debug connector, switch and LEDs.

5.3 Layout

In addition to the schematic editor, EAGLE provides a printed circuit board (PCB) layout editor with auto-router and computer-aided manufacturing (CAM) features. Starting from the reported schematics, the layout of Winter was designed by using such software. Specifically, the layout is composed of two layers only, to reduce the production costs: the top layer (Figure 5.11) integrates the processing, the sensing, the power management and the I/O components, whereas the Bluetooth module as well as the SD card socket are mounted on the bottom layer (Figure 5.12). Power supply decoupling capacitors are connected to supply lines of each component.

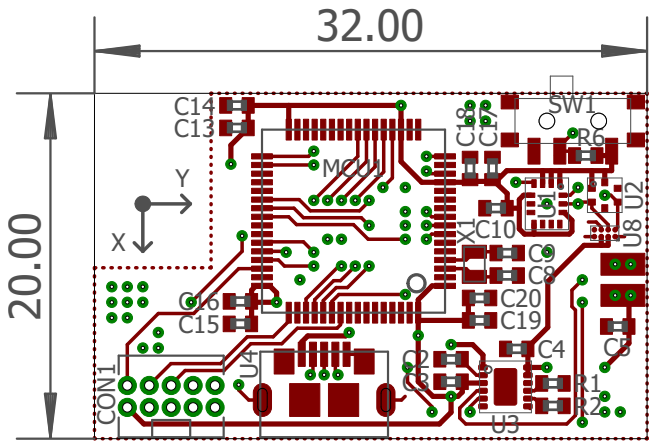


Figure 5.11 – Top view of the Winter layout (units in millimeters).

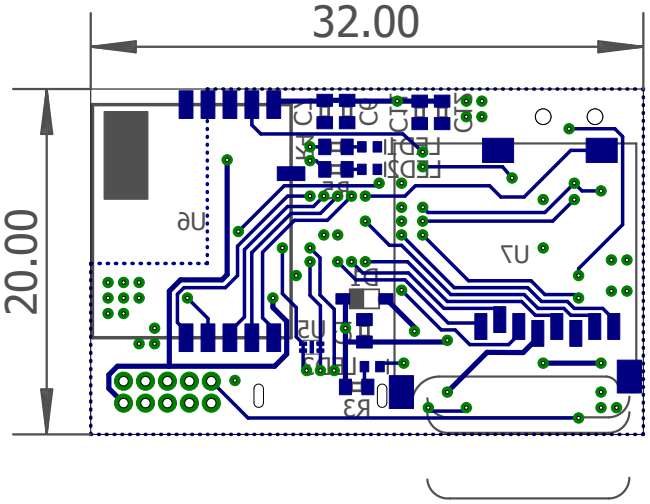


Figure 5.12 – Bottom view of the Winter layout (units in millimeters).

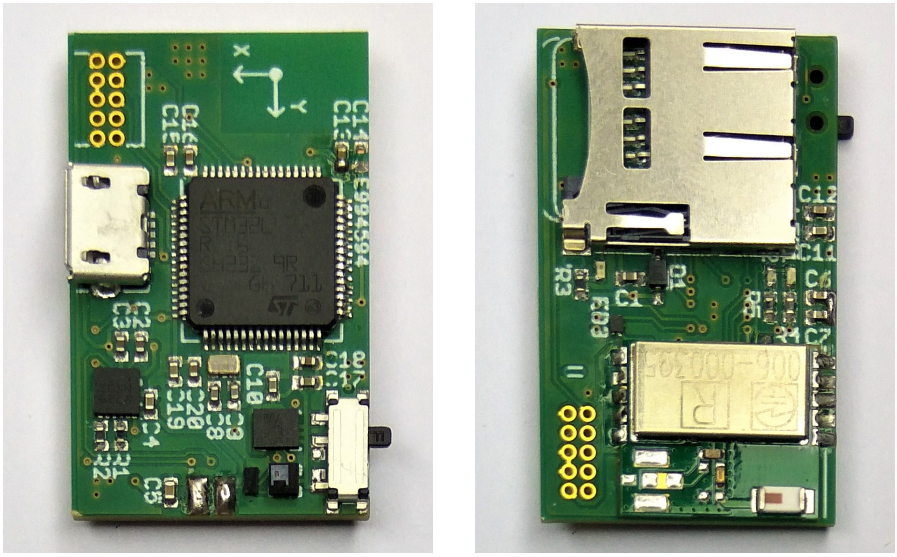


Figure 5.13 – Top and bottom view of the assembled Winter platform.

5.4 Finite State Machine

Based on the applications for which the Winter platform is thought to be used, a *finite-state machine* (FSM) was designed. This mathematical model of computation is an abstract machine that can be in exactly one of a finite number of *states* at any given time. The FSM can change from one state to another in response to some external inputs: the change from one state to another is called a *transition*. Figure 5.14 depicts the first version of the Winter FSM, whereas the states and the transitions are described in Tables 5.2 and 5.3, respectively.

When the board is powered on for the first time, the platform's peripherals are initialized by a startup routine (INIT state). If the initialization succeeds, the platform enters in an IDLE state in which the sensing components are powered

Table 5.2 – List of Winter FSM states. For each state, the microcontroller operational mode and the Bluetooth status is reported.

Name	Description	MPU Mode	BT Status
INIT	Peripherals initialization	Run	Advert/Sleep
SLEEP	Device not in use, system in ultra-low power mode	Sleep	Standby
IDLE	Idle state	LPRun	Advert/Sleep
READY	Device ready to receive BT commands	LPRun	Connection
	***Activity Tracker* mode**		
AT_CONN	Device ready to receive BT commands Raw data collection and analysis ON Raw data storage OFF	LPRun	Connection
AT_WAIT	Device waiting for a BT connection Raw data collection and analysis ON Raw data storage OFF	LPRun	Advert/Sleep
AT_LPL	Device in low power log Raw data collection and analysis ON Raw data storage OFF	LPRun	Standby
	***Data Logger* mode**		
DL_CONN	Device ready to receive BT commands Raw data collection and storage ON Raw data analysis OFF	LPRun	Connection
DL_WAIT	Device waiting for a BT connection Raw data collection and storage ON Raw data analysis OFF	LPRun	Advert/Sleep
DL_LPL	Device in low power log Raw data collection and storage ON Raw data analysis OFF	LPRun	Standby

Table 5.3 – List of Winter FSM state transitions. XX stands for either AT (Activity Tracker mode) or DL (Data Logger mode).

N.	From State	To State	Action
0	OFF	INIT	Device first power-on
1	INIT	IDLE	Peripherals initialized
2	IDLE	SLEEP	Timeout
3	SLEEP	IDLE	Double-click
4	IDLE	READY	BT connection
5	READY	IDLE	BT disconnection
6	READY	READY	Configuration commands
7A–7B	READY (XX_CONN)	XX_CONN (READY)	Mode-related commands
8	XX_CONN	XX_WAIT	BT disconnection
9	XX_WAIT	XX_LPL	Timeout
10	XX_LPL	XX_WAIT	Double-click
11	XX_WAIT	XX_CONN	BT connection

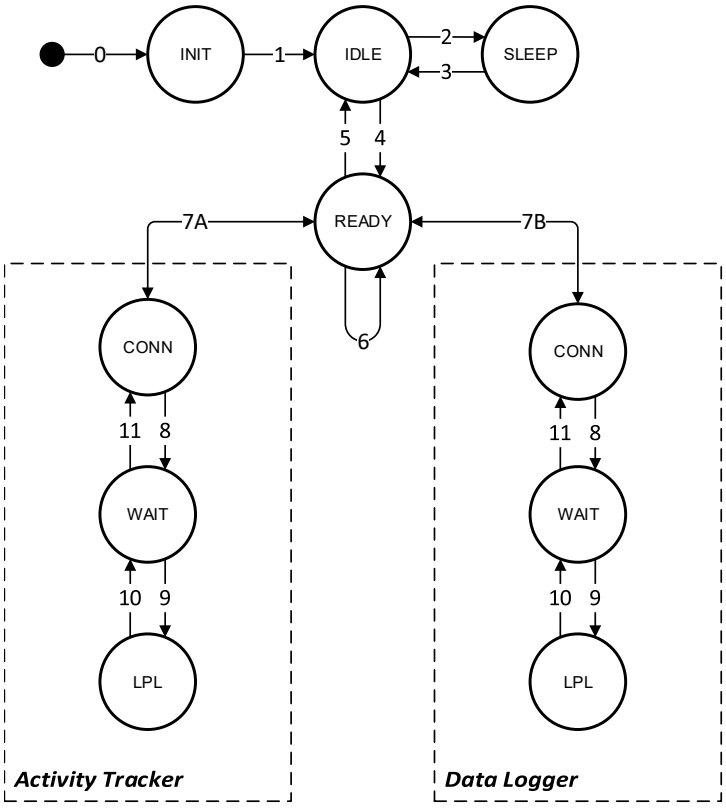


Figure 5.14 – Finite-State Machine of the Winter platform.

down, but the Bluetooth module alternates between a sleep and an advertising state (about every 300 ms) and is thus ready to accept a connection. If this does not happen within a specific time window (i.e. one minute), the system enters in SLEEP state: the power consumption is reduced to the minimum by switching the Bluetooth state to standby mode and by setting the microcontroller to operate in sleep mode. The double-click event detection embedded in the inertial module is used to exit from this low-power state and return to the IDLE state, with the Bluetooth module now operative again.

From the IDLE state, an active Bluetooth connection makes the FSM moving to the READY state. Here, the platform is ready to receive any Bluetooth command from the master device, such as configuration commands (e.g. device name, sensors' full scale and RTC synchronization) and mode-related commands (e.g. operational mode and sampling frequency). The platform returns to the IDLE state as soon as the Bluetooth connection is no longer active.

By means of the mode-related commands, Winter can be set to operate in two different modes:

Activity Tracker (AT) In this mode, the microcontroller continuously collects and analyzes raw data, alternating the Low-Power Run mode and the Stop

mode to reduce power consumptions. The output of the analysis algorithms (e.g. type of activity, number of steps, etc) are periodically stored into the SD card. Raw data are collected by the accelerometer only.

Data Logger (DL) In this mode, the microcontroller continuously collects raw data from the sensors previously selected, alternating the Low-Power Run mode and the Stop mode to reduce power consumptions. Bunch of raw data are periodically stored into the SD card. No real-time analysis is performed on collected data.

The FSM states and transitions are equivalent for both these operational modes. Starting from the READY state, the device can be set to operate in XX mode (either AT or DL) through a specific Bluetooth command: this causes it to enter the XX_CONN state and to start behaving according to the mode specifications. As soon as the Bluetooth connection is closed, the device enters the XX_WAIT mode: if no other connections are created within a one-minute window, the device enters the low-power log state (XX_LPL) and the Bluetooth switches to standby mode. In this state, the device is running with minimized power consumptions, but it is not able to accept any radio connections until the Bluetooth module is in *Advertising* mode again. To achieve this, a double-click event needs to be identified by the accelerometer, causing the FSM to be brought back to the XX_WAIT state with the Bluetooth being in advertising mode again for one minute: if an active connection is set up within this time, the platform switches back to the XX_CONN state, from which the output of analysis algorithms can be retrieved and the operational mode can be stopped.

5.5 Preliminary Performance Tests

Two different testing firmwares were implemented and embedded (separately, one at a time) in the Winter's microcontroller, in order to preliminary check the functionality and the correct connections of the integrated components.

Firstly, the C initialization codes for the microcontroller and its peripherals were generated by means of the STM32CubeMX, a graphical software configuration tool which includes a pinout-conflict solver, a clock-tree setting helper, and an utility performing MCU peripheral configuration (e.g. GPIO, USART, etc) and middleware stacks (e.g. USB, TCP/IP, etc). Then, the generated programs were edited and enhanced by using the Atollic TrueSTUDIO for STM32 9.0, an IDE dedicated for 32-bit ARM Cortex-M-based microcontrollers. Lastly, each of the

firmware version were uploaded on the flash memory of the device's microcontroller through the 10-pin debug connector.

The developed firmware realize part of the FSM presented previously. Specifically, in both versions the INIT state was fully implemented and comprises:

- Initialization of the microcontroller's peripherals, such as communication interfaces (I^2C, SDMMC and USB), GPIO ports, clock, RTC, and the FAT file system for SD card handling;
- Configuration and test of the sensing modules LSM6DSL and HTS221;
- MicroSD card mounting and write test;
- Bluetooth module power on;
- LEDs power on test.

At the end of the initialization procedure, the microcontroller writes on the SD card (if mounted) a summary file indicating which tests were successful.

The first testing firmware implements the IDLE and the READY states. In this sense, at the end of the INIT state the device automatically moves to the IDLE state (characterized by powered-down sensors and LED slowly blinking), and switches between this and the READY one on the basis of the presence of an active Bluetooth radio connection. The microcontroller runs with a clock frequency of 2 MHz and reacts to a change of the Bluetooth module internal state by piloting the FSM state transition.

Because of the lack of both a Bluetooth commands protocol — and in general, of an external input detection — the device could not yet switch from the READY state to any different one (such as AT_CONN or DL_CONN). For this reason, a second testing firmware was developed to skip the IDLE and READY states, in order to make the device to automatically enter the DL_LPL state right after the INIT one. Following the FSM specifications, in this state the microcontroller is set to continuously collect the data from the inertial module (accelerations and angular rates) and to periodically store them. Specifically, data are collected at a frequency rate of 25 Hz (selected for debug purpose) and saved into a temporary 5 KBytes-size buffer: during this task, the microcontroller works in LPRun mode at 2 MHz to reduce the power consumption. Once the buffer is full, the microcontroller switches to Run mode at 24 MHz: data are quickly transferred to the MicroSD card thanks to the higher clock frequency, the buffer is emptied and the microcontroller restarts collecting data in LPRun mode.

By means of the developed testing firmware, a preliminary evaluation of the overall power consumption was carried out for the states implemented so far (i.e. IDLE, READY, DL_LPL). Moreover, intermediate states such as DL_CONN and DL_WAIT were implemented by editing the firmware to selectively power on/off the inertial sensors and the Bluetooth module, for the purpose of this analysis only.

Table 5.4 – Average power consumption of the implemented Winter states. Supply voltage: 4 V. Inertial sensors ODR: 104 Hz.

State	Settings	Current Consumption
OFF	-	140 µA
IDLE	LPRun @ 2 MHz	900 µA
READY	LPRun @ 2 MHz	1.61 mA
DL_CONN	LPRun @ 2 MHz + Run @ 24 MHz Accelerometer only	1.75 mA
DL_CONN	LPRun @ 2 MHz + Run @ 24 MHz Gyroscope only	2.15 mA
DL_CONN	LPRun @ 2 MHz + Run @ 24 MHz Accelerometer + Gyroscope	2.3 mA
DL_WAIT/ DL_LPL	LPRun @ 2 MHz + Run @ 24 MHz Accelerometer only	1.00 mA
DL_WAIT/ DL_LPL	LPRun @ 2 MHz + Run @ 24 MHz Gyroscope only	1.45 mA
DL_WAIT/ DL_LPL	LPRun @ 2 MHz + Run @ 24 MHz Accelerometer + Gyroscope	1.55 mA

Table 5.4 reports the average current absorption for such states: the power consumption is always below 2.5 mA, resulting in more than 3 days of continuous data collection if the Winter device is equipped with a battery of 210 mAh. Despite a measured mean current consumption of about 30 mA associated with the SD writings, this power drain is negligible thanks to the short time of such operations (about 15 ms) and their rarity (one writing every 30 s). Lastly, power consumptions in DL_WAIT and DL_LPL states resulted to be almost identical despite of the different Bluetooth operative modes: this fact is related to the Bluetooth power management internal system. The module, in fact, automatically enters the sleep mode to save power (about 2 µA according to the datasheet) and switches to advertising mode for a brief time interval (few milliseconds) after a defined period (i.e. 300 ms). Power consumptions related to this operative sequence are similar to those reported for the standby mode (about 1.5 µA), both of them being negligible when evaluating the current absorption of the entire system in DL_WAIT and DL_LPL states.

In addition to the power consumption evaluation, inertial data collected by

Winter (by means of the second testing firmware) were compared to those collected by a MuSe platform. To perform this, both devices were firstly set to operate at the same operative conditions (i.e. frequency rate and sensors full scale) and fixed with tape on the same rigid support; then, two synchronized data acquisitions were carried out while moving the support. Figure 5.15 depicts the data collected by the devices both in static and dynamic conditions: the former include slow rotations of the support around different axes and linear movements along x and y axes (0–80 s time interval), while the latter is achieved by simulating the arm oscillations while walking (80–120 s time interval).

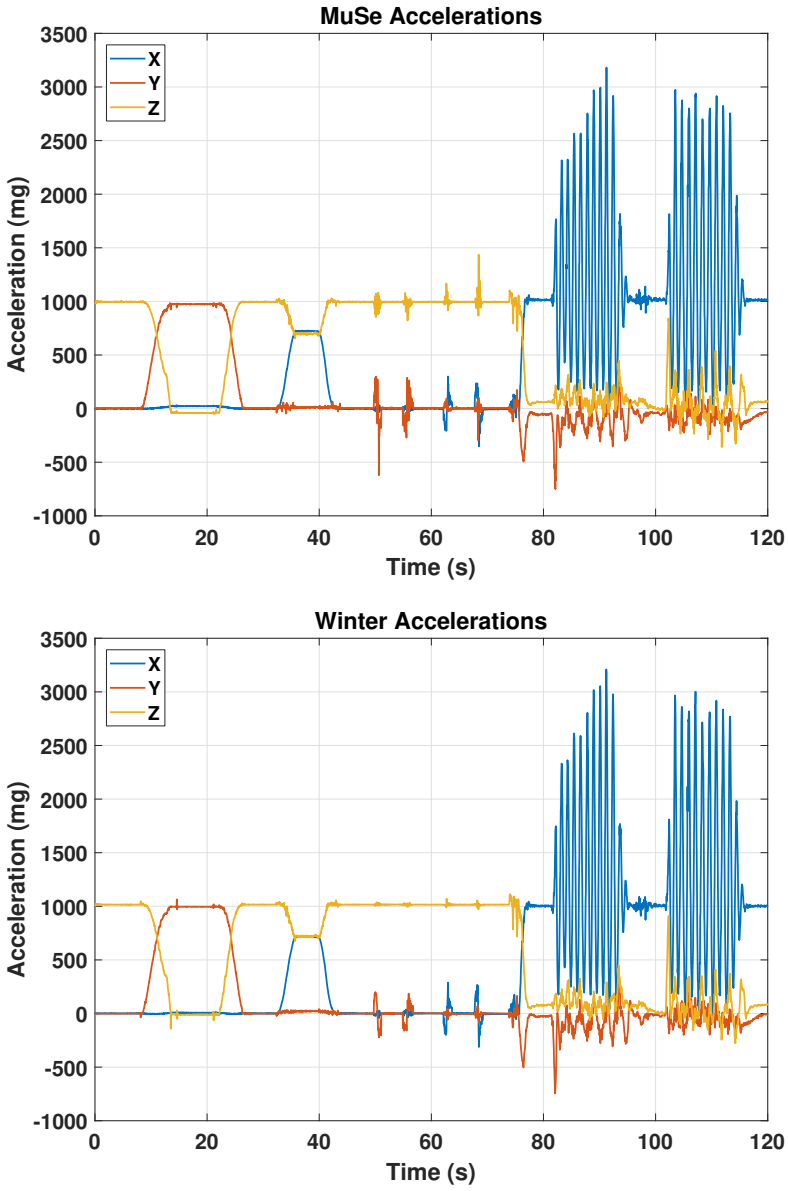


Figure 5.15 – Comparison of accelerations collected by MuSe (top) and Winter (bottom) devices both in static and dynamic conditions.

Acceleration signals collected by the two platforms are almost identical on each

axis, thus confirming the possibility to use the Winter device in place of the MuSe one when the collection of inertial data is needed. Nevertheless, some negligible differences between the platform can still be observed by considering the magnitude of the acceleration vectors, as depicted in Figure 5.16. In particular, the major contribution can be ascribed to the difference between the 0 g-offset values of the MuSe and the Winter devices' inertial modules, i.e. ±90 mg (LSM9DS1 module) and ±40 mg (LSM6DSL module), respectively: such differences could be smoothed once the Winter's accelerometer calibration procedure will be implemented and performed. Moreover, differences in accelerations amplitude can be observed during the execution of the linear movements: this fact might be due to the usage of tape to fix the devices onto the rigid support.

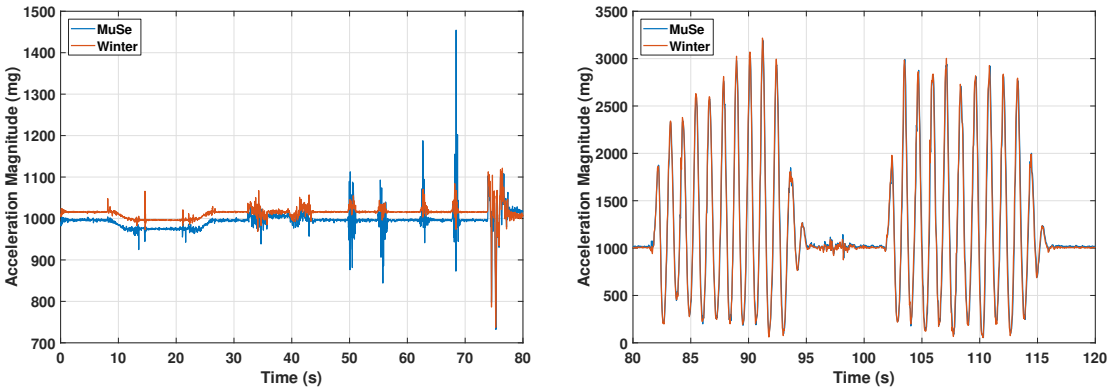


Figure 5.16 – Comparison of acceleration magnitude collected by MuSe and Winter devices in quasi-static (left) and dynamic (right) conditions.

5.6 Future Developments

Unfortunately, the first version of the Winter's firmware was under development at the time of writing of this thesis: the integration of the finite-state machine described previously was at its early stage, with only the INIT, IDLE and READY states successfully implemented, as well as a preliminary version of the DL_CONN and the DL_WAIT states. In this sense, the first goal within the next future will be to complete the development of the firmware, as well as the definition of the Bluetooth v4.1 communication protocol.

Once the firmware will be ready, the overall power consumption of the device will be characterized for each operational mode (both Activity Tracker and Data Logger). Moreover, ADL analysis algorithms described in section 4.5 will be embedded in the firmware and set to run in real-time. The power requirements of such algorithms will be evaluated, and the results of this process will potentially lead to their

further modification and optimization. Then, a comparison with the previously used wearable platforms (Fitbit, eMotion and Muse devices) will be conducted, in order to determine the possibility for the Winter device to be actually used in place of them in their respective projects.

Parallel to the firmware implementation, two different Android apps will be developed based on the Winter's Bluetooth communication protocol. A first, general usage app will allow a research-grade user to:

- Edit the overall configuration of the device (e.g. name, sensors settings, etc);
- Start and stop data acquisition sessions by setting the platform in DL mode;
- Upload new data analysis algorithms versions;
- Set the platform in AT mode.

The second Android app, instead, will be oriented to a consumer usage: it will allow the user to connect to the Winter platform while operating in Activity Tracker mode, and to retrieve the output of the data analysis algorithms such as the level of the physical activity per day, the number of steps walked and other useful information about the daily life activities (e.g. type and duration).

Several expansion boards might be designed to enhance the sensing capability of the Winter device, dependently on the projects it will be used for. Besides their usage within the tremor classification study, a second use case could be the monitoring of the user's heart rate by means of a peripheral photoplethysmography (PPG) sensor, mounted on the finger of the subject. This can be achieved by developing miniaturized, battery-less, sensor-equipped expansion devices connected to Winter through the Micro-USB port: in this way, the master platform will provide the peripheral with the power supply needed to work, and will retrieve data collected by the peripheral sensors according to a custom communication protocol to be defined.

Finally, an updated version of the Winter platform is foreseen in the next future. In particular, one of the most important hardware improvements related to the current device regards the STNS01 module: in fact, although this component minimizes the area occupancy by integrating both a battery charger and an LDO regulator, it also comes with a power consumption above the average of other commercial modules. For this reason, two separated components will replace the STNS01 module, and the LDO regulator will be substituted by a step-down converter with a consequent increased power efficiency.

References

[1] STMicroelectronics, *STM32L475xx*, 2018. [Online]. Available: https://www.st.com/resource/en/datasheet/stm32l475rg.pdf.

[2] Abracon, *ABS06*, 2013. [Online]. Available: https://abracon.com/Resonators/ABS06.pdf.

[3] STMicroelectronics, *LSM6DSL*, 2017. [Online]. Available: https://www.st.com/resource/en/datasheet/lsm6dsl.pdf.

[4] ——, *HTS221*, 2016. [Online]. Available: https://www.st.com/resource/en/datasheet/hts221.pdf.

[5] ——, *SPBTLE-RF*, 2017. [Online]. Available: https://www.st.com/resource/en/datasheet/spbtle-rf.pdf.

[6] ——, *HSP061*, 2015. [Online]. Available: https://www.st.com/resource/en/datasheet/hsp061-2.pdf.

[7] ——, *STNS01*, 2017. [Online]. Available: https://www.st.com/resource/en/datasheet/stns01.pdf.

[8] Maxim Integrated, *MAX17048/MAX17049*, 2016. [Online]. Available: https://datasheets.maximintegrated.com/en/ds/MAX17048-MAX17049.pdf.

Conclusions

This thesis presented simple, wearable-based methods and algorithms oriented towards the monitoring of patients affected by CNS disorders. Applications range from the detailed analysis of particular diseases performed in supervised environments (e.g. hospital and clinics), to a more general evaluation of the physical performances, potentially adaptable to several movement disorders.

Specifically, wireless inertial platforms were exploited:

- to enhance the tremor diagnosis, by building a portable classification system able to differentiate between parkinsonian tremor and essential tremor;
- to instrument some standard clinical motor tests and to allow the development of automated metrics extractors, potentially usable to assess impaired subjects (either affected or not by a CNS disease, e.g. parkinsonian patients and elder adults) in both supervised and unsupervised environments;
- to design and develop the analysis plan and the related algorithms to remotely monitor MS patients during activities of daily living, in order to collect information about their physical disabilities while at home.

On the basis of the available sensors and the usage context, different data analysis approaches were presented. Some algorithms are based on the platform orientation, computed by the sensor fusion algorithm embedded on board. Despite the possibility to extract useful information by applying very simple data analysis procedures, such methods require a preliminary setup of the system to account for possible non-ideal conditions of the environment in which it is intended to be used (e.g. presence of magnetic interferences that might affect the sensor fusion output). Since this operation is not always feasible (non-skilled users, multiple recording locations), methods based only on inertial sensors were designed as an alternative to the orientation-based ones: results of such algorithms were then compared to assess the interchangeability of data sources. Nevertheless, it is important to remember that although the usage of only inertial sensors results in less complex acquisition systems (no more need to embed a sensor fusion algorithm, nor to know the environment

properties), in some cases part of the information might be no longer available or reliable, as in the case of arm swing amplitude in the ETUG test.

Most of the presented studies need further investigations and improvements to be performed in the next future. The main aspect of such work will concern (1) the enlarging of subject samples, to reinforce and support the obtained models and results, (2) the validation with respect to gold standard procedures, and (3) the inclusion of patients in studies where this was not possible. In fact, the unavailability of MS and PD patients in the RADAR-CNS project and motor skill assessment respectively, induced the author to test the developed algorithms on samples of healthy subject only: therefore, some parameters and procedures that worked well for this category of people, might need some further adjustments when dealing with motor-impaired patients, dependently on the disease severity.

The following list summarizes discussions and conclusions presented in this thesis work, divided study by study.

- The development of a wearable-based tremor classification system (see chapter 2) resulted in a set of models able to label tremor-related data (collected from the subject's hand) with accuracies above 80 %. Two feature extraction approaches were developed, and three different data sources were investigated, namely orientation, acceleration and angular speed of the wearable platform. Best results were achieved by classifiers who exploited inertial-based, power-related features, with an average out-of-sample accuracy above 90 %: among them, the simultaneous usage of features from multiple sensors led to better performances. The built models will be embedded either in the developed Android app or in the wearable device itself, on the basis of their nature and complexity. Results encourage the extension of the study to other tremor types, to increase the differentiation capabilities of the developed system.

- Methods to extract information about the execution of motor skills assessment tests were discussed in chapter 3. Among the typical tests used to assess the physical conditions of patients affected by Parkinson's disease, a subset of walking tests (i.e. ETUG test, 6MWT) and balance tasks were selected. Due to the lack of available patients at the time of writing, only healthy young and elder subjects were recruited to be monitored during the execution of such tests by means of scalable body sensor networks. The number of BSN nodes, the availability of on-board sensors and the characteristics of the recording environment are the main parameters to choose in order to select the most appropriate analysis algorithms. Results of the preliminary study confirmed

the equivalence of such methods, however their accuracies will need to be assessed by means of a visual motion tracking system. Moreover, the sample set will have to include patients too, in order to evaluate the robustness of the developed algorithms.

- According to the RADAR-CNS project needs, a first version of algorithms was developed with the purpose of remotely monitoring and evaluating the level of physical activity of MS patients (see chapter 4). Two feature extraction methods were investigated to build classification models able to distinguish among five classes of activities. Despite the relatively low general accuracies obtained from the classifiers (ranging from 70 to 90 %), the main misclassification source (i.e. "Resting" and "Working at PC" differentiation) will not be very critical within the RADAR-MS study, since the main goal is to identify the number of minutes the user spent in sedentary, moderate or vigorous activities only. In addition, a step detection algorithm based on acceleration data provided by a wrist-worn device was developed and tested. Its performance resembles that of the current commercial pedometers, with good reliability at both slow and fast walking paces, and low false positive rate. Nevertheless, such results were obtained from a sample of healthy subjects: in order to be considered as a valid instrument for physical assessment, the algorithm needs to be tested on patients affected by MS.

Finally, driven by the limitations of the wearable devices used in the previous studies, a new inertial platform named Winter was produced. The device was designed to be both an effective activity tracker, potentially embedding the algorithm developed in chapter 4, and a long-term data logger, providing up to a few days of continuous inertial data log thanks to the reduced power consumption and the high-capacitance SD memory. Preliminary performance tests aimed to assess the similarities of inertial data collected simultaneously by Winter and the wearable device used in the rest of this work: results support the possibility of using the new device in place of the older one. Moreover, the first version of the finite-state machine implementation confirmed the capability of the platform to work as data logger in ultra-low power regimes.

For this reasons, the new wearable device together with the developed analysis techniques, will facilitate and enhance the monitoring of patients, in particular those affected by CNS diseases, both in supervised and unsupervised contexts.

Appendix A

MuSe Bluetooth Protocol

MuSe is a Multi-Sensor miniaturized, wireless, low-power, Attitude Heading Reference System (AHRS). With flash storage on board, wireless connectivity (Bluetooth), automated power on/off functions and regulated rechargeable power, the device is a versatile system for data acquisition in a multipurpose fashion. MuSe operates with a Serial Connection over Bluetooth V3.0: the following appendix reports the principal API used for the purposes of this thesis.

A.1 Configuration

To retrieve the device configuration parameters, issue the following ASCII sequence:

?	!	G	E	T	P	A	R	A	M	!	?

The device returns the following 8-Bytes message:

P	P	p_1	p_2	p_3	p_4	p_5	p_6

where:

- PP: message header in the form of ASCII characters;
- p_1 (8-bit unsigned integer): gyroscope full scale [dps];
- p_2 (8-bit unsigned integer): accelerometer full scale [g];
- p_3 (8-bit unsigned integer): magnetometer full scale [G];
- p_4 (8-bit unsigned integer): accelerometer HDR full scale [g];
- p_5 (8-bit unsigned integer): log mode (see below for further information);
- p_6 (8-bit unsigned integer): log frequency (see below for further information).

Table A.1 – Configuration commands

Command	Description	xxx Values
?!GYLFSxxx!?	Set gyroscope full scale	000: ±500 dps 008: ±1000 dps 016: ±2000 dps 024: ±4000 dps
?!ACLFSxxx!?	Set accelerometer full scale	000: ±2 g 008: ±4 g 016: ±6 g 024: ±8 g 032: ±16 g
?!MAGFSxxx!?	Set magnetometer full scale	000: ±500 G 032: ±1000 G 064: ±2000 G 096: ±4000 G
?!ACHFSxxx!?	Set accelerometer HDR full scale	000: ±100 g 016: ±200 g 048: ±400 g
?!LOGMDxxx!?	Set log operating mode	000: None 001: Quaternions 002: HDR 003: Raw 004: Raw and Quaternions 005: High-Resolution
?!LOGFSxxx!?	Set log acquisition frequency	Desired acquisition frequency in Hz (1–250 Hz)

To change any of the parameters described previously, issue the related API listed in Table A.1. To permanently store the device configuration parameters, issue the following ASCII sequence:

?	!	S	T	O	R	E	P	A	R	!	?

Note that, to make the device configuration permanent, after the invocation of set commands, it is necessary to invoke the store command. Otherwise, the specified parameters will be used only for the current session (i.e. the configuration will be lost when the device is turned off).

A.2 Log

If the device has been properly configured by specifying a log mode and frequency, than it is possible to start and stop the log acquisition by pressing the ON/OFF button. In this way, the MuSe device can be used as a completely autonomous system. Alternatively, the log start command can be sent via Bluetooth by the

related API: however, such commands are not reported since this operation mode was not used in this thesis.

A.2.1 Reading a File

To retrieve the list of files in memory, issue the following ASCII sequence:

?	!	G	E	T	F	I	L	E	S	!	?

The device returns the following message:

F	F	n_2	n_1	$ts_{4,1}$	$ts_{3,1}$	$ts_{2,1}$	$ts_{1,1}$	...	$ts_{4,n}$	$ts_{3,n}$	$ts_{2,n}$	$ts_{1,n}$

where:

- FF: message header in the form of ASCII characters;
- n_2, n_1 (2-Bytes unsigned short, big-endian): number of files;
- $ts_4 ... ts_1$ (4-Bytes float, big-endian): timestamp in tenths of milliseconds (10 = 1 ms).

Each timestamp represents the epoch value of the first packet of each file, in the form of a 32-bit unsigned integer.

To retrieve the log corresponding to one particular file, the file list should be firstly acquired. Then, issue the following ASCII sequence:

?	!	R	E	A	D	F	x	x	x	!	?

where "xxx" is the file number, ranging from 1 to N, as obtained from the aforementioned file list. The device returns the following 6-Bytes message:

N	N	p_4	p_3	p_2	p_1

where:

- NN: message header in the form of ASCII characters;
- $p_4 ... p_1$ (4-Bytes unsigned integer, big-endian): Bytes to be read [Bytes].

In this case, the number of Bytes to be read is referred to the selected file for download. Thereafter, the device automatically starts to send the stored packets (i.e., $N/512$) according to the following pattern.

PACKET OF 512 BYTES									
check_B4	check_B3	check_B2	check_B1	...	...	...	...	...
Byte #513	Byte #514	Byte #515	Byte #516					

Each transmission packet is followed by 4 Bytes indicating the 32-bit unsigned integer checksum (i.e. check_B4 ... check_B1 Bytes in the table above), which is computed as the sum of all the 8-bit unsigned integer elements of the packet, header included. Furthermore, for the sake of data integrity, the device requires that the master (e.g. PC, smartphone, etc.) confirms continuously each 20 packets that data have been correctly received. To do so, each 20 packets correctly received, issue the following ASCII sequence:

?	!	P	A	C	K	E	T	O	K	!	?

The device resumes the transmission process as soon as the confirmation command is received. In the case of any data corruption, issue the following ASCII sequence within 1.5 seconds from the last packet transmission:

?	!	P	A	C	K	E	T	K	O	!	?

The device transmits again the last 20 packets as soon as the negative acknowledge is received. If no command is sent back to the device within 1.5 seconds from the last 512 Bytes packet transmission, the overall download process will be ended.

A.2.2 Packet Format

On the basis of the log mode selected, packages within a file are arranged as follows.

Raw

Data are arranged in packets of 512 Bytes each, stored as follows:

P	K	m	f	ts_4	ts_3	ts_2	ts_1	Acc	Gyr	Mag
								x28		
512 Bytes										

where:

- PK: message header in the form of ASCII characters;
- m: log mode;
- f: log frequency;

- $ts_4...ts_1$ (4-Bytes float, big-endian): timestamp in tenths of milliseconds (10 = 1 ms);
- Acc (2-Bytes signed short each, big-endian): accelerometer components;
- Gyr (2-Bytes signed short each, big-endian): gyroscope components;
- Mag (2-Bytes signed short each, big-endian): magnetometer components.

As above, the accelerometer, gyroscope and magnetometer components are represented using 6 Bytes total each, 2 Bytes for each axis.

Raw and Quaternions

Data are arranged in packets of 512 Bytes each, stored as follows:

<table>
<tr><td>P</td><td>K</td><td>m</td><td>f</td><td>ts_4</td><td>ts_3</td><td>ts_2</td><td>ts_1</td><td>Acc</td><td>Gyr</td><td>Mag</td><td>q_i</td><td>q_j</td><td>q_k</td></tr>
<tr><td colspan="8" rowspan="2"></td><td colspan="6">x21</td></tr>
<tr><td colspan="14">512 Bytes</td></tr>
</table>

where:

- PK: message header in the form of ASCII characters;
- m: log mode;
- f: log frequency;
- $ts_4...ts_1$ (4-Bytes float, big-endian): timestamp in tenths of milliseconds (10 = 1 ms);
- Acc (2-Bytes signed short each, big-endian): accelerometer components;
- Gyr (2-Bytes signed short each, big-endian): gyroscope components;
- Mag (2-Bytes signed short each, big-endian): magnetometer components;
- q_i, q_j, q_k (2-Bytes signed short each, big-endian): quaternion components.

As above, the accelerometer, gyroscope and magnetometer components are represented using 6 Bytes total each, 2 Bytes for each axis. Furthermore, also in this case, only the imaginary parts of each quaternion is saved. Hence, the corresponding real component can be computed as follows:

$$q_w = \sqrt{1 - \frac{q_i^2}{2^{15} - 1} - \frac{q_j^2}{2^{15} - 1} - \frac{q_k^2}{2^{15} - 1}}.$$

Appendix B

Machine Learning Overview — Classification

In the field of computer science, *machine learning* is a subset of artificial intelligence that often uses statistical techniques to give computers the ability to "learn" (i.e., progressively improve performance on a specific task) with data, without being explicitly programmed. It explores the study and construction of algorithms that can make data-driven predictions or decisions, through building a model from sample inputs. By means of these techniques, computers do what comes naturally to humans and animals — learn from experience: the algorithms adaptively improve their performance as the number of samples available for learning increases.

Machine learning finds natural patterns in data by using two types of techniques:

Supervised learning The machine learning task of learning a function that maps an input to an output based on example input-output pairs. In supervised learning, each example is a pair consisting of an input object (typically a vector) and a desired output value (a label). A supervised learning algorithm analyzes the training data and produces an inferred function, which can be used for mapping new examples.

Unsupervised learning The machine learning task of inferring a function that describes the structure of "unlabeled" data, i.e. data that has not been classified or categorized. The aim of unsupervised learning algorithms is to find hidden patterns or intrinsic structures (such as groupings) in input data. Since the examples given to the learning algorithm are unlabeled, there is no straightforward way to evaluate the accuracy of the structure that is produced by the algorithm.

Supervised learning problems can be further grouped into *regression* and *classification* problems, on the basis of the response type.

Classification The problem of identifying to which of a set of *discrete* categories (or "classes") a new observation belongs, on the basis of a training data set containing observations whose category membership is known. For instance, whether an email is genuine or spam, or whether a tumor is cancerous or benign are typical classification problems. The corresponding unsupervised procedure is known as *clustering*.

Regression The problem of predicting a *continuous* quantity output for a new observation. This can be done by approximating a mapping function from input variables to a continuous output variable. Predict changes in temperature or fluctuations in power demand are typical regression problems.

Since the use of machine learning in this thesis is restricted to the PT-ET differentiation problem, the following sections are mainly focused on the standard steps of a classification task, although many parts are valid for regression and clustering problems too.

B.1 Preparing Data

A typical machine learning workflow starts with the preparation of data. For a supervised learning task, a given dataset can be split into two subsets, X and Y. The former denotes the "feature set", defined by the following notation:

$$X = \left\{\vec{x}^{(n)} \in R^d\right\}_{n=1}^{N}. \tag{B.1}$$

This subset contains N samples, or *observations*. Each observation is a d-dimensional vector $\vec{x}^{(n)} = \left[x_1^{(n)}, x_2^{(n)}, ..., x_d^{(n)}\right]$ called a feature vector, while each of its dimensions is called feature, variable, or *predictor*. Features can be of a quantitative (numerical) or a qualitative (categorical) nature. On the other hand, the "label set" Y records what label a feature vector corresponds to, and is defined as follows:

$$Y = \left\{y^{(n)} \in R^d\right\}_{n=1}^{N}. \tag{B.2}$$

Values in this subset are also called *responses*. When dealing with unsupervised learning tasks, the label set is absent or unknown.

One of the first steps in the data preparation involves a data quality check. Sometimes datasets include data that may alter the learning process, such as outliers–data points lying outside the rest of the data. It is important to decide whether these outliers are due to some acquisition system fault and can thus be

ignored, or whether they indicate a phenomenon that the model should account for. In other cases, some values in the dataset may be missing. One solution might be to simply ignore the missing values, but this would reduce the size of the data set; alternatively, missing values might be substituted with approximations obtained by interpolating or using comparable data from another sample.

In supervised learning tasks, three data sets are commonly used in different stages of the creation of the model (Figure B.1). These datasets may either be given or obtained by splitting the original if only one dataset is provided.

- The model is initially fit on a *training dataset*, that is, a set of observations used to fit the model's parameters.
- A second dataset called *validation set* is then used to provide an unbiased evaluation of a model fit on the training dataset while tuning model hyperparameters (for the definition of hyperparameter, see appendix B.4). The evaluation becomes more biased as skill on the validation dataset is incorporated into the model configuration. In some occasions, the validation set might be a portion of the training set held out from the actual training phase.
- Lastly, a third dataset called *test set* is used to provide an unbiased evaluation of the chosen model fit on both the training dataset. This data set is never used to choose among two or more models. The final model could be fit on the aggregate of the training and validation datasets.

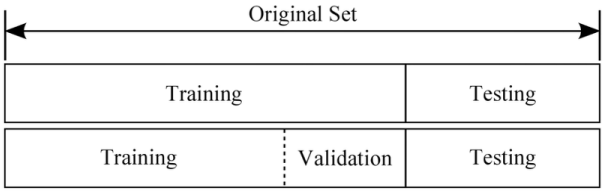


Figure B.1 – Training, validation and test data subsets.

B.2 Choosing the Classification Learning Algorithm

Selecting a machine learning algorithm is a process of trial and error. It is also a trade-off between specific characteristics of the algorithms, such as speed of training, memory usage, predictive accuracy on new data and transparency or interpretability (how easily the user can understand the reasons an algorithm makes its predictions). The following list contains the most commonly used classification algorithms, the mechanisms of which are also depicted in Figure B.2.

k-Nearest Neighbor (kNN) This algorithm categorizes objects based on the classes of their nearest neighbors in the dataset, assuming that objects near each other are similar: the predicted class label will be set as the most common label among the k closest neighboring points. Distance metrics such as Euclidean, city block, cosine, and Chebychev are used to find the nearest neighbor. kNN works well on basic recognition problems. Nevertheless, the algorithm does not learn anything from the training data, it simply uses the training data itself for classification: this can result in an algorithm not generalizing well and also not being robust to noisy data. Moreover, the algorithm must compute the distance and sort all the training data at each prediction, which can be slow if there are a large number of training examples.

Support Vector Machine (SVM) This algorithm classifies data by finding the linear decision boundary (hyperplane) that separates all data points of one class from those of the other class. The best hyperplane for an SVM is the one with the largest margin between the two classes, when the data is linearly separable. If this is not the case, SVM may use a kernel transform to transform nonlinearly separable data into higher dimensions where a linear decision boundary can be found. SVM is best used for two-classes classification problems (even if it may be extended to multiclass problems); it is also a simple and easy-to-interpret classifier. However, it has several key parameters that need to be set correctly to achieve the best classification results: the user may have to experiment with a number of different parameter settings in order to achieve a satisfactory result.

Naïve Bayes A naïve Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature. It classifies new data based on the highest probability of its belonging to a particular class. It is best used for small datasets with many variables, but the best results are only achieved when the class-condition independence assumption holds, which is not always true.

Discriminant Analysis Discriminant analysis classifies data by finding linear combinations of features. Discriminant analysis assumes that different classes generate data based on Gaussian distributions. Training a discriminant analysis model involves finding the parameters for a Gaussian distribution for each class (namely, the mean and the covariance matrix). The distribution parameters are used to calculate boundaries, which can be linear or quadratic functions: these boundaries are used to determine the class of new data. The

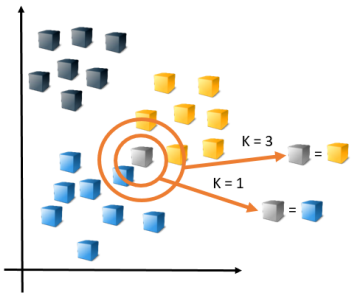
algorithm is characterized by low memory usage during the training and fast prediction time, but relies on the hypothesis that the class conditional densities of clusters are approximately normal.

Decision Tree A tree consists of branching conditions where the value of a predictor is compared to a trained weight. The number of branches and the values of weights are determined in the training process. Additional modification, or pruning, may be used to simplify the model. The main advantage of a decision tree is that the model is particularly fast at classifying new input samples. On the other hand, the main limitation is that very large models will frequently overfit the training data.

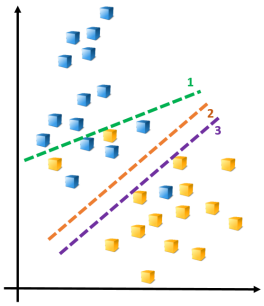
B.3 Model Validation and Evaluation

Once a model and its hyperparameters have been selected, the training dataset is used to fit parameters of the model. In order to check its performance, the trained model needs to be validated on a labeled set of data to obtain the so-called *misclassification error*, that is, the number of misclassified observation. The result of the validation may be then used to decide whether to train a different model or whether to tune the current model's hyperparameter, in order to achieve better performances. In this sense, many training-validation runs produce a biased estimate of the skills of the final tuned model, since the final choices depend on the validation results. For this reason, once the final model is chosen, it is fed with data from the test dataset: because these data were never been used, the result of this evaluation will be an unbiased and more realistic measure of the model performance.

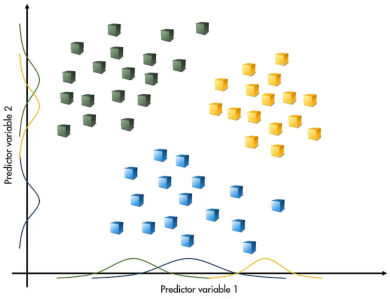
After the selection and removal of the test samples from the original dataset, the subsequent partitioning into training and validation sets can be performed by using several strategies, the choice of which depends on different factors such as the number of available samples and the required computational costs. One simple strategy is the *hold-out validation*. In the hold-out method, the observations are randomly assigned to the training and validation sets, the latter being typically smaller than the former. Once the highest validation performance is reached, the training and validation sets are used to train the final model: the testing set is then applied to the model to evaluate the generalization performance. Although this method is very simple to implement and has a low computational impact, it also comes with some disadvantages:



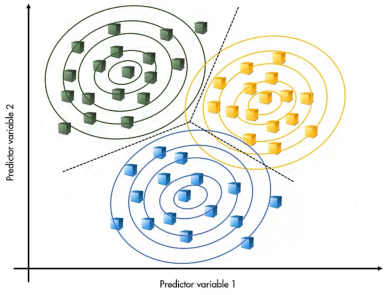
(a) k-Nearest Neighbor. The class of the unknown point depends on the value of k.



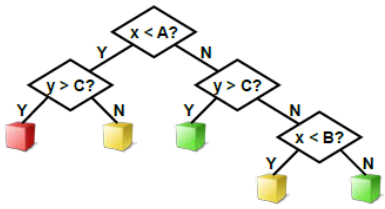
(b) Support Vector Machine. The decision boundary in the middle is the one with the largest margin.



(c) Naïve Bayes. For each feature, the class probability distribution is determined.



(d) Discriminant Analysis. The multivariate Gaussian distributions and their boundaries are computed.



(e) Decision Tree. Predictors x and y are compared to the trained weights A, B and C.

Figure B.2 – Supervised learning, classification algorithms.
Source: MathWorks, *Machine Learning with MATLAB*

- The evaluation may depend heavily on which data points end up in the training set and which end up in the validation set, and thus the evaluation may be significantly different depending on how the division is made.

- Data from the validation set are never used to train a model (except for the final version). If the size of the dataset is small, the model may need every possible data sample to adequately determine its parameters.

- Each model is validated with a single data set, which has limited ability to

characterize the uncertainty in the results.

A popular strategy to improve over the hold-out method is the *k-fold cross-validation*. Within this technique the reference to a general "validation dataset" disappears, and the original data set is split into training and test sets only. K-fold validation evaluates the data across the entire training set, but it does so by dividing it into k "folds" — or subsections — and then the hold-out method is repeated k times: each time, one of the k subsets is used as the current validation set, whereas the other $k-1$ subsets are put together to form the current training set. Lastly, once the best parameter combination has been found, the model is retrained on the starting training data, and the average error across all k trials is computed. The advantage of this method is that it is independent from how the data gets divided: every data point gets to be in a validation set exactly once, and gets to be in a training set $k-1$ times. The disadvantage of this method is that the training algorithm has to be rerun from scratch k times, which means it takes k times as much computation to make an evaluation.

To allow the visualization of the performance of a classification algorithm, prediction results are commonly reported by means of the *confusion matrix*. Within this table layout, each row represents the instances in an actual class while each column represents the instances in a predicted class (or vice versa). The name refers to the fact that it makes it easy to see if the system is confusing two classes.

		Predicted Class		
	N = 100	**Positive**	**Negative**	
Actual Class	**Positive**	TP = 45	FN = 9	54
	Negative	FP = 6	TN = 40	46
		51	49	100

Figure B.3 – A typical confusion matrix. TP: True Positives; FP: False Positives; FN: False Negatives; TN: True Negatives.

Figure B.3 depicts a two-class confusion matrix. Given the classes "positive" and "negative", TP (true positives) and TN (true negatives) represent the number of observations that were correctly classified. On the other hand, FP (false positives) and FN (false negatives) represent the misclassified observations. More detailed analysis on the classification model can be performed by combining the information in the confusion matrix: the following list reports the most popular measures.

Sensitivity, Recall or TPR (True Positive Rate) Number of observations correctly identified as positive out of total actual positives:

$$TPR = \frac{TP}{TP + FN}$$

Specificity or TNR (True Negative Rate) Number of observations correctly identified as negative out of total actual negatives:

$$TNR = \frac{TN}{TN + FP}$$

Precision Number of observations correctly identified as positive out of total observations identified as positive:

$$Precision = \frac{TP}{TP + FP}$$

Fall-out or False Positive Rate (FPR) Number of observations wrongly identified as positive out of total true negatives:

$$FPR = \frac{FP}{TN + FP} = 1 - TNR$$

Miss rate or False Negative Rate (FNR) Number of observations wrongly identified as negative out of total true positives:

$$FNR = \frac{FN}{TP + FN} = 1 - TPR$$

Accuracy Percentage of total observations classified correctly:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}$$

F1 Score The harmonic average of precision and recall, reaching its best value at 1 and its worst at 0:

$$F_1 = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}} = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

The particular metric to use for the model evaluation depends on the specific problem. Accuracy is the most intuitive performance measure and it works best when the detection of both classes is equally important, that is, if false positives and false negatives have similar costs. On the other hand, when the ability to detect correctly positive samples is the main focus (correct detection of negatives examples is less important to the problem) and/or the positive class is much smaller

than the negative one, precision and recall (or their combination) are preferred. For instance, a classification model that correctly detects 99 negatives and erroneously classifies the single positive sample in the population, has an accuracy of 99 %: this model should not be considered good, even more so if the cost related to the misclassification of the positive class is high (e.g., classify a sick subject as healthy, or a fraudulent bank transaction as legal). In this case, precision and recall are more descriptive than the accuracy, being undefined ($0/0$) and 0 % ($0/1$) respectively.

B.4 Improving the Model

B.4.1 Dimensionality reduction

Learning algorithms are often computation intensive and reducing the number of predictors can have significant benefits in calculation time and memory consumption. Also, reducing the number of predictors results in simpler models which can be generalized and are easier to interpret. Two common ways to reduce the number of predictors in a model are *feature transformation* and *feature selection* techniques.

A commonly used method for feature transformation and dimensionality reduction is Principal Component Analysis (PCA). This is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called *principal components*. This transformation is defined in such a way that the first principal component has the largest possible variance (that is, accounts for as much of the variability in the data as possible), and each succeeding component in turn has the highest possible variance under the constraint that it is orthogonal to the preceding components (see Figure B.4). The resulting vectors are an uncorrelated orthogonal basis set. If the components are ordered by the variation explained in the data, PCA can be used for dimensionality reduction by discarding the components beyond a chosen threshold of explained variance.

The data often contains predictors which do not have any relationship with the response. These predictors should not be included in a model. For example, the patient-id in the heart health data does not have any relationship with the risk of heart disease. The data can also contain highly correlated predictors so that only one of them needs to be included in the model. Feature selection techniques choose a subset of predictors to be included in the model. A feature selection algorithm can be seen as the combination of a search technique for proposing new feature subsets, along with an evaluation measure which scores the different feature subsets. The simplest algorithm is to test each possible subset of features finding the one

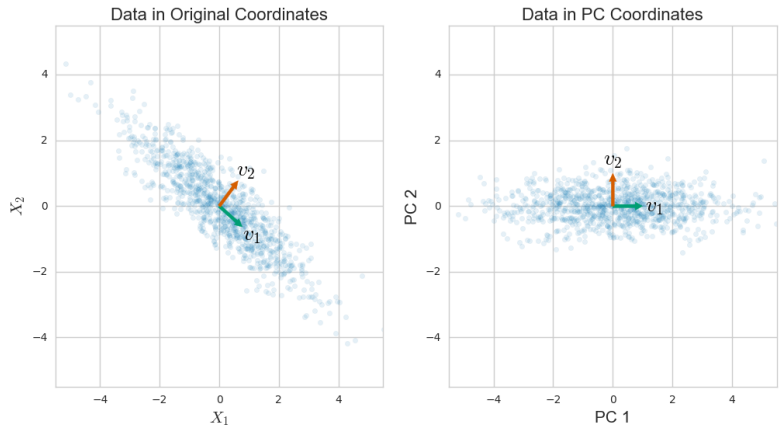


Figure B.4 – Principal Component Analysis performed on a set of 1000 samples drawn from a multivariate Gaussian.
Source: *Intoli*, How are Principal Component Analysis and Singular Value Decomposition related?

which minimizes the error rate. This is an exhaustive search of the space, and is computationally tractable only for small feature sets. The choice of evaluation metric heavily influences the algorithm, and it is these evaluation metrics which distinguish between two main categories of feature selection algorithms: *wrappers* and *filters*.

- Wrapper methods use a predictive model to score feature subsets. Each new subset is used to train a model, which is tested on a hold-out set. Counting the number of mistakes made on that hold-out set (the error rate of the model) gives the score for that subset. As wrapper methods train a new model for each subset, they are very computationally intensive, but usually provide the best feature set for that particular type of model.

- Filter methods use a proxy measure instead of the error rate to score a feature subset, such as the correlation with the variable to predict. Filters produce a feature set which is not tuned to a specific type of predictive model. These methods are particularly effective in computation time and robust to overfitting. However, filter methods tend to select redundant variables because they do not consider the relationships between variables. Many filters provide a feature ranking rather than an explicit best feature subset.

B.4.2 Hyperparameter Optimization

Other than selecting an appropriate feature subset, a common way to improve a model performance is to properly tune its *hyperparameters*. In machine learning,

a hyperparameter is a parameter whose value is set before the learning process begins; by contrast, the values of other parameters are derived via training. For a polynomial regression curve, for instance, the curve coefficients represent the parameters of the model, while the order of the polynomial is a hyperparameter. The problem of choosing a set of optimal hyperparameters for a learning algorithm is called *hyperparameter optimization*: this process finds a tuple of hyperparameters that yields an optimal model which minimizes a predefined loss function on given independent data. Among the different optimization procedures, the following ones are the most commonly used.

Grid search An exhaustive searching through a manually specified subset of the hyperparameter space of a learning algorithm.

Random search Replaces the exhaustive enumeration of all combinations by selecting them randomly. It can outperform grid search, especially when only a small number of hyperparameters affects the final performance of the model. Moreover, it can be easily separated into parallel tasks.

Bayesian optimization Builds a probabilistic model of the function mapping from hyperparameter values to the objective evaluated on a validation set: by iteratively evaluating a promising hyperparameter configuration based on the current model, and then updating it, Bayesian optimization aims to reveal as much information as possible about this function and, in particular, the location of the optimum. Bayesian optimization provides better results in fewer evaluations compared to grid and random searches, due to the ability to reason about the quality of experiments before they are run.