Università degli Studi di Padova

Università degli Studi di Padova

Department of Information Engineering

Master Thesis in Automation Engineering

Design of a sensorized knee brace for safety systems

Supervisor:
Prof. Giada Giorgi

Master Candidate:
Davide Galvani

Academic Year 2021/2022

Abstract

Anterior cruciate ligament (ACL) injury is the most frequent injury occurring in skiers. This study is part of a wider project that aims to design a system to prevent this type of injury on ski slopes. The work exposed in this thesis focuses on the design of a method capable of monitoring the dynamics of a knee in real time. To achieve this, it is necessary to understand the mechanisms that cause injuries, through a medical-mechanical analysis of the forces and torques exerted on the ligaments of a knee, investigating above all the conditions and risk factors that are most likely to predispose an athlete to incur a knee injury. This step is essential to understand which parameters need to be measured to allow us to choose the most suitable sensors.

To accomplish our project we chose to use inertial measurement systems, which enable us to easily assess the angles formed by the knee and the relative rotations between the thigh and shin. In order to achieve our final goal, we decided to proceed by splitting our purpose into sub-problems, designing a first analogue measuring system that allows us to estimate the degrees of movement of the knee, in quasi-static conditions. This phase helped us to evaluate the limitations of the measurement system adopted, leading us to develop a new, more advanced digital system capable of monitoring the dynamics of a knee in real time.

The results presented in this thesis have shown great promise. It should be noted that in addition to the purpose already enunciated, the work shown in this paper can also be used as a method to record data in real time, in order to provide medical information in the event of an accident. Regardless of how this thesis may be used, the ultimate goal of the research exposed in this document remains to lay a solid foundation for the future development of an advanced and portable system for monitoring a critical part of a skier's body. Each step presented in this thesis is supported by detailed explanations so that it is possible to reconstruct the philosophy behind what has been presented and, if necessary, improve upon what has been done.

Sommario

La lesione del legamento crociato anteriore (LCA) è la causa più comune di infortunio negli sciatori. Questo studio fa parte di un progetto più ampio il cui scopo ambisce a concepire un sistema per prevenire questo tipo di trauma sulle piste da sci. Il lavoro presentato in questa tesi si è concentrato sulla progettazione di un metodo in grado di monitorare la dinamica di un ginocchio in tempo reale. Per raggiungere questo obiettivo, è stato necessario comprendere i meccanismi che causano gli infortuni, attraverso un'analisi medico-meccanica delle forze e delle coppie esercitate sui legamenti del ginocchio, indagando soprattutto le condizioni ed i fattori di rischio che più probabilmente predispongono un atleta a subire un infortunio al ginocchio. Questo passo è stato essenziale per capire quali parametri devono essere misurati al fine di scegliere i sensori più adatti.

Per il nostro progetto abbiamo scelto di utilizzare dei sistemi di misurazione inerziale, in quanto ci permettono di valutare facilmente gli angoli formati dal ginocchio e le rotazioni relative tra la coscia e la gamba. Per raggiungere il nostro obiettivo finale, abbiamo deciso di procedere scomponendo il nostro scopo in sottoproblemi, progettando un primo sistema di misurazione analogico che permette di stimare i gradi di movimento del ginocchio, in condizioni quasi-statiche. Questa fase ci ha aiutato a valutare i limiti del sistema di misurazione adottato, portandoci a sviluppare un nuovo sistema digitale più avanzato in grado di monitorare la dinamica di un ginocchio in tempo reale.

I risultati presentati in questa tesi si sono rivelati molto promettenti. Si noti che oltre allo scopo già enunciato, il lavoro mostrato in questo elaborato può essere utilizzato anche come un metodo per registrare dati in tempo reale, al fine di fornire informazioni mediche in caso di incidente. Indipendentemente dal modo in cui questa tesi può essere utilizzata, l'obiettivo finale della ricerca esposta in questo documento resta quella di porre delle solide fondamenta per sviluppare in futuro un sistema avanzato e portatile per il monitoraggio di una parte critica del corpo di uno sciatore. Ogni passaggio presentato in questa tesi è supportato da spiegazioni dettagliate in modo che sia possibile ricostruire la filosofia dietro a ciò che è stato presentato e, se necessario, migliorare ciò che è stato fatto.

Contents

List of Figures

List of Tables

List of Codes

Chapter 1

Introduction

1.1 Purpose of the thesis

The Alpine ski is an Olympic discipline since 1936 and every year is practiced by millions of amateurs and professionals. The equipment necessary to perform this sport has evolved over the years to make it more comfortable to practice and, at the same time, to reduce the number of injuries. A high level skier can found on the market some very advanced articles that can protect him, including: *mechatronics bindings* able to understand when is necessary the unhooking of the skis during falls, *smart helmets* with airbag to avoid blows to the head and *sensorized boots* able to detect incorrect movements of the legs. However, these improvements in equipment have generated a curious fact: the parts that have undergone fewer updates today are the ones that get injured more easily. For example, several surveys of beginners and professional skiers have shown that the knee is the body part that suffers the most injuries. This is due to two distinct facts: the knee is a very exposed part while skiing and new equipment is difficult to develop. Anterior cruciate ligament (ACL) injury is the most common lesion that occurs on the ski slopes, followed by other ligament injuries and tibia and fibula tears. The ideal situation would be to have a tool that can prevent all knee injuries. However, we might actually consider it optimal to get a garment that can significantly reduce the number of accidents.

This thesis is part of a large project, the aim of which is to develop a mechanism capable of protecting the knee, reducing the number of injuries to which skiers are subjected, avoiding worsening their mobility and performance. In particular, the purpose of this study is to design a measurement system capable of analysing in real time the dynamics of a skier's knee while skiing. The first

step in designing the final system is to understand how knee injuries occur, what causes them, and the mechanics of accidents. We also need to understand how a knee is made, assess where the forces and torques that cause injuries are exerted and what actions can be taken to reduce these forces. After that, it is essential to identify the signals we need to measure, and then we can prepare an initial prototype to see if these signals are sufficient to identify incidents. Finally, we need to validate what has been done and possibly modify what is not working. So the idea behind the whole project is to design a knee brace whose ultimate goal is to make skiers' knees immune to injury. Obviously this is only the ideal that we would like to achieve. In this thesis we design a first system capable of measuring the dynamics of a knee. Initially, we conducted a practical feasibility study to assess whether it is possible to design a knee brace with a device that can act when conditions occur that cause an accident. After discovering that this is not always possible, we propose some ideas that can theoretically strengthen the skier's knee to avoid injury.

To accomplish our task, we decided to use a commercial knee brace as a frame and apply some inertial sensors, such as accelerometers and gyroscopes, in different configurations on it. We then validated each configuration and finally chose the one that best replicates the dynamics of the knee and is most reliable, accurate and economical. Any other parts related to the development of active safety systems, such as airbags or mechanical-pneumatic components and their activation, are left for future work.

Regardless of the purpose of designing an active safety system for skiers, this thesis can also be understood in a more general way: the benefits of developing a portable system capable of tracking knee dynamics can also be exploited in many other fields, particularly for rehabilitation. Although similar systems already exist in physical therapy clinics, they are mostly intended to be used only in the presence of an expert who controls this system. In contrast, the tracking solution developed in this thesis could allow for continuous monitoring of patients, without any external intervention and, above all, does not require a calibration procedure, which is typical of existing instruments..

1.2 Thesis structure

The thesis is organized as follows:

- In Chapter 2, we expose a practical feasibility study, analyzing the injury

statistics of skiers and trying to understand the mechanisms that cause ACL injuries. We also describe how a knee is made from an anatomical point of view and present a short mechanical study of a skier, to understand if effectively the initial idea can be pursued.

- The Chapter 3 presents the specifications that the system we have to design must satisfy. Next, we describe how we chose the sensors we would like to use. Then we present a literature research on the state of the art in knee dynamics monitoring. Finally, we describe the mathematical and statistical aspects we have used for the implementation of the algorithms, including the Kalman filter, the complementary filter, rotation matrices and quaternions.

- Chapter 4 shows how we designed the first system capable of estimating the range of motion of a knee, based on analogicWe then present the instrument setup used, how we calibrated the sensors used, and the algorithms we implemented to achieve our aim. We analyse the results of practical tests obtained in the laboratory and compare these data with experiments conducted using various layouts for sensor positioning. Finally, we evaluate the advantages and disadvantages of the techniques used.

- In Chapter 5 we propose a new prototype developed from the results obtained from the analogue system presented in the previous chapter. To accomplish this task, we used an Arduino Mega and digital IMUs, exploiting statistical filtering methods and quaternions to merge data from the various sensors. We then analyse the results of these experiments, comparing them with each other and with those obtained from the analog system, and evaluating which aspects could be improved.

- In Chapter 6 we summarise what has been exposed in this thesis, focusing on the problems encountered and the results obtained. Finally, we analyse the progress that could be made in future work, also proposing solutions that we were not able to implement within this study.

Chapter 2

Feasibility study

The first step in developing an instrumented knee brace for the active safety of skiers is to analyze the accidents that occur on the ski slopes. This step allows us to evaluate the usefulness of the study that follows in this thesis: skier injuries are very frequent both at the amateur and professional level and their reduction would make possible to practice this sport with less possibility of injury.

The epidemiological study described in this chapter is divided into two parts: first, a *descriptive epidemiological* research is exposed, whose purpose is to analyze the characteristics of skiers' injuries. Next, an *analytical epidemiological* research is presented, which investigates the causes that generate the accidents described in the previous part. In addition, an anatomical description of how a knee is made is exposed, highlighting the parts that can be most damaged while skiing. Finally, the actual technical feasibility study related to the initial project idea that led to this study is presented.

2.1 Statistics on skiers' injuries

A brief literature review was conducted to understand the types of injuries that can occur on a ski slope. Because this work is very time consuming and, more importantly, beyond the scope of this paper, we exploited some of the most recent literature reviews. We performed our search on databases typically used in the medical field, such as *PubMed, Pedro, Ovid* and *Google Scholar*. The search terms we used were *alpine ski injuries*. Then we skimmed all the results obtained through the following criteria: the articles we were looking for must be published between 2010 and 2020, in English, and about skiers only, excluding injuries from other winter sports. We adopted the latter criterion because other categories,

such as snowboarders, might be more prone to different types of injuries than skiers. For the first criterion, however, we assumed that the equipment used for skiing has been greatly improved over the years; therefore, not all accidents that happened too long ago are still happening today. Alternatively, they may still occur, but with a much lower frequency.

An example of how advances in equipment lead to significant benefits is helmets. In [33] *Shealy et al.* conducted an epidemiological study of all injuries (for recreational skiing) at a specific location during 17 ski seasons. They found that the percentage of head injuries decreased from 8.4% to 6.8% , while helmet use increased from 8% to 84%. They also found that the prevalence of potentially serious head injuries decreased from 4.2% to 3.0%. This is due to two concomitant factors: first, helmet use has increased exponentially among the population, leading to greater protection of a highly exposed body part. Second, head protection has advanced, inducing cheaper helmets made of lightweight, yet very durable materials.

From 290 results on *PubMed* alone, we reduced the selection to a smaller number of articles of our interest. Initially, we searched for articles covering injuries incurred by professional skiers during ski racing. This is important to emphasize, because athletes are well trained and typically use the most advanced equipment. So the injuries they suffer are either very rare or are so common that even the latest equipment cannot protect them. In particular, we assumed that knee injuries are fairly common for any level of skier, so we were looking for confirmation of this hypothesis. Developing injury prevention strategies common to professionals also benefits beginning skiers.

Tarka et al. [36] led a literature review of all injuries to skiers at the World Cup level from 1976 to 2008. They found that the most commonly injured body parts in adult and youth skiers are, in order of cadence: the knee, spine, hand, and lower leg/foot/ankle. They found that the alpine skiing discipline called *downhill* has the highest incidence of injury, followed by *super giant slalom*, *giant slalom* and *slalom*. Age is not relevant in the frequency or type of accidents. Regarding the sex of the injured, studies done so far are conflicting: some report that females are injured more often, while others find no difference between the sexes. Most of the accidents, which led to injuries, occurred during competitions, due to falls or unbalanced situations; the remainder occurred during training. In particular, the conditions that generate injuries are mainly caused by errors of judgment, fatigue, inattention, excessively challenging routes and poor visibility.

Focusing on specific injuries, they note that the most common are the following:

- **Anterior Cruciate Ligament (ACL)**: some studies have observed that females are more likely to suffer this injury, while others claim that there is no difference between males and females. Regarding the risk factors, two studies state that there is a greater likelihood of injury on the left leg, although it is not specified which is the predominant leg in skiers. For the mechanism that leads to this injury, we leave the discussion to Section 2.3.

- **Back and Hip Pain**: a low back pain afflicts over a third of World Cup skiers. A study found out that elite ski racers show an anterior endplate lesions more often compared to non-skiers. It is probably due to repetitive forces coupled with forward and lateral bendings, which cause accelerated wear of the vertebral discs.

- **Articular Cartilage**: 20 alpine ski racers reported a knee microfracture, with "full thickness chondral lesions". After surgery and more than 13 months, 19 of 20 patients returned to competitive skiing, improving their World Cup ranking.

Jordan et al. [19] performed a literature review about ACL injury/reinjury during alpine ski racing. They analyzed 62 papers published between 1991 and 2017, regarding only professional skiers. They too have found that most accidents occur during the discipline of *downhill* compared to other events. In particular, they analyzed data from Injury Surveillance System (ISS), founded by International Ski Federation in 2006, discovering that the ACL injury is the most frequent diagnosis, with an incidence of 14% among all injuries. They also claim that there is no statistical evidence of knee injuries suffered between males and females, even if in this sport the females show a higher frequency of injury than in any other discipline. However there are some sex differences in ACL injury rates among young skiers. Although ACL injury associated with meniscus and chondral injuries can lead to early development of osteoarthritis, not all knee injuries lead to the end of an injured skier's career: only 19% of ACL reconstructed skiers suffered a reinjury to the same knee, while 31% suffered of a bilateral ACL tears. Furthermore, skiers recovered from this injury have longer careers and better performance than non-injured skiers.

According to the literature review just cited, the ACL injury is the most common injury for professional skiers. Now our research focuses on confirming

that this type of injury is also the most common among amateur skiers, as we would expect. To avoid running into literature that is not about what we are looking for, we focused our research only on knee injuries of non competitive skiers.

Posch et al. [29] studied the knee injuries in an Austrian ski area during winter seasons 2016/17 and 2019/20. They questioned 282 non professional skiers that suffered of knee injuries, divided in $48,2\%$ males and $51,8\%$ females, with a mean average of 44 ± 10.1 years, 1.73 ± 0.08 m of mean height and 72.3 ± 10.3 kg of mean weight. They found out that any injury they analyze was an ACL injury, where 35.5% was an isolated ACL tear, while 64.5% was an ACL tear with other concomitant injuries, such as injuries to Medial Collateral Ligament (MCL) (50.5%), Medial Meniscus (MM) (40.1%) and Lateral Collateral Ligament (LCL) (22.5%). They highlight the fact that MCL, MM and LCL injuires are typically consequences of an ACL tear: when the anterior cruciate is injured, the knee is less stable and therefore other injuries arise more easily. In the majority of cases, the ACL tears are more serious than other lesions. They did not found important differences among sex, age and physical fitness, but an interesting aspect was found: females show more frequent ACL injuries with concomitant injuries, while males show typically only an isolated ACL tear.

In conclusion, we can confirm that ACL injuries are the most frequent injuries that occur on the ski slopes. The study reported in this thesis is therefore supported by the data exposed above and the final goal is to try to reduce as much as possible the accidents related to the knee.

2.1.1 Anterior cruciate ligament injury risk factors

Once we are satisfied that knee injuries are the most common among skiers, it is necessary to analyze the risk factors that lead to causing these injuries. Every risk factor allow us to understand if there is any correlation between accidents and other elements and they are typically divided between intrinsic and extrinsic: the former concern the age, gender and fitness of the skiers, while the latter focus more on environmental and equipment factors. It is very important to analyze these elements, since from them it is possible to understand how to act to improve the prevention of accidents.

Tarka et al. [36] identify as high risk factors for ACL injuries *"a parent with a history of ACL injury and lower strength in the upper body and legs"*. However, they did not consider either a high level of athlete fitness or fatigue as influencing

injury frequency and intensity.

Posch et al. [29] highlighted that the quality of the ski slopes is a fundamental element in the risk factors. However, according to them, the biggest risk factor comes from the equipment: carving skis, which are very narrow, can create very high bending and torsional forces, easily leading to injuries. Ski bindings are also very important: failure to release a ski can lead to very high torque which can lead the skier to very serious injuries.

Jordan et al. [19] conducted a large study about risk factors. As *Tarka et al.*, they found that there is a genetic link between adolescent who suffer ACL injury and their parental injuries history. However, unlike them, they believed that fatigue is another risk factor, since a more tired skier is more prone to injury than a rested one. They also claimed that a potential risk factor for skiers' injuries is the athlete's level of fitness, despite there is no statistical evidence. In support of this thesis, they noticed that a more trained athlete has stronger muscles that are less prone to injury. They also add that the skier's own technique is a source of risk for injuries: through the analysis of 20 videos, they discovered that the imbalances that lead skiers to ski are due to technical errors. Poor track conditions, visibility, high speeds, jumps and rapid changes of direction are believed to be some important environmental risk factors. In any case, they considered that the greatest responsibility in accidents is due to equipment risk factors. The shape of the skis, their length, the bindings and the boot stiffness are the most important risk factors. In fact, the shorter and narrower the skis, the faster the skier can go, at the expense of more torque exerted on the shin, which can facilitate knee injuries. In order to avoid many injuries, the FIS (from the French *Fédération Internationale de Ski*) has modified its ski size regulations, imposing minimum size standards to be maintained.

Finally, we highlight the very important fact that none of the papers we analyzed reported age or gender as a risk factor, as a demonstration that anyone is equally susceptible to be subject to injury while skiing.

2.2 The knee

The descriptive epidemiological research reported in the previous section, showed that the knee is the most injured body part for skiers, whether they are professionals or amateurs. So before we could design a device that can protect this joint, it is essential to understand what it looks like, what sensors we should use to recon-

struct its dynamics and where to place them. To do this, we exploited what has been exposed by *Neumann et al* in the book of *Kinesiology of the Musculoskeletal System*. Since this is an anatomy book and it is not our task to analyze in detail how a knee is made, only the topics of our interest are highlighted, taking care to avoid digressing.

2.2.1 Knee movements

The knee serves important biomechanical functions in the human body: it absorbs impact during walking, provides stability and transmits motion generated from the hip to the foot. Injury to ligaments is a typical consequence of excessive strain on the articulation, and anatomical knowledge is a fundamental prerequisite for understanding how injuries occur and how to prevent them. The knee is composed by the *lateral and medial tibiofemoral joints* and the *patellofemoral joint* and is anatomically included between the femur and tibia. It allows motion in two planes: flexion and extension in the sagittal plane, internal and external rotation in the horizontal plane (Appendix A). Movement in the frontal plane of the knee occurs only passively, limited to about $6° - 7°$.

Knee flexion and extension are performed around an axis of medial-lateral rotation. The term *flexion* refers to the approach between the calf and the thigh, and *extension* to their departure (Figure 2.1). The range of motion for a healthy knee is between $130° - 140°$ of flexion and $5° - 10°$ of hyperextension. During knee flexion, there is a slight migration of the center of rotation, causing not only knee rotation, but also a small shift to maintain contact between the femur and tibia (as the femoral condyles have an eccentric curvature).

Internal and external rotation of the knee occurs in a horizontal plane around a vertical axis of rotation. A knee flexed to $90°$ allows approximately $40° - 50°$ of total rotation; the range of motion of external rotation is twice than the internal rotation. When fully extended, however, rotation in the horizontal plane is almost absent. Rotation is blocked by passive tension of the ligaments and mechanical congruence between the bones.

2.2.2 Osteology and arthrology

At the distal end of the femur there are the *lateral and medial condyles*, which are small round bony prominences (Figure 2.2). The *lateral and medial epicondyles* protrude from each condyle, providing attachment sites for the collateral liga-

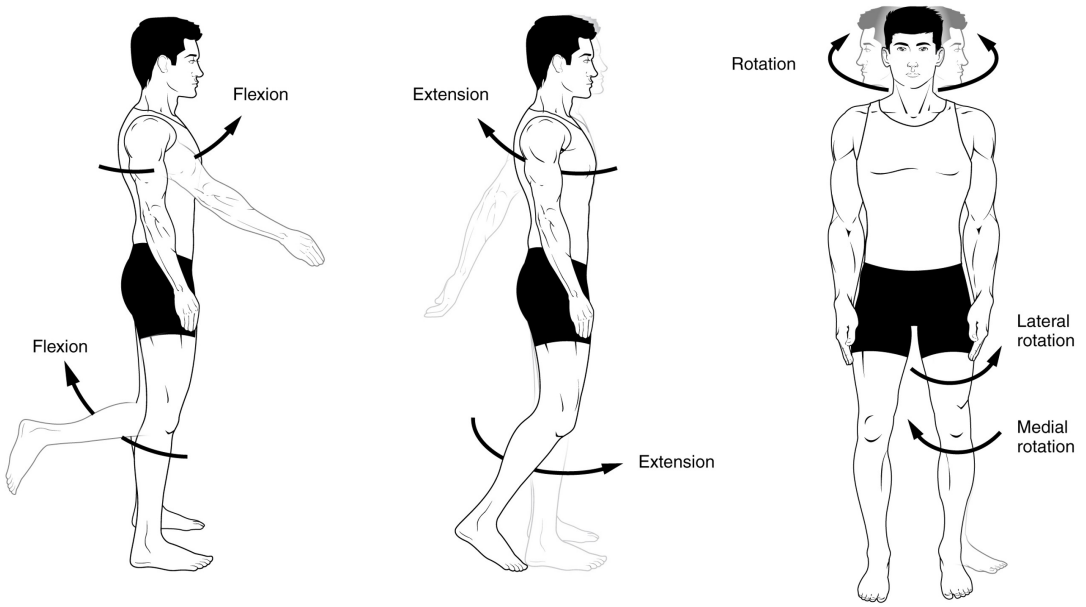

Figure 2.1: Image from [40] and [41]. Graphic description of the movements that a knee can perform: (left) flexion, (centre) extension, (right) rotation

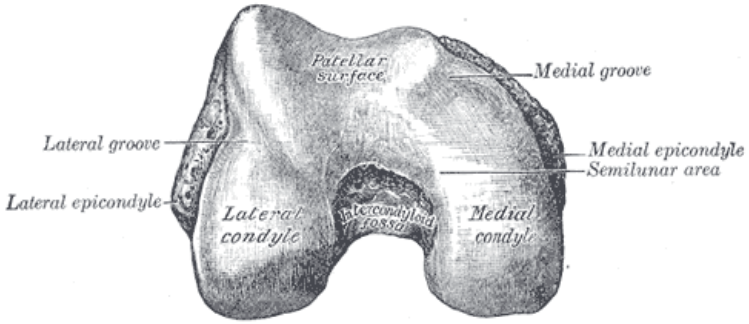

Figure 2.2: Image from [43]. Lower extremity of right femur viewed from below

ments. The passageway for the cruciate ligaments is called the *intercondylar notch* and separates the lateral and medial condyles. Note that a narrower than average notch may increase the likelihood of injury to the anterior cruciate ligament. On the tibia, there are two small imperfect housings for the femoral condyles, called *intercondylar eminences*, which are in turn separated by a shallow intercondylar fossa (Figure 2.2). The cruciate ligaments and meniscus connect at these intercondylar regions. Finally, to complete the knee, there is also the *patella*, which is an almost triangular shaped bone, embedded by the quadriceps tendons and placed frontally.

The axis through the femur tilts slightly as it descends toward the knee. This oblique orientation is due to the natural 125° angle of inclination between the femur and hip. The axis through the tibia is nearly horizontal, so the knee has a lateral angle of approximately $170° - 175°$. This normal alignment of the knee in the frontal plane is referred to as *genu valgum*. If this angle exceeds 180°, it is termed *gene varum*, if it is below 165°, it is called *excessive gene valgus*.

2.2.3 Ligaments

Ligaments are strong fibrous structures that connect two bones or two parts of the same bone together. Ligaments have a stabilizing function: they prevent particular movements or external forces resulting from trauma can alter the position of the structures to which they are connected. In the knee there are various ligaments, the most important of which are: *anterior cruciate ligament*, *posterior cruciate ligament*, *medial collateral ligament* and *lateral collateral ligament* (Figure 2.3).

The ones we are most interested are the cruciate ligaments. The term cruciate describes the physical relationship between them, which cross in the intercondylar notch of the femur (Figure 2.4). They are also attached to the tibia and both ligaments are thick and strong, reflecting their important role in providing stability to the knee. Acting together, the anterior and posterior cruciate ligaments resist all movements of the knee. However, they provide most of the resistance to antero-posterior shear forces between the femur and tibia. Injury to the cruciate ligaments can lead to obvious knee instability. Because the cruciates do not heal on their own, surgical reconstruction often requires a transplant. Although these reconstructions are fairly successful in restoring basic stability, the natural kinematics of the repaired knee almost never return to pre-injury levels.

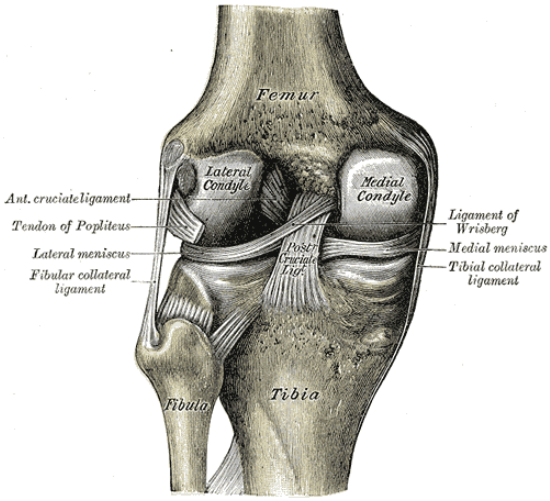

Figure 2.3: Image from [42]. Left knee-joint from behind, showing interior ligaments

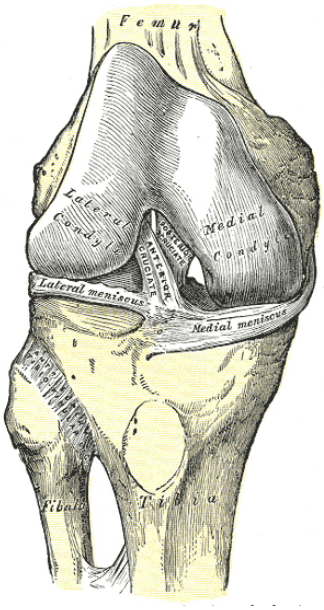

Figure 2.4: Image from [44]. Right knee-joint, from the front, showing interior ligaments

2.3 Knee injury mechanism

After presenting the *descriptive epidemiological* study and having exposed how a knee is made, we can introduce the *analytical epidemiological* research, that consists in the description of the mechanisms that generate knee injuries. To accomplish this task, a further search is carried out on search engines *PubMed, Pedro, Ovid* and *Google Scholar*. As explained in the Section 2.1, our research focused primarily on ACL injuries, since this type of injury is the most common for skiers. *Järvinen et al.* [20] found that there are three different mechanisms

Figure 2.5: Image from [37]. Representation of a valgus external rotation: the knee is externally rotated and the torque generated between the femur and tibia leads to ACL injury

that can cause knee injuries while skiing:

- **Valgus external rotation**
- **Flexion internal rotation**
- **Boot induced**

In the following we describe each of them.

Valgus external rotation

This type of injury is the most frequent and occurs when the skier falls forward: the foot is constrained to the ground and because the body is pushed forward by the force of inertia, the lower leg is abducted and rotated outward from the thigh (Figure 2.5). The valgus angle between the femur and the tibia is greatly increases as the ski considerably magnifies this torque acting as a moment arm. These movements lead to a rupture of the anterior cruciate ligament, although the medial collateral ligament is the one that is most stressed.

Flexion internal rotation

This type of trauma is rarer than the valgus external rotation, but manifests in a similar manner to it and in fact it is also identified as *varus internal rotation* or *slip catch*. Typically it occurs when the skier is turning and get out of balance;

Figure 2.6: Image from [37]. Representation of a flexion internal rotation: the knee is inernally rotated and the torque generated between the femur and tibia leads to ACL injury

the athlete extends the leg, attempting to reestablish grip with the outer ski. Then the ski impacts the snow surface and, since foot is bounded to the ground and the skier continues his forward motion, an inward rotation occurs relative to the athlete's direction of travel (Figure 2.6). The varus angle between the femur and tibia is accentuated, leading the ACL to injury.

Boot induced

This mechanism occurs as a result of a sudden retreat of the skier's center of gravity, with a consequent attempt by the athlete to reestablish the centrality of the attitude by levering on the rear edge of the boot. This situation can lead to injury in three different ways:

- *Falling back recovery*: the athlete loses balance, leans predominantly on a single ski and backs away from his center of gravity; the knee of the supporting limb is in hyperflexion and extrarotation, while the quadriceps contract maximally in an attempt to restore stability (Figure 2.7). This leads to overstretching of the ligaments, which ultimately results in injury.

- *Flat landing*: A technical error during any jump can cause a retreat of the barycenter during the landing that, if it occurs on ground with poor slope, leads to an increase in the energy of impact on the ground (Figure 2.8). Under these conditions, the backward movement of the center of

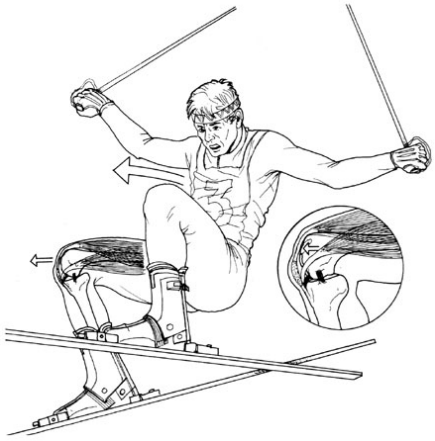

Figure 2.7: Image from [37]. Representation of a falling back recovery: the skier loses stability, bringing its center of gravity back, externally rotating the knee

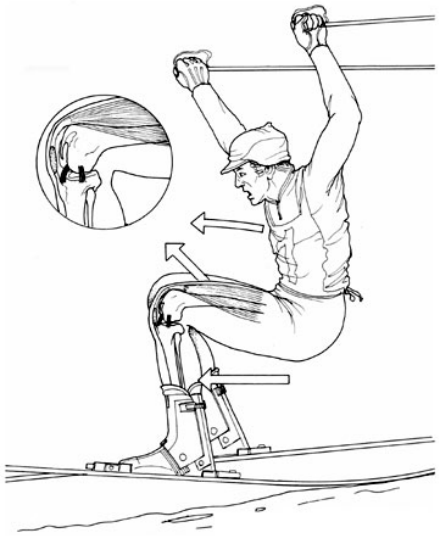

Figure 2.8: Image from [37]. Representation of a flat landing: the skier loses stability and and the force on the tibia opposes the force generated on the quadriceps, resulting in ACL injury

gravity causes a hyperflexion of the knee that is difficult for the athlete to counteract and can produce a rupture of the anterior cruciate ligament.

- *Steep landing*: The athlete, during the flight phase of a jump, moves back and tilts his center of gravity laterally, so that the first phase of the landing takes place discharging all the energy on the rear portion of only one of the two skis, that undergoes a significant deformation in bending. This asymmetrical loading on one of the two knees, completely hyperextended, pushed it forward, causing a relative slip between femur and tibia, producing the rupture of the ACL (Figure 2.9).

Figure 2.9: Image from [37]. Representation of a steep landing: the contact between the ski and the snow is increased by the weight force, which generates in a forward force on the tibia, causing the ACL to break

2.3.1 Etiology of anterior cruciate ligament injury for skiers

Jordan et al. [19] have analyzed the etiology of the ACL injuries, based on a biomechanics research. The anterior cruciate ligament can resist to anterior directly shear and and internal rotations applied between femur and tibia. Forces are developed within the knee by the quadriceps femoris, which can place stress on the ACL especially when it is completed extended. Simultaneously, the biceps femoris acts in conjunction with the ACL, generating a posteriorly directed shear moment on the tibia. These forces combine and can be amplified by both the skier's physical exertion and the equipment, leading to ACL injury.

Generally, relative rotation between the femur and tibia is involved in ACL rupture, however *Hame et al.* [15] have attempted to replicate the mechanisms that should lead to anterior cruciate ligament injury on thirty-seven cadaveric knee specimens. They applied three manual torque to the tibia (0 $N \cdot m$, 10 $N \cdot m$ internally, 10 $N \cdot m$ externally) with the knee bent at 0°, 90°, fully extended and hyperextended. Although the forces exerted in this study were applied quasistatic, resulting in much lower forces than those that occur while skiing, they give us a more complete view of how injuries occur. In fact, they discovered that hyperflexion is the main factor causing ACL rupture and not the twisting component of the fall, since it generates the highest torque. Similarly, hyperextension

of the leg, accompanied by torsion of the tibia, also easily generates injury to the anterior cruciate ligament. Therefore, ACL injuries are typically due to a combination of multiple factors, and when the knee is subjected to excessive stress situations, it is weaker, as expected.

Bere et al. [8] [9] conducted two studies in which they analyzed videos of injuries during World Cup Alpine Skiing to assess whether there are any elements of particular interest. These studies have only corroborated what has been presented so far, clearly identifying the three types of mechanisms explained above. However, they highlighted two issues that need special attention: first of all during the *steep landing* injury, the ACL tear is due to a combination of three factors and no relative rotation between the thigh and calf appears along the vertical axis. In fact, the injury is due to a tibiofemoral compression, combined with an advancement of the tibia and a retraction of the femur. This displacement could be see also as a two dimension rotation about the axis parallel to axis of the ski that impacts on the ground. This aspect therefore makes identification of this type of injury mechanism more difficult. The second aspect to consider concerns the mechanics of injuries during the *slip catch* phenomenon. In one of the cases analyzed, the knee flexion angle increased from 26° to 63° in 60 ms and, at the same time, the internal rotation of the tibia was 12°, bringing the valgus angle up to 13° before the peak tibial internal rotation. In the second case analyzed, again in 60 ms, the knee flexion angle increased from 39° to 69°, while the tibia underwent an internal rotation of 12° and the valgus angle reached 14° before the peak tibial internal rotation. Therefore, it has also been proved through the use of videos that this type of injury is accompanied by the combination of the two phenomena previously analyzed, i.e. rotation of the tibia and hyperextension of the knee.

Finally, to better understand how injuries occur when a *boot induced* mechanism arises, we exploited the research performed by *Bally et al.* [7], which studied a mathematical model to quantify the forces on the ACL. They, too, confirmed that what makes this type of injury very unique is, as mentioned above, that it manifests in two dimensions, whereas typically femur and tibia rotations occur in three dimensions. They also found that the force applied on the ACL during ski-to-snow impact is directly related to boot stiffness: the leather boots used in the past, which were very soft, generate less force than the newer models, which are stiffer. So this type of injury is closely related to the equipment and nowadays occurs more frequently than in the past.

2.3.2 Knee injuries prevention

The studies presented so far gives us a an idea on the dynamics of skiers' knee injuries. Before going into the core of this thesis, it is important to note that several methods have already been studied that aim to minimize injuries on the ski slopes. In Section 2.1.1 we discussed the ACL risk factors; it is obvious that reducing these factors decreases the probability of being subject to injury. But where it is easier to modify some factors, such as those related to equipment or fatigue or visibility, for others it is impossible to act, such as genetics or occasional errors.

Senner et al. [32] performed a literature review about equipment that can help skiers to reduce knee injuries. They found that the elements that are most easily improved are the skis themselves and the boots. According to them, it is possible to act on the skis in two different ways: either by changing their shape, or by improving bindings. Regarding the shape of the skis, it must be pointed out that they are designed to allow the athlete to control them as simply as possible. They must also be able to absorb any type of shock, be lightweight, glide on the snow and not cause damage to the skier when worn. Many studies, such as the one led by *Zorko et al.* [45], have shown that the wider the ski, the more probably is the chance to injure the knee if the ground is frozen, condition that easily occurs during competitions. Indeed, despite the fact that a smaller width leads to greater stress on the knee joint, it seems that this can be beneficial during descents on icy slopes. Without going into too much detail, a narrower ski prevents the knee from approaching the end of the range of motion in the transversal and frontal planes.

Another improvement that can be made on the ski would be to shorten it. Doing so would generate a shorter arm between the point of contact between the ski and the snow and the binding, greatly reducing the torque that is generated on the knee. Unfortunately, the ski could be shortened only in front of the foot, as the back allows the skier to maintain balance, but not excessively, because a ski that is too short would not allow adequate shock absorption and would make control more difficult.

Regarding boots, it was mentioned earlier that those currently on the market have too much stiffness protecting the shin and ankle. In theory, to decrease knee injuries, the height of the ski boot should be reduced, allowing the knee to be less prone to high torques; but this change could lead to more stress on the ankle, resulting in more injuries to this joint. Thus, as far as boots are concerned,

improvements cannot be made that would justify intensive studies on them.

Bindings, on the other hand, are one of the main causes of knee injuries (and not only), especially during falls. In fact, also as pointed out by *Fischer et al* [14], almost all ski accidents occur without the skis being properly released. The main purpose of the binding is to connect the boot with the ski and to release it if the leg is overloaded. They must be meticulously calibrated, since the binding allows the counterbalance of three moments and forces between the boot and the ski. However professional skiers tend to gauge bindings in such a way that they never come off their boots, in order to prevent them from losing their skis during competitions. To avoid skiers suffering serious injuries during falls because their skis are not released by the bindings since they are not correctly calibrated, new mechatronic bindings have been developed in recent years. Unlike purely mechanical ones, they have various sensors that detect measurements such as forces exerted and rotations, which are then communicated to a microprocessor that can decide whether it is appropriate to activate the release of the ski. This technology has undergone considerable development recently, and various algorithms and methods have been proposed to decide whether activation is necessary. For instance, *Hermann et al.* [17] suggested a set of input parameters that would be best to include when designing a mechatronic binding, such as: knee angle, leg muscle activity, forces and torques exerted on the foot, skier's speed and parameters about the individual (age, gender, weight, etc...).

In conclusion, the problem related to knee injuries in skiers is well known and, over the years, efforts have been made to minimize the number of accidents related to this discipline. With the work presented in this thesis, we will attempt to find a way to further reduce injuries that occur on ski slopes, trying to develop a method to achieve this goal, but without hindering skiers and trying to remain as economical as possible, without sacrificing reliability.

2.3.3 Effect of knee pads on reducing ACL injuries

A commonly used tool to reduce knee injuries is the knee pad. Thus, we could think that the use of this equipment could lead to a reduction in the number of injuries recorded by skiers. Furthermore, this garment could be used for our final purpose, acting as a chassis for the sensors we would like to design. We performed a small literature search to try to answer this question. We found that the studies on this topic are very contradictory, but we can say that knee supports are most probably not useful for reducing the number of knee injuries in skiers.

An example of those who support the fact that they are useful is given by *Negrin et al.* [25], who asked themselves whether knee braces really prevented knee injuries, and to do so they performed a review of the available literature. They focused on both strengthening knee braces for those who have never been injured and those designed to help who have already suffered a knee injury. Biomechanically, no major differences were seen between wearing or not wearing a knee brace: strain on the ACL was reduced, but not at a level high enough to justify extensive use of knee pads. In relation to this, it was noted that skiers who wore one manifested a proprioceptive change that made them more confident, but nothing can be confirmed with objective data. With regard to skiers who had already suffered an ACL injury, it was seen that if they wore a knee brace, they could reduce the chance of an injury recurring. However, the studies that have shown this aspect are affected by minor form flaws: the groups of skiers on which this research was done were not properly randomized, making this study unreliable.

Sterett et al. [35], on the other hand, primarily analyzed the effect of using a knee brace for skiers who had undergone anterior cruciate ligament reconstruction in the past two years. Despite the fact that the study they carried out has some problems because the athletes involved in the research were conditioned, they led to results that have been confirmed by other researchers, namely that a knee brace can reduce the probability of suffering a further injury to the anterior cruciate ligament.

Finally *Moon et al.* [35] carried out a very important study, to actually evaluate the effect that a knee brace and knee sleeve have on the force exerted on the ACL during a jump. The disconcerting discovery they made was that athletic performance, shear force, internal rotation moment and the ACL force are not altered at all by the use or non-use of the two supports, while there is a reduction in maximum flexion, adduction and abduction when worn. Their conclusion is that if someone could develop a knee brace that could control the shear force to which the knee is subjected and the moment of internal rotation, then it would be of practical use, but until then there is no need to wear it. This last study is extremely useful because, although it was conducted in a situation under laboratory control, it allows us to understand quite clearly the forces to which a knee is actually subjected with and without a knee brace, which are difficult data to calculate. In addition, we know that ACL tears also occur due to high torque on the ligament, and if the knee pads currently on the market are unable to reduce this force, then they cannot be used as a form of prevention.

In almost all of the studies shown so far, the wearing of knee pads by skiers has been found to be particularly annoying, because it tends to shift from its initial position. Thus, this tool represents more of an impediment rather than a help. Despite the problems just discussed, we could anyway choose to use a kneepad as a support for an active safety system, because if it would effectively reduce injuries in skiers, maybe its use could be reconsidered. Furthermore, we could use it as a chassis to easily attach sensors without making the tool we would like to design too complicated to wear.

2.4 Practical feasibility of active safety systems

As presented previously, the aim of this thesis is to develop an instrument capable of monitoring the dynamics of the knee and this mechanism must be integrated into a wider project, such as protecting skiers from injury or be able to provide medical information about the type of injury sustained immediately after the accidents occurred. In particular, a more comprehensive project could start from the idea of using the here developed knee monitoring system in an active control system, based on the design of a knee brace equipped with an airbag, so that it would activate when certain conditions met in order to prevent serious injuries to skiers. Analyses discussed in the previous sections about the knee dynamics pose serious problems, that should be deeply investigated in future, about the effectiveness of active protection systems. To better explain this observation, it is more convenient to see the kinematic chain of a person. Specifically, the lower limbs of a skier, form a closed kinematic chain, where the articulations of the knees, ankles and hips can be modeled as spherical joints with a limited range of motion (Figure 2.10). The thighs, pelvis, and shanks are the links between the joints: the hip joint connects the pelvis and thigh, the knee connects thigh with shank, and finally the ankles connect the shank with the ground.

Regarding the degrees of freedom of the joints, we can say that in a normal situation, the hip has 3 DOFs, the knee 1 DOF (2 if it is flexed) and the ankle 3 DOFs. According to *Ren et al.* [30], Table 2.1 shows a summary of the movements that each joint can perform, with the corresponding range of motion.

However, a skier has constraints that prevent him or her from making all the movements that a normal person can do: the hip can only rotate while the ankle now has impediments in every direction, being bound to the ski by the bindings (Figure 2.11). Therefore, the degrees of freedom for a skier are limited to 1 for the

Joints	DOFs	Movement	ROM
Hip	3	internal/external rotation	$-120°$ to $65°$
		abduction/adduction	$(-30°$ to $-35°)$ to $40°$
		extension/flexion	$(-15°$ to $-30°)$ to $60°$
Knee	1 (2)	extension/flexion	$(-10°$ to $0°)$ to $(120°$ to $160°)$
		internal/external rotation	$-10°$ to $(30°$ to $40°)$
Ankle	3	internal/external rotation	$(-15°)$ to $(30°$ to $50°)$
		abduction/adduction	$(-20°)$ to $(40°$ to $50°)$
		extension/flexion	$(-30°$ to $-35°)$ to $(15°$ to $20°)$

Table 2.1: Movement form and motion angle range for hip, knee, ankle joints

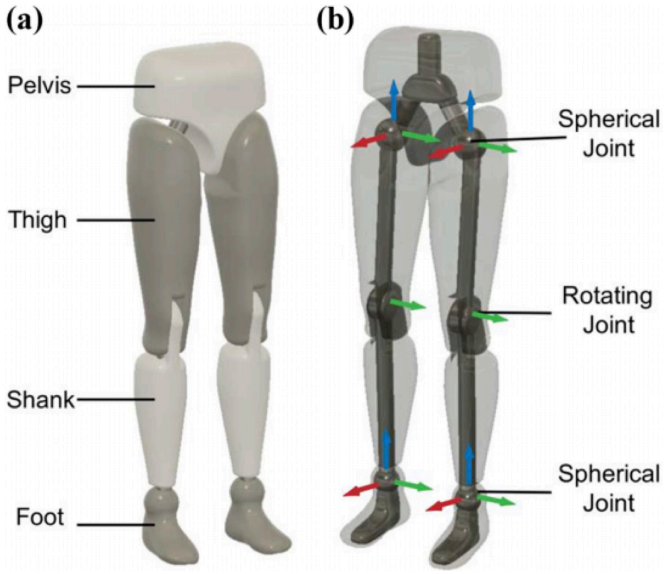

Figure 2.10: Image from [30]: human lower limb chain modeling. (a) The hierarchy of rigid bodies represents for the human skeleton. (b) Joints with the specified axes as the reference revolution axes of each DOF. DOF: degrees of freedom

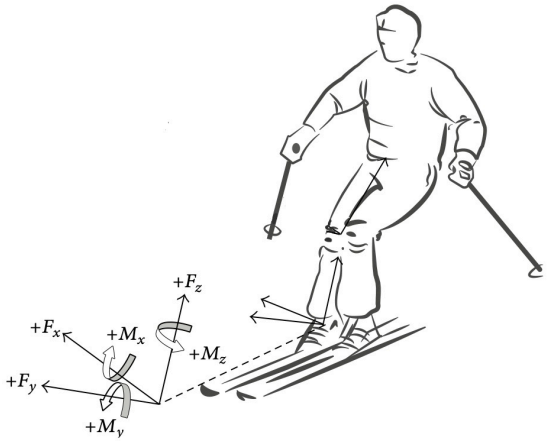

Figure 2.11: Image from [21]: Definition of the local constraints at the ankle for a skier: it has constrained reactions that prevent it from any linear displacement or rotation

hip and knee and 0 for the ankle. Assuming that there is a knee airbag and that it triggers when a potentially dangerous cruciate ligament situation is detected, it could be that it causes worse injuries than it is supposed to prevent. Let, for example, that the skier is subject to *valgus external rotation* and the system registers this dangerous situation. In theory, the airbag should activate, allowing the knee to stiffen to prevent it from being subjected to extreme forces. However, the movements of the articulation are limited by the fact that the foot of support cannot leave the ground since it is the only point of contact that remains to the skier and, at the same time, the athlete who is falling does not have the ability to recover the position of balance using the other leg (if this were not so, there would not be this dangerous situation). So the knee is actually impeded in all its movements and has, in reality, no degree of freedom that would allow it to extend. Any eventual attempt to straighten it, could result in much more serious injuries than the one you would like to avoid.

Even in the event that the skier should find himself in a risky situation that could lead to a *boot-induced* knee injury, the activation of the airbag would lead to a straightening of the joint that would be physically impossible (since it is in hyperflexion), risking further injury to a leg that has already inevitably suffered a rupture of the ligaments. Similar situations can be found in other hazardous situations: the extreme case can be found by analyzing *steep landing* injuries. In this case the knee is in hyperextension. The eventual deployment of the airbag would lead to an attempt to straighten the joint that is actually already completely upright, and the consequences would be disastrous for the unfortunate skier.

There are cases where this mechanism could be of considerable help: if, for example, there was the possibility of recovering one or more degrees of freedom along with the airbag deployment, then maybe this would justify the adoption of this system. But in order to do so, it would be necessary to develop technology that would allow ski release. Thus, evaluating the costs and benefits of adopting an airbag system embedded in a sensorized knee pad, it would be best to avoid this solution. If, on the other hand, we simply wanted to try to limit the number of skier injuries without them having to resort to mechatronic bindings, then we have to fundamentally change the system we want to design. An idea could be to design a mechanical system, perhaps pneumatic, that when it detects a dangerous situation, stiffens the knee, but without depriving it of any degree of freedom. This system should be able to detect at any time the dynamics of the knee and

to recognize a limit situation. Obviously, such a system should not disturb the skier, neither when it is active, nor when it is not working. Or, another solution, could be to create a mechanism that can constantly support the leg, decreasing the load on the knee. However, such an application could be despised because it could be seen as an aid to the skier during competitions, and therefore it would be better to avoid it, which instead could be well appreciated by amateur skiers, if not too expensive.

Chapter 3

System design and prerequisites

The purpose of this thesis is to design a system that allows us to reconstruct knee dynamics in real time. In this chapter we outline the minimum requirements that our system must meet, through an analysis of the characteristics it must satisfy. This step is critical to understand what kind of performance our system should have to be considered reliable and valid. Following, we propose an initial idea of the system we want to design, focusing in particular on the sensors we could use. We then analyse the state of the art in knee dynamics reconstruction by conducting a literature search, examining in depth the articles that could be most useful to us. This last step is very important, because if there are some studies that can be used, it would greatly reduce the time and effort needed to achieve our purpose. Finally, we introduce the mathematical and statistical tools that we consider useful for our purpose of reconstructing the dynamics of a knee. In this phase, we expose the theoretical bases necessary for understanding chapters that follow, leaving reference to specific texts for further details.

3.1 Specifications of the system

Understanding what system specifications we need is necessary to assess which quantity needs to be measured and which conditions the system must fulfil to be considered a good solution. The requirements that the project must meet can be divided into three categories:

- **system specifications**: they concern the technical aspects of the system to be designed, such as sampling time, response time, bandwidth, reliability, calibration, etc;

- **shape specifications**: such as the wearability and comfort of the system, that must resist impact, must protect the skier's knee, does not disturb the athlete while skiing, must not damage him during possible falls, etc;
- **secondary specifications**: these specifications refer to those aspects that are not strictly related to the project, but that must be taken into account after the first prototypes have reached minimum standards and they include the battery life, data storage, wireless connectivity for external monitoring in real time, etc.

For obvious reasons, in this thesis we mainly deal with the first ones, while we leave to future works the task of developing the other points.

The first and most important specification we have to meet is that the system we want to design must be reliable and secure. No skier would want to wear an active safety tool that acts up at the wrong time, causing them discomfort and potential falls. At the same time, if it is designed to operate in certain situations, it must be reliable and do its task whenever it detects a dangerous situation. Therefore, it must not detect false positives or false negatives and must be triggered whenever a potential injury situation is about to occur. In fact, it have to be activated before the injury has occurred and not after, otherwise it completely loses its usefulness. Using the information from the previous chapter, especially the knowledge about how accidents occur, we know approximately how long it takes for accidents to occur (Section 2.3.1). Our system must be such that the sampling rate is high enough to detect this time interval: if an injury occurs roughly in 0.06 s, it means the minimum frequency to detect it should be equal to $f = \frac{1}{0.06\ s} \simeq 16.67\ Hz$. However, a system at frequency f is not sufficiently efficient for two reasons: first, it cannot act before the accident occurs, and second, it is statistically almost impossible for this sampling rate to detect exactly the instant when the accident occurs. So we need at least twice the sampling rate to try to detect exactly when the accident happens. But even so it may not be enough, because in reality we do not have exactly a precise time interval to measure and that the injury occurs in 60 $[ms]$ is only our assumption based on the data at our disposition. As noted earlier, injuries do not occur in a single instant, but they are the composition of several factors that are created within a few seconds. We can use this element to prevent accidents so that our safety mechanism is triggered promptly. To be sure that our system is most likely reliable, a sampling frequency of at least 1000 Hz is required, but higher frequencies are also acceptable.

Obviously the response time of the system is related to all of the above, and it

is given by the sum of the response time of the sensor that makes the measurement and that of the microprocessor. It must be as small as possible, to avoid delays between the sampling of the signal and its processing by the microprocessor. Fortunately, even for systems that are not real-time, the response time of the signal processing is negligible compared to that of the sensor, as it is always much smaller. So the choice of sensor becomes even more important than it was before.

The last system specification that our project has to comply with concerns calibration: this aspect is of fundamental importance because it allows the measurement to be reliable, but the process of calibration can be long, complex and annoying for a user. So the ideal would be to design a tool that does not require to be calibrated at all or, if this is not possible, that allows to be calibrated with a very simple and fast procedure. Connected to the last topic is the question of the easiness of wearing the system and the fact that it should be considered reliable even if it is not worn perfectly, as long as it is not worn completely wrong.

Other specifications could be included among those already analysed, but to a first approximation we can be satisfied if we are able to design a system that fulfils all the conditions just examined. Therefore the initial part of our project consist in developing a prototype of the system that can satisfy the specifications we have just seen, in order to understand which could be the problems related to a similar device, before passing to the real design of the tool.

3.2 Sensor selection and state of the art

In the previous chapter, Section 2.3, we saw that anterior cruciate ligament injuries generally occur due to the concomitance of two factors: hyperextension or hypereflexion of the knee and relative rotation between the femur and tibia. It is natural to think that the use of sensors capable of measuring the position of the thigh and tibia is the most correct choice. Consequently, we chose to use inertial sensors, such as accelerometers and gyroscopes. Since the relative rotation between above and below the knee is crucial, the possibility of placing one sensor at the height of the femur and another on the tibia was evaluated. Many other sensors could serve our purpose, such as strain gauges, maybe placed laterally with respect to the patella, or bioimpedance, which measure muscle signals and calculate knee flexion, or even fibre optics, which change the reflection of light inside them depending on how they are bent. However, we decided to use iner-

tial sensors for several reasons: firstly, they are readily available, low cost and offer excellent performance. Secondly, they can simultaneously assess both knee flexion and relative rotation between the thigh and shin. They are also easy to use, requiring no special prior knowledge or complicate calibration procedures. Finally, there are many studies on how to perform sensor fusion to retrieve the data we need.

Once we know what kind of sensors we could use, it is necessary to figure out how and where to place them and then how to use the measurements. Taking advantage of what was presented in the previous chapter, we choose to attach these inertial sensors to a knee brace so that we could easily apply them to the parts of the body that we are interested in. Also the number of these sensors is also very important, as we may only need two or more. However, in order to answer these questions, it is necessary to carry out experiments, so these topics are discussed more in detail when we explain the practical experiments we have conducted. After careful consideration of the availability and cost of the sensors, we choose to use a minimum of two and a maximum of four sensors. What we already know to be necessary is that one sensor (or two, depending on the number used) needs to be applied to the thigh, while the other sensor (or two) needs to be applied to the shin/calf.

To accomplish our final aim, we decide to divide the problem of monitoring knee dynamics into more and more complicated subproblems each time. First, we opt to use only simple analogue accelerometers to calculate the Range of Motion (ROM) of the knee. This step is done to understand where we could place the sensors and how many would be convenient to use. Subsequently, the design is improved by using digital sensors, exploiting sensor fusion algorithms, to obtain more accurate results.

Since inertial sensors are widely used in the medical and sports fields, we conducted a brief literature research in order to find out which technologies are already present in the reconstruction of knee movements. In particular, our research focused on the methods needed to identify or reconstruct the knee's flexion angle and the further movements it is capable of making. We found that there are already lots of techniques to reconstruct the dynamics of a knee, but most of them use IMUs in combination with other sensors. Many of them use manual techniques to correctly position the sensors on precise points of the leg, but we can not use these techniques, as our aim is to obtain a device that works as soon as it is applied, regardless of where the sensors are located. Generally, these methods

are exploited to reduce errors from inertial sensors and to increase the reliability of measurements. Our purpose is still that of creating a device that uses only the IMU to reconstruct the dynamics of the knee, for the reasons explained above.

Vargas-Valencia et al. in [38] exploited measurements coming from two IMUs and from an intensity variation based *Polymer Optical Fiber (POF)* curvature sensor to compute the flexion angle of a knee. The technique they developed, as well as many others, avoids the use of the magnetometer and combines data from several sensors by using a filter, in this case an extended Kalman filter. The method they proposed is very interesting, but the use of fibre optics may not be optimal for a skier: in the event of a fall, it may be subject to deformations that are not strictly linked to the movement of the knee and may therefore give invalid values.

Faivre et al. in [13] were some of the first to propose portable technology to measure the flexion angle of a knee. Since theirs was one of the first studies conducted, their results are not considered reliable enough today, as their techniques have been surpassed by more advanced and reliable technologies. It is very interesting the fact that they chose to use only gyroscopes and not other sensors. It should also be noted that their study was carried out to assess the health of a knee that had been subject to ACL surgery. This study was aimed at finding out whether it was possible to reconstruct the dynamics of a knee after it had undergone surgery, and it was not their intention to study the behaviour of their device in the sporting environment. Nevertheless, it is worth mentioning because it is based only on sensors that measure the angular velocity of the thigh and shin, and this information may be useful for our purpose.

A very similar work to that carried out in [13] was developed by *Bravo-Illanes et al.* in [10]: they designed a sensor embedded knee brace (with IMU) capable of evaluating the kinematics of the knee after injury to the anterior ligament. An important feature they have worked on is that their system does not need to be calibrated and they do not use magnetometers either. Unlike other similar studies, they use quaternions to calculate the orientation of the sensors and then they use the information from the IMUs to calculate pitch, roll and yaw angles. This process allows for a reasonably low computational cost and the evaluation of knee flexion angle can be done very quickly and in real time. However, they point out that the algorithm they have developed shows an error that becomes larger and larger the faster the movement is executed. Therefore, although this method may be promising for our purpose, it is not usable unless radical modifications

are made to allow acceptable reliability even with high-speed movements.

Cooper et al. in [12] created an algorithm that uses data from accelerometers and gyroscopes to assess knee flexion angle. What distinguishes them from other studies is that they validated their work both by analysing a patient walking and by analysing the same patient while running. The results are very promising, as only 3.4° of error was calculated while the subject was running, in contrast to a less dynamic condition, such as walking, where errors of the order of 0.7° were recorded. Unfortunately, also this study has limitations, as pointed out by the authors themselves, including the fact that the knee is considered a perfect hinge joint, which does not allow relative translation between the upper and lower parts. This simplification is excessive for us, as we have shown previously that the knee joint is also prone to injury when relative displacement occurs between the femur and tibia. Despite this, we can still use the information in this article on how roll and pitch angles are calculated.

Bakhshi et al. in [6] have presented a very interesting approach to measure and monitor human body joint using inertial sensors. Their aim was to develop a device for continuous monitoring, capable of helping therapists with the active remote assistance of patients. This paper presents a measurement system based on IMUs, in particular on accelerometers and gyroscopes: the former provide the necessary measurements, while the latter correct the data acquired with the accelerometers. Finally, the data collected in this way are sent to a computer via Bluetooth protocol for further analysis. This article is very interesting for us, as they have developed a system very similar to the one we would like to design ourselves, with the main difference being that ours should be used while practising sport, and so in much worse conditions than those considered in this article. We therefore proceeded with a more detailed analysis of their work, in order to see if what *Bakhshi et al.* have developed can be useful to us. In this paper they used two digital IMUs, each consisting of a 13-bit resolution triaxial accelerometer, a triaxial gyroscope and a triaxial magnetometer. The first IMU was placed above the knee, the second below, and each device was connected to a computer via a Bluetooth protocol, as each sensor was wireless and powered by a 3.6 V battery. Each sensor was previously calibrated on a surface is parallel to the ground and assumed that thigh and shank segments were in the same plane. They calculated the angle γ formed by the thigh as the angle between the vector indicating the force of gravity and the resultant coming from the sensor placed on the thigh. In the same way, they calculated the angle λ formed by

the shin as the angle between the vector indicating the force of gravity and the resultant coming from the sensor placed on the shin. They then corrected the data from the accelerometers with the calculated data from the corresponding gyroscopes. Finally, to calculate the knee flexion angle, considering the axis parallel to the ground as the 0° reference, they subtracted from 180° the sum of the thigh and shin angles calculated as above, i.e. $flex = 180° - (\gamma + \lambda)$. Finally, they validated their work by applying the designed devices to a person and verifying the difference between the angles measured by the tools and the angles measured by a sophisticated motion capture system.The data obtained showed very valid results, as the maximum error measured was 3.06°. The paper just presented is really very interesting, as it allows to have a very good result using a low cost, easy to implement and wireless device. What we found difficult to implement, but at the same time was not considered a problem by the authors of this article, concerns the synchronisation of the devices applied to the leg, as they do not have a physical connection between them and necessarily need another type of synchronisation, which can only be done via software, but it is difficult to apply. Another element that we found hard to understand concerns the method used to calculate the knee flexion angle, since the geometry they answered is only applicable in certain situations, but the authors did not explain in depth a standard procedure to be able to carry out the measurements. Finally, the last point that does not allow us to use the same technique exploited in this article, lies in the fact that the calculation power is delegated to a device external to the measurement system, leading to delays between the execution of the movement and its detection, a factor that it is vital for us to keep synchronized.

Seel et al. in [31] designed a device that could analyse a human's gait and knee flexion angle, using a combination of accelerometers and gyroscopes. They put a pair of sensors above the knee and, in a mirror image, applied similar ones to the shin. Their aim was to improve the technology available until then, by avoiding calculating in advance where to place the sensors on the leg. Consequently, it became necessary to introduce a calibration procedure, which would allow the sensors to understand their orientation. This procedure was done by using the data coming from the gyroscopes and then combining these measurements with those coming from the accelerometers, by using a complementary filter (or a Kalman filter). So they calculate the knee flexion angle through the rotation matrix, which expresses the difference between the orientation of the two sensors. This method has many aspects that we can take into account in the

development of our application, but at the same time we cannot use exactly the same techniques, since from a computational point of view this procedure is very expensive: at each instant a cost function is evaluated and then minimised. This implies the need of a very high computational capacity, which is reflected in the final cost of the device. In addition, similar results can be obtained by exploiting the layout of the sensors, without necessarily knowing their precise position or orientation, but relying on a priori knowledge.

McGinnis et al. in [22] developed a system based on accelerometers to assess the range of motion of a knee, since this parameter can be used to evaluate a person's state of health and to understand any complications arising from surgery. Their aim was to design a system that would avoid taking manual measurements on patients and allow anyone to take them, as they typically require experienced staff. The system they developed consists of two digital three-axis accelerometers, one placed on the thigh and the second on the shin. These sensors are synchronised and operate at 50 Hz. The recorded data were downloaded from a customized application and finally analysed offline. Once the sensors have been applied to the patient, they have to be calibrated through a short procedure where the user must first stand still for 10 s and then walk slowly for 30 s seconds. After that, the patient must perform a maximum knee extension test and a maximum knee compression test and then can position the patient at 45° and 90°. Each of these tests was compared with a manual measurement of the knee angle performed using an anatomical protractor. The calibration procedure is extremely useful as it is unknown beforehand how the sensors have been oriented. Consider Figure 3.1, where the axes of the anatomical reference systems of the thigh and shin are shown, respectively $(\hat{i}_T, \hat{j}_T, \hat{k}_T)$ and $(\hat{i}_S, \hat{j}_S, \hat{k}_S)$. They then considered the measurements made by the sensors $\vec{\alpha}_{T,S}$ as a combination of translation $\vec{a}_{T,S}$ acceleration, gravity $\vec{g}_{T,S}$ acceleration and white noise $\vec{w}_{T,S}$

$$\vec{\alpha}_{T,S} = \vec{a}_{T,S} + \vec{g}_{T,S} + \vec{w}_{T,S} \tag{3.1}$$

In the calibration phase where the subject was standing still, the translational component was assumed to be negligible compared to the gravitational component, i.e. $||\vec{g}_{T,S}||_2 >> ||\vec{a}_{T,S}||_2$, allowing the vertical direction $\hat{j}_T$ and $\hat{j}_S$ to be defined as the opposite direction to the data from the initial calibration phase. In the second calibration phase it was then possible to define the direction of the axis $\hat{k}_{T,S}$, assuming that the walk takes place in the $(\hat{i}_T - \hat{j_T}, \hat{i}_S - \hat{j_S})$ plane. In fact, during the walk, it is possible to identify the direction of the K axis as the

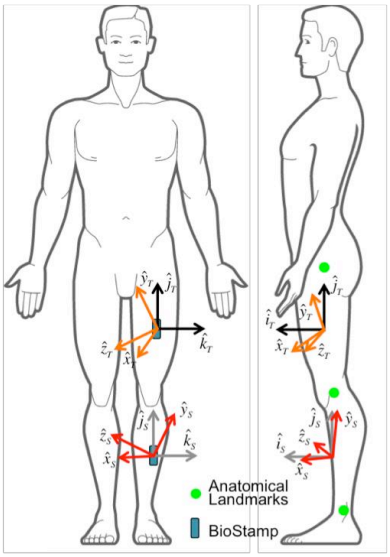

Figure 3.1: Image from [22]. Definition of the position used for the sensors and corresponding frames

main direction, i.e. the direction in which the greatest acceleration occurs. In other words, it is the direction along which the translational component overcomes the gravitational component, namely $||\vec{a}_{T,S}||_2 > ||\vec{g}_{T,S}||_2$. The last axis $\hat{i}_{T,S}$ was calculated by means of the vector product of the previous two, so that the reference system obtained is right-handed. Once the orientation of each sensor was obtained, it was possible to calculate the thigh and shin angles simply by evaluating the difference between the resultant of the corresponding sensor with the previously obtained axes, using the mathematical tool of *direction cosines*. Finally, the knee flexion angle was obtained by subtracting the shin angle with the thigh angle. The results obtained by this technique were compared with measurements taken manually by experts. The outcomes obtained showed to be very reliable, in particular a correlation of 0.99 was found between the measurements made with the system designed by the authors of the article and the measurements made with the anatomical goniometer.

Finally, a very interesting technique that could be taken as a starting point for further improvements is the one developed by *Hermann et al.* in [16]. They designed two pants for measuring knee flexion using piezoresistive or capacitive sensors. These pants require a short calibration procedure and are then able to accurately return the knee flexion angle. Eventually this value can be communicated to the ski bindings, and if the data measured are so high that they exceed the safety limits, then the bindings should open, allowing the skis to be

released. Conceptually, the idea behind it is very similar to the one we would like to develop, but the technology behind it is completely different, showing that similar results can be obtained also using totally different techniques. An important limitation of their work is that the results they have obtained show an error that is too high compared to studies with the same target, namely reconstructing the knee flexion angle. The technology developed by them, even if it presents unacceptable results for us, is very impressive because it is integrated directly into the garment, making more convenient to use a mechanism that can perform calculations in real time.

3.3 Useful theory for knee dynamics reconstruction

After choosing the type of sensors we would like to use, it is necessary to introduce mathematical and statistical tools that allow us to interpret and analyse the acquired data. The following mathematical instruments permit us to describe a three-dimensional body in space and to give it an orientation using various techniques and representations. Statistical tools, on the other hand, allow us to reconstruct data by removing uncertainties and disturbances, such as drift in gyroscopes. Hereafter, only the information necessary to understand the techniques we use to reconstruct the dynamics of the knee will be given, without going into detail.

3.3.1 Pose of a rigid body in three dimensions

By using [34] and [11] we can define a *rigid body* as a solid body where the deformation is zero or so small it can be neglected and where the distance between two given points remains constant in time independently on the transformations the body could be subjected. A rigid body is completely described in space by its position and orientation, typically defined as *pose*, with respect to a reference frame. Let O_w be the center of the *world frame*, that is an orthonormal reference frame, with X, Y, Z be the unit vectors of the frame axes, as shown in Figure 3.2. To describe the rigid body orientation, we can consider an orthonormal frame attached to the body (*body frame*) and express the rotation of this frame with respect to the reference frame by means of the so called *rotation matrix* $\boldsymbol{R}$, where the columns of it represent the coordinates with respect to the reference frame of the axes of body frame, namely X', Y', Z'.

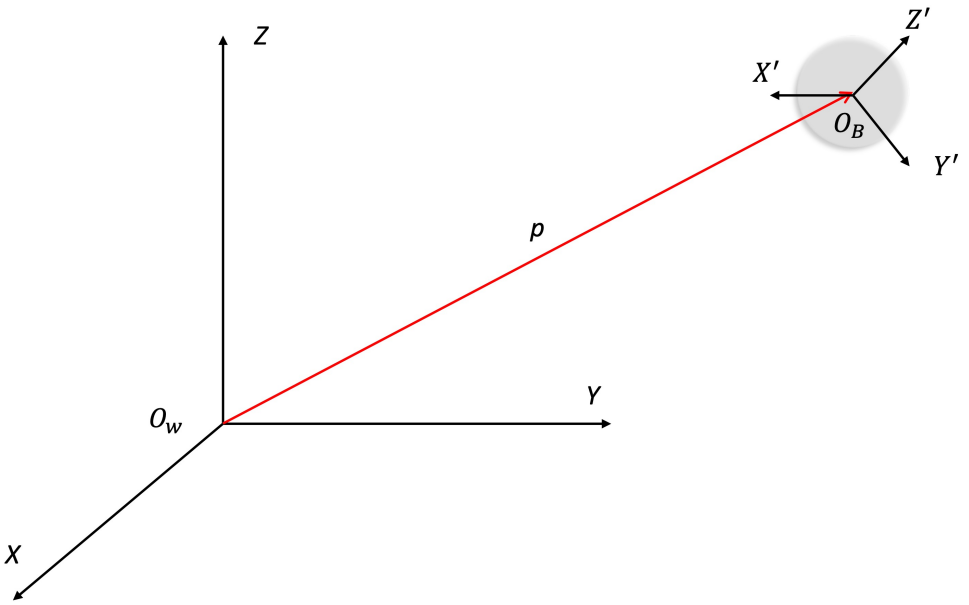

Figure 3.2: Graphical representation of position and orientation of a rigid body

The rotation matrix is an *orthogonal* matrix, i.e.

$$\mathbf{R}^T\mathbf{R} = \mathbf{I_3} \tag{3.2}$$

where $\mathbf{I}_3$ denotes the (3×3) identity matrix and if we post-multiply each side of the previous equation by $\mathbf{R}^{-1}$ we obtain

$$\mathbf{R}^T = \mathbf{R}^{-1} \tag{3.3}$$

Note that if $det(\mathbf{R}) = 1$ the frame is right-handed, while if $det(\mathbf{R}) = -1$ it is left-handed.

Consider now the frames obtained via *elementary rotations* of the frame about one of its axes. These rotations are positive if they are made counter-clockwise about the relative axis. We can distinguish three elementary rotations:

$$\mathbf{R}_Z(\alpha) = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.4}$$

$$\mathbf{R}_Y(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \tag{3.5}$$

$$\mathbf{R}_X(\gamma) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{bmatrix} \tag{3.6}$$

which correspond, respectively, to a rotation of the reference frame by an angle α about the Z axis, an angle β about the Y axis and an angle γ about the X axis. By analyzing previous equations (3.4)-(3.6), we can easily note that

$$\mathbf{R}(-\delta) = \mathbf{R}^T(\delta) \tag{3.7}$$

We introduce the rotation matrices because they have an important a geometrical meaning, that is the matrix $\mathbf{R}$ describes the rotation about an axis in space needed to align the axes of the world frame with the corresponding axes of the body frame. Let consider a point P in the space. It can be represented using its coordinate $\mathbf{p}$ with respect to a frame $\mathcal{F}$ or with respect to another frame $\mathcal{F}'$ by using $\mathbf{p}'$. It follows that

$$\mathbf{p} = \mathbf{R}\mathbf{p}' \tag{3.8}$$

where the matrix $\mathbf{R}$ represents the transformation matrix of the vector coordinates from frame $\mathcal{F}'$ into the coordinates of the frame $\mathcal{F}$. By (3.3), the inverse transformation is given by

$$\mathbf{p}' = \mathbf{R}^T\mathbf{p} \tag{3.9}$$

3.3.2 Euler angles

The rotation matrix describe the orientation of a frame by nine elements, which are not independent. By the orthogonality propriety (3.3), the elements of the rotation matrix are related by six constraints that yield this representation a redundant description. In fact, a minimal representation of the special orthonormal group $\mathcal{SO}(m)$ requires $m(m-1)/2$ parameters, thus three parameters are needed to parameterize $\mathcal{SO}(3)$, whereas only one parameter is needed for a planar rotation $\mathcal{SO}(2)$. Therefore, it follows that a minimal representation of orientation can be obtained by using a set of three angles. As a result

Theorem 1 *A generic rotation matrix can be obtained by composing a suitable sequence of three elementary rotations while guaranteeing that two successive ro-*

tations are not made about parallel axes.

This implies that 12 distinct sets of angles are allowed out of all 27 possible combinations, each set representing a triplet of *Euler angles*. In the following paragraphs we focus about ZYX angle set, also called *Pitch–Roll–Yaw* angles, to denote the typical changes of attitude of an aircraft in the aeronautical field. In this case, the angles $\Phi = [\rho, \phi, \theta]$ represent rotations defined with respect to a fixed frame attached to the centre of mass of the aircraft.

In literature there are various conventions for defining the angles that form *pitch-roll-yaw* and the one we used is the following: the rotation adopting *pitch-roll-yaw* angles can be obtained by first rotating the reference frame by the angle ρ about axis X (*pitch*), then rotating it by the angle ϕ about axis Y (*roll*), and finally by the angle θ about axis Z (*yaw*). The resulting frame orientation is obtained by composition of rotations with respect to the reference frame, and then it can be computed via pre-multiplication of the matrices of elementary rotation:

$$\mathbf{R}(\Phi) = \mathbf{R}_z(\theta)\mathbf{R}_y(\phi)\mathbf{R}_x(\rho) \tag{3.10}$$

$$= \begin{bmatrix} c_\theta c_\phi & -s_\theta c_\rho + c_\theta s_\phi s_\rho & s_\theta s_\rho + c_\theta s_\phi c_\rho \\ s_\theta c_\phi & c_\theta c_\rho + s_\theta s_\phi s_\rho & -c_\theta s_\rho + s_\theta s_\phi c_\rho \\ -s_\phi & c_\phi s_\rho & c_\phi c_\rho \end{bmatrix} \tag{3.11}$$

where c and s are shorter notations to denote *cosine* and *sine* respectively.

Conversely, the transformation from matrix $\mathbf{R}$ to the Φ is described in the following formulas. We define

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \tag{3.12}$$

and the solution for $\phi \neq \pm\frac{\pi}{2}$ is given by

$$\rho = \text{atan2}(r_{32}, r_{33}) \tag{3.13}$$

$$\phi = \arcsin(-r_{31}) \tag{3.14}$$

$$\theta = \text{atan2}(r_{21}, r_{11}) \tag{3.15}$$

otherwise if $c_\phi = 0$ it is possible to determine only the quantity [1] $\rho \pm \theta$.

3.3.3 Unit quaternion

The quaternion is an hyper-complex entity that represents the extension of a complex number in a higher dimensional space

$$\mathbf{q} = \eta + i\epsilon_i + j\epsilon_j + k\epsilon_k = \eta + \boldsymbol{\epsilon} = \begin{bmatrix} \eta \\ \boldsymbol{\epsilon} \end{bmatrix} \tag{3.16}$$

where η is the real part, $\boldsymbol{\epsilon}$ is complex part and (i, j, k) is a right-handed coordinates frame that respect Hamilton's rules if we are using the Hamilton's convention

$$i^2 = j^2 = k^2 = ijk = -1 \tag{3.17}$$

$$ij = k \qquad jk = i \qquad ki = j \tag{3.18}$$

$$ji = -k \qquad kj = -i \qquad ik = -j \tag{3.19}$$

The other convention used for quaternions is the one derived from *NASA's Jet Propulsion Laboratory (JPL)*, where the complex part (i, j, k) is a left-handed coordinates frame

$$i^2 = j^2 = k^2 = ijk = -1 \tag{3.20}$$

$$ij = -k \qquad jk = -i \qquad ki = -j \tag{3.21}$$

$$ji = k \qquad kj = i \qquad ik = j \tag{3.22}$$

We can provide a physical interpretation of the quaternion if we consider a rotation (given by η) around a certain axis (given by $\boldsymbol{\epsilon}$) and we assign

$$\boldsymbol{q} = \begin{bmatrix} \eta \\ \boldsymbol{\epsilon} \end{bmatrix} = \begin{bmatrix} \cos(\frac{\theta}{2}) \\ \boldsymbol{e}\sin(\frac{\theta}{2}) \end{bmatrix} \tag{3.23}$$

with $\boldsymbol{e}$ identifying a unit vector: $\boldsymbol{q}$ is the rotation of θ around $\boldsymbol{e}$ and the quaternion becomes a *unit quaternion*. Indeed, it follows the norm computation:

$$||\boldsymbol{q}||^2 = \cos^2(\frac{\theta}{2}) + \sin^2(\frac{\theta}{2}) = \eta^2 + \boldsymbol{\epsilon}^T\boldsymbol{\epsilon} = 1 \tag{3.24}$$

[1] When this condition occurs, we lose one degree of freedom. This phenomenon is known as *Gimbal Lock*

Before describing the relation between the unit quaternions and the rotation matrices, we introduce the **Rodrigues' rotation formula**, which gives an efficient method for computing the rotation matrix $\mathbf{R}$ corresponding to a rotation by an angle θ about a fixed axis specified by the unit vector $\omega \in \mathbb{R}^3$

$$\mathbf{R}(\theta, \omega) = \mathbf{I}_3 + \frac{\sin\theta}{\theta}[\boldsymbol{\omega}]_x + \frac{1-\cos\theta}{\theta^2}[\boldsymbol{\omega}]_x^2 \tag{3.25}$$

where $[\omega]_x$ denotes the antisymmetric matrix. Through the Rodrigues' formula and trigonometric formulas, the unit quaternion can be put in relation with the rotation matrix as follows

$$\mathbf{R}(\boldsymbol{q}) = \mathbf{I}_3 + 2\eta[\boldsymbol{\epsilon}]_x + 2[\boldsymbol{\epsilon}]_x^2 \tag{3.26}$$

$$= \begin{bmatrix} \eta^2+\epsilon_i^2-\epsilon_j^2-\epsilon_k^2 & 2(\epsilon_i\epsilon_j-\eta\epsilon_k) & 2(\epsilon_i\epsilon_k+\eta\epsilon_j) \\ 2(\epsilon_i\epsilon_j+\eta\epsilon_k) & \eta^2-\epsilon_i^2+\epsilon_j^2-\epsilon_k^2 & 2(\epsilon_j\epsilon_k-\eta\epsilon_i) \\ 2(\epsilon_i\epsilon_k-\eta\epsilon_j) & 2(\epsilon_j\epsilon_k+\eta\epsilon_i) & \eta^2-\epsilon_i^2-\epsilon_j^2+\epsilon_k^2 \end{bmatrix} \tag{3.27}$$

where we use the fact that $\theta = ||\boldsymbol{\omega}||$ and Equation (3.23). From (3.27) it is possible to note the fundamental propriety of quaternions called *double coverage*, i.e the rotation $\mathbf{R}(\boldsymbol{q})$ can be represented by both $\boldsymbol{q}$ and $-\boldsymbol{q}$. On the other hand, the relationship between quaternion and *Roll-Pith-Yaw* angles is described by means of

Finally, it is worth to note the relationship between quaternions and *Roll-Pith-Yaw* angles is described by

$$\rho = \text{atan2}(2\epsilon_j\epsilon_k + 2\eta\epsilon_i, \eta^2 - \epsilon_i^2 - \epsilon_j^2 + \epsilon_k^2) \tag{3.28}$$

$$\phi = \arcsin(2\eta\epsilon_j - 2\epsilon_i\epsilon_k) \tag{3.29}$$

$$\theta = \text{atan2}(2\epsilon_i\epsilon_j + 2\eta\epsilon_k, \eta^2 + \epsilon_i^2 - \epsilon_j^2 - \epsilon_k^2) \tag{3.30}$$

3.3.4 Kalman filter and complementary filter

Kalman filter is an algorithm that exploits a series of continuous measurements, including statistical noise, and returns estimates of unknown variables that are more accurate than those based on a single measurement alone, by estimating a joint probability distribution over the variables for each instant [28]. Kalman filtering is based on a dynamic model, known inputs to that system and continuous measurements to estimate the state of the system itself. This algorithm is

typically employed to perform sensor fusion and in our case, it allows us to merge the measurements made by accelerometers and gyroscopes. In fact, these two sensors provide very uncertain measurements: the former are unable to assess whether the reference system is moving, while the latter, although accurate, suffer from measurement drift, leading to results that are totally different from what is expected. However, by combining the measurements made by these individual sensor types, we can accurately estimate the position and orientation of an object by knowing its initial state.

Without going into excessive detail, consider the discrete time Kalman filter

$$\begin{aligned} \boldsymbol{x}(k) &= A\boldsymbol{x}(k-1) + B\boldsymbol{u}(k) + \boldsymbol{v}_x(k) \\ \boldsymbol{y}(k) &= C\boldsymbol{x}(k) + \boldsymbol{v}_y(k) \end{aligned} \tag{3.31}$$

where k is the discrete time, $\boldsymbol{x} \in \mathbb{R}^n$ is the vector state, $\boldsymbol{y} \in \mathbb{R}^m$ is the vector of measurements, $\boldsymbol{u} \in \mathbb{R}^d$ is the input vector, $A \in \mathbb{R}^{n \times n}$ is the state transition matrix, $B \in \mathbb{R}^{n \times d}$ and $\boldsymbol{v}_x \in \mathbb{R}^n$ and $\boldsymbol{v}_y \in \mathbb{R}^m$ are, respectively, the noise vector of the state and of the measurements, which are Gaussian distributed, uncorrelated, with zero mean and variance

$$\mathbb{E}\begin{bmatrix} \boldsymbol{v}_x(k) \\ \boldsymbol{v}_y(k) \end{bmatrix} = \begin{bmatrix} \mathbb{E}[\boldsymbol{v}_x(k)\boldsymbol{v}_x(k)^T] & \mathbb{E}[\boldsymbol{v}_x(k)\boldsymbol{v}_y(k)^T] \\ \mathbb{E}[\boldsymbol{v}_x(k)\boldsymbol{v}_y(k)^T] & \mathbb{E}[\boldsymbol{v}_y(k)\boldsymbol{v}_y(k)^T] \end{bmatrix} = \begin{bmatrix} Q(k) & 0 \\ 0 & R(k) \end{bmatrix} \tag{3.32}$$

At any instant k, the Kalman filter computes an *estimation stage* and a *prediction stage* :

- The *estimation stage* estimates the current state based on the previous states

$$\hat{\boldsymbol{x}}(k|k-1) = A\hat{\boldsymbol{x}}(k-1|k-1) + B\boldsymbol{u}(k) \tag{3.33}$$

and the previous *error covariance matrix*

$$P(k|k-1) = AP(k-1|k-1)A^T + Q(k) \tag{3.34}$$

In this stage we also compute *innovation covariance*, given by

$$\Lambda(k) = CP(k|k-1)C^T + R(k)$$

and the *Kalman gain* as

$$L(k) = P(k|k-1)C^T\Lambda(k)^{-1}$$

The *innovation* associated to the measurement is defined as

$$\boldsymbol{e}(k) = \boldsymbol{y}(k) - C\hat{\boldsymbol{x}}(k|k-1)$$

- The *prediction stage*, using results from the estimation stage, evaluates

$$\hat{\boldsymbol{x}}(k|k) = \hat{\boldsymbol{x}}(k|k-1) + L(k)\boldsymbol{e}(k)$$

and the a *posteriori error covariance matrix*

$$P(k|k) = (I - L(k)C)P(k|k-1)$$

where I denotes the identity matrix.

A key element for the use of a Kalman filter is the definition of the initial conditions ($\boldsymbol{x}(0)$ and $P(0)$) and the definition of the covariance matrices $R(k)$ and $Q(k)$.

Complementary filter is an algorithm that allow us to merge data from accelerometers and gyroscopes without performing a statiscal analyze of the signals received and its working is much more intuitive than a Kalman filter. This tool allows us to calculate the orientation of the sensor based on measurements made by both devices simultaneously. In its most simple form, the filter looks as follows

$$\angle = \alpha \cdot (\angle + \boldsymbol{m}_{gyro} \cdot dt) + (1 - \alpha) \cdot \boldsymbol{m}_{acc} \tag{3.35}$$

where $\angle$ denotes the angle we would like to measure, $\alpha \in [0, 1]$ is the filter parameter, $\boldsymbol{m}_{gyro}$ and $\boldsymbol{m}_{acc}$ are, respectively, the measure of the gyroscope and the accelerometer and dt is the interval between two consecutive measurement. In this filter, the gyroscope data is integrated every timestep with the current angle value and after this it is combined with data from the accelerometer. However, this filter, although easier to implement than the previous one, has considerable limitations, including the choice of the alpha parameter, which must be as large as the gyroscope used is reliable.

Chapter 4

First system design: an analogic tool

In this section we begin the experimental study trying to reconstruct the dynamics of the knee. Initially, we focus our attention on finding a method capable of evaluating the angles formed by the knee when it is in a static position. To achieve it, we carry out experiments using triaxial accelerometers, without using additional sensors. This choice inevitably involv compromises, including the impossibility of studying a dynamic system. In the following, the above-mentioned aspects are analysed in depth and the algorithms used are explained in detail.

4.1 Purpose of the system design

After having presented the fundamental requirements that our system must meet and having reviewed the current literature on the reconstruction of the dynamics of a knee, it is possible to move on to the design of the proper system. Exploiting the theory presented in the works of *Bakhshi et al.* in [6] and of *McGinnis et al.* in [22], we decide to carry out a first prototype using analog accelerometers, to estimate the angles of interest formed by a knee under static conditions. This step is mainly done in order to investigate the possible criticalities of a similar system. In addition, tests are performed to assess where it is most convenient to place the sensors and to determine whether it is sufficient to use two sensors or it is necessary to use four or more.

The decision to use only analog accelerometers also means accepting a compromise, namely the impossibility of perfectly reconstructing the flexion angle of a moving knee. In fact these sensors have intrinsic problems that can be solved

only by combining data from them with data from gyroscopes and magnetometers. For example, a triaxial accelerometer always measures the acceleration of gravity along one of its axes, but it is not always possible to determine which axis is perpendicular to the ground, even if the initial positioning of the sensor is known. If the accelerometer is moving, the force exerted on an axis different from that perpendicular to the ground measures the greatest magnitude, making it difficult to understand the orientation of the sensor itself. Therefore, following the approach presented above, i.e. dividing the main problem into sub-problems that are easier to solve and being aware of these factors, we decide to use triaxial accelerometers only to evaluate the range of motion of a knee.

It is necessary to make one clarification, in order to allow a better understanding of what is presented below: at this stage we decide to avoid to use a kneepad as a support for the sensors, since this instrument is a simple support tool, which can be assumed to be irrelevant to the values we have to calculate.

We have already introduced the concept of range of motion when we talked about the practical feasibility of the active safety system (Section 2.4) and the state of the art (Section 3.2), but we have not yet given a correct definition of this concept. *Range of motion* or *ROM* is a measure of the amount of movement around a specific joint or body part. There are three different ways of looking a knee range of motion:

- **Active Knee ROM**: this is how much the knee can bend and straighten on its own, i.e. knee muscles actively contracting without any external help;
- **Passive Knee ROM**: how far the knee can bend and straighten when moved by an external force, usually another person. In this case the leg and the knee muscles are completely relaxed;
- **Active Assisted ROM**: how far the knee can move when it is weak or in pain with some assistance. Knee muscles are working to move the leg but with some help, such as a physical therapist.

Usually the passive ROM shows greater values with respect to the active ROM. The typical values for the passive ROM of a knee are shown in the Table 2.1.

4.2 Instrumental setup

Our initial idea is to replicate the measurement of range of motion using only accelerometers: one (or two) is placed anywhere on the thigh above the knee,

while another sensor (or two) is placed anywhere on the shin. This first phase, although it might not seem useful for our purpose of reconstructing the dynamics of a knee, allows us to familiarise ourselves with the sensors and, at the same time, to make an initial assessment of the number of accelerometers we might need, the sampling frequency we may use, and finally the type of results we could obtain.

The instrumental setup we use to calculate the range of motion of a knee consists of:

- Two or four analog triaxial accelerometers *ADXL335*, from Analaog Devices (Figure 4.1);
- A Data Acquisition Device (DAQ) of National Instruments, USB-6211 (Figure 4.3);
- A Personal Computer with MATLAB;
- Various cables for connection.

In the following, firstly we characterise the instruments used in the experiments and then we describe the algorithms that allowed us to calibrate the system.

4.2.1 ADXL335

The ADXL335 is an analog triaxial accelerometer and it is optimal for our purpose as it is very small, light, requires little power and measures accelerations with a full scale of $\pm 3\ g$. Another fundamental aspect is that its trans-characteristics is strictly linear, which allows us to have a direct correspondence between the measured quantity and the voltage output produced by the sensor. To function properly, it requires a power supply of around $3.3\ V$, or in any case between $1.8\ V$ and $3.6\ V$. An important feature of this sensor is that the variation of bias and sensitivity along the three axes remain almost constant for temperatures between $-30°C$ and $+90°C$ (Figure 4.2), which could possibly allow us to use it without the data being distorted even in environments with a temperature below $0°C$, such as a ski slope.

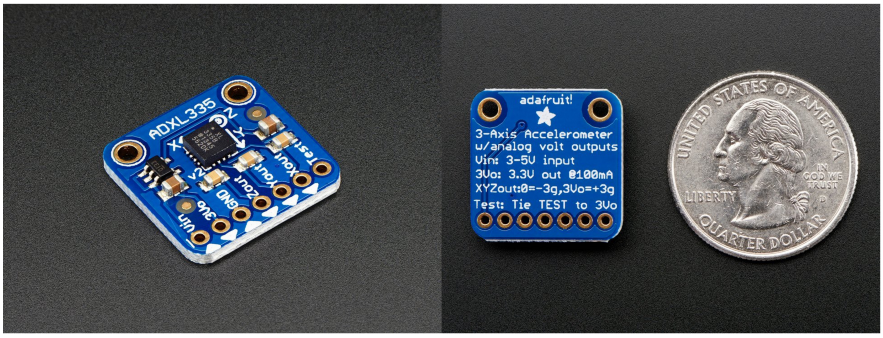

Figure 4.1: Front and back view of an analog triaxial accelerometers ADXL335 - Adafruit

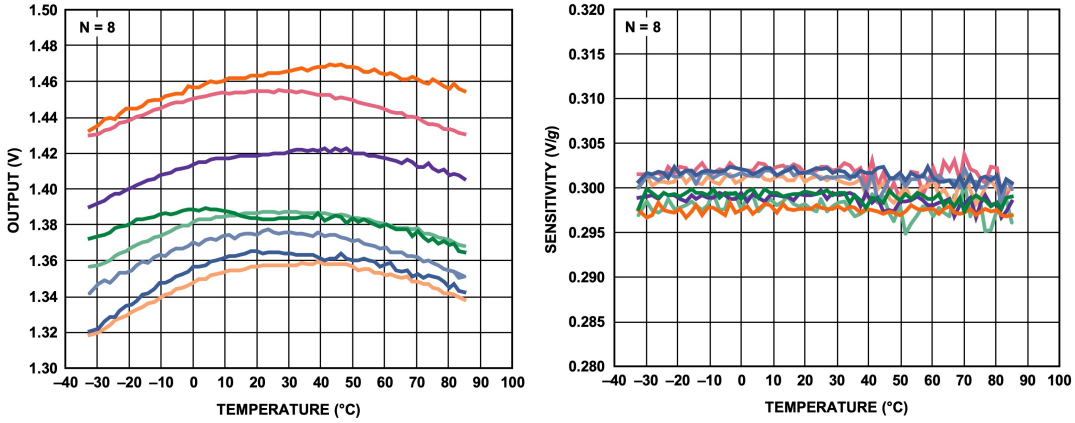

Figure 4.2: Image from [4] - Left: Z-Axis Zero g Bias vs. Temperature; Right: Z-Axis Sensitivity vs. Temperature

4.2.2 NI DAQ USB 6211

The NI DAQ USB-6211 is a multifunction I/O device and consists in 16 analog input and 2 analog output, both with a maximum sampling frequency of 250 kS/s, 4 digital input and 4 digital output. The analog inputs could be used as 8 differential channels, or as 16 single ended, with selectable input range of $\pm 0.2\ V, \pm 1\ V, \pm 5\ V, \pm 10\ V$. The crosstalk at 100 kHz between adjacent channels is of about $-75\ dB$ and the input FIFO size is of 4095 samples. The operating temperature is recommended to be between $0°C$ and $45°C$, with a maximum altitude of 2000 m, and indoor use is recommended. Because of these latter characteristics, its use is limited to a laboratory only, so it cannot be used outside a controlled environment. However, since its purpose is to create an interface between the sensors and the computer, it can easily fulfil its purpose of reconstructing the ROM of a knee, but it cannot be used to reconstruct the dynamics of a knee in a hostile environment such as a ski slope.

Figure 4.3: NI DAQ USB-6211

4.2.3 Accelerometer calibration

Since we use the accelerometers of an analogue type, the sensors need an initial calibration, as the data they return may be affected by offsets and sensitivities different from those reported on the datasheet. Therefore, we need to develop a calibration procedure which must be carried out whenever a new sensor is connected to the NI DAQ.

First of all, the sensors we want to calibrate are connected via copper wires to the DAQ, which is connected to the PC via a suitable USB cable; each sensor output is assigned to a different DAQ input. To avoid cross talk phenomena between adjacent channels, we decided to use non-adjacent analogue inputs when calibrating two sensors simultaneously; this same approach could not be applied when it is necessary to calibrate four sensors together, since we need to use 12 channels simultaneously out of the 16 available. The DAQ is then set to a sampling rate of 20 $\frac{kS}{s}$ for an interval of 5 s for each scan interval. Every input channel used is configured as *single ended*, with an acceptable range of ± 5 V. Finally, to calculate the correct offset and sensitivity values for each sensor, we proceed as presented in Algorithm 1.

The data resulting from the calibration procedure are then stored in an external file, so that they could be retrieved each time a measurement is performed by the accelerometer.

These sensors, since they are of an analogue type, return a voltage value in correspondence of a given acceleration. In order to allow a user to understand the

Algorithm 1 Accelerometer calibration procedure

for $\forall$ *accelerometer* **do**
  for *iteration* $= 1, \ldots, 6$ **do**
    Pose the accelerometer with max value on any axis
    Take values for 5 seconds
    Compute max and min values of axis perpendicular to the ground
    Compute mean of other axes
  end for
  Compute offset: sum each mean for the same axis and divide by 4
  Compute sensitivity: sum max and min for the same axis and divide by 2
  Save offset and sensitivity
end for

meaning of the value produced by the accelerometer, every time it is necessary to perform a "translation" operation of the acquired data, using the process shown in the Listing 4.1. This MATLAB function enables the value of the acceleration measured by the sensors to be calculated using the voltage value produced by the accelerometer during the measurement and the values calculated during the calibration process.

```
function cal_data = CALIBRATION_ACC(raw_data, offset, ...
    sensitivity)
cal_data = zeros(size(raw_data));
    for i = 1:size(raw_data,1)
        cal_data(i,1) = (raw_data(i,1) - ...
            offset(1,1))/sensitivity(1,1);      %X axis
        cal_data(i,2) = (raw_data(i,2) - ...
            offset(2,1))/sensitivity(2,1);      %Y axis
        cal_data(i,3) = (raw_data(i,3) - ...
            offset(3,1))/sensitivity(3,1);      %Z axis
    end
end
```

Listing 4.1: MATLAB function for calibrating measurements from accelerometers

For simplicity, in the following we will consider all measurements made with analog accelerometers as calibrated measurements.

4.3 Rotation angle measurement with accelerometer

Every signal coming from the accelerometers represents an acceleration onto the X, Y and Z axes, but these information need to be processed in order to understand how a knee is bent. Using the accelerations measured by the sensors, it is possible to calculate the *pitch, roll* and *yaw* angles of the sensor itself and then we can use this information to calculate the adduction/abduction[1], the flexion/extension and the rotation angle of the shin relative to the thigh. Referring to Figure 4.4, consider an accelerometer whose Z axis points up, while X and Y axes directed such that the reference frame is right-handed.

To compute the pitch angle, it is necessary to consider the acceleration measured in each axis. Using a geometric approach, it is possible to calculate the angle between the X axis and the ground by calculating the *arcotangent* of the acceleration on X divided by the square root of the sum of the accelerations on the other two axes squared (4.1).

In the same way it is possible to define the roll angle as the angle formed between the Y axis and the ground. Again, it is calculated as the pitch angle, but now the accelerations on the X axis and Y axis are inverted (4.2).

The yaw angle is defined differently from the previous two: θ is the angle between the Z axis and the vertical axis along which the force of gravity acts, i.e. the axis perpendicular to the ground plane. The yaw angle is calculated in a similar way as the angles of roll and pitch, however it is evident that it is necessary to invert the relationship between the accelerations calculated on the X and Y axes and the acceleration calculated on the Z axis (4.3).

$$\rho = \arctan\left(\frac{A_X}{\sqrt{A_Y^2 + A_Z^2}}\right) \tag{4.1}$$

$$\phi = \arctan\left(\frac{A_Y}{\sqrt{A_X^2 + A_Z^2}}\right) \tag{4.2}$$

$$\theta = \arctan\left(\frac{\sqrt{A_X^2 + A_Y^2}}{A_Z}\right) \tag{4.3}$$

[1]To be precise, in theory adduction for a knee is not a practicable movement; however, during passive mobilisation it is possible to deviate the shin with respect to the vertical axis ideally passing through the leg, forming an angle, which is considered as the angle of adduction/abduction of the knee. The range of this angle is significantly lower than the range of flexion and rotation.

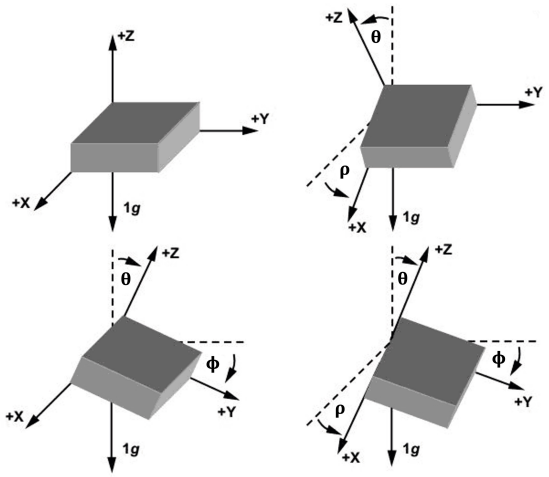

Figure 4.4: Graphical representation of pitch ρ (top-right, i.e. angle between X-axis and ground), roll ϕ (top-left, i.e. angle between Y-axis and ground), yaw θ (top-bottom, i.e. angle between Z-axis and vertical axis)

The angles described above are of fundamental importance for calculating the ROM of a knee. However, the fact that they are calculated as the inverse of a goniometric function means there are approximations that can not be neglected. It must be pointed out there are various techniques that allow the calculation of the pitch, roll and yaw angles, but the method just described is the most intuitive, since it uses geometric relations, and is the most computationally efficient, despite the presence of inverse goniometric functions.

4.3.1 Knee range of motion evaluation

After having presented the method used to calculate pitch, roll and yaw angles using an accelerometer, we have to show that in theory it is possible to measure the adduction, flexion and rotation of a knee using two or more sensors. First of all, it is necessary to define where we would like to place the sensors: one accelerometer should be placed above the knee, at an indefinite point of the thigh; the second accelerometer could be positioned on the tibia, at an undetermined point of it. Then, synchronous acquisition of both devices could begin.

Before explaining the procedure that has been devised to calculate the angles of interest in the knee, it is important to expose the two assumptions we have made. The first one is that we assumed that a knee's movements during this evaluation are limited to one plane at the same time. This means that if we are estimating the knee flexion angle, this can be calculated by computing the difference between the roll angles of the different sensors, or by considering the

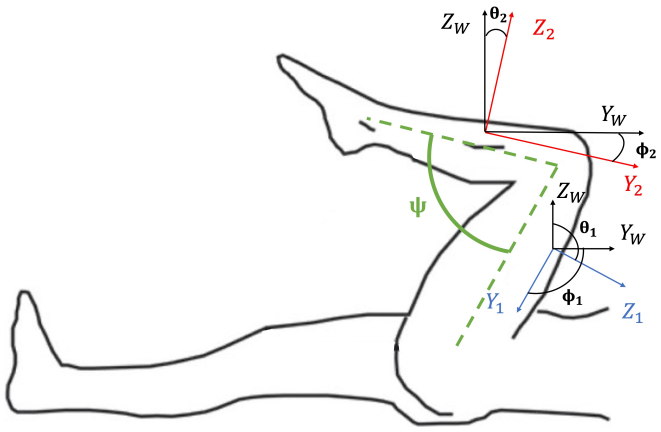

Figure 4.5: Graphical representation of the angles formed by accelerometer 1 and accelerometer 2 with respect the world frame - by considering the ground flat - during the flexion. The sensors are placed with the Y-axis parallel to the knee bones and pointing the pelvis: accelerometer 1 has the Y-axis parallel to the femur, accelerometer 2 has the Y-axis parallel to the tibia

difference between the yaw angles of the two sensors. This hypothesis, although it may seem excessive, is supported by the fact that the assessment of the passive range of motion is carried out by a subject external to the person on whom this valuation is conducted and therefore, following a precise protocol this scenario can always be considered feasible. The second assumption we made is about the orientation of the sensors: in fact, we have assumed that, unless a reasonable error, the axes of the two sensors are aligned with each other. This hypothesis can always be considered valid, since in our final idea the sensors used must be mounted on a support, such as a kneepad, capable of keeping the orientation of the accelerometers fixed.

Once these premises have been made, for a better understanding of how the angles defining the ROM of a knee are calculated, consider the case where flexion is measured. Using the procedure presented in Section 4.3, it is possible to calculate the roll and yaw angles of the sensor on the thigh (ϕ_1, θ_1) and similarly for the sensor on the shin (ϕ_2, θ_2) (Figure 4.5). The flexion angle Ψ is simply calculated by subtracting the values measured with the accelerometer on the shin with the values calculated with the sensor on the thigh

$$\Psi_{roll} = \phi_2 - \phi_1 \tag{4.4}$$

$$\Psi_{yaw} = \theta_2 - \theta_1 \tag{4.5}$$

Note that, according to the first hypothesis we have assumed, the two angles

should be equal, namely $\Psi_{roll} = \Psi_{yaw}$, since the flexion should only occur in the sagittal plane.

The calculation of knee adduction and vertical rotation are performed in a similar fashion: in the first case the pitch and roll angles are used, while in the second case the pitch and yaw angles are employed.

In Algorithm 2 is shown the pseudocode used to calculate knee flexion. The full MATLAB code can be found in Appendix B.

Algorithm 2 Knee flexion angle evaluation

```
for iteration = 1, 2, ... do
    Acquire data from thigh sensor
    Acquire data from shin sensor
    for iteration = K, 2K, 3K, ... ← K = 10 do
        Compute roll and yaw angles of thigh sensor φ1, θ1
        Compute roll and yaw angles of shin sensor φ2, θ2
        Compute knee flexion Ψ = φ2 − φ1 = θ2 − θ1
    end for
end for
```

4.4 Algorithm validation

After the explanation of how to calculate the angles that allow the evaluation of the ROM of a knee, it is necessary to carry out a practical validation of what has been shown to be successful in theory. To do this, as mentioned above, no support, such as a knee brace, is used, because the sensors are simply applied directly to the leg of a patient. The first experiments employed two accelerometers, positioned according to the following procedure (Figure 4.6):

- The participant is lying on his back, with his body facing upwards and his legs fully extended;
- A first sensor is placed approximately 7 cm above the patella, with the Y axis pointing the pelvis and the Z axis directed upwards;
- A second sensor is placed on the shin, at a distance of approximately 7 cm from the patella, again with the Y axis facing the pelvis and the Z axis directed upwards.

Both sensors, previously calibrated, are connected to the same DAQ, supplied with an input voltage and connected to the same ground, so that they have the

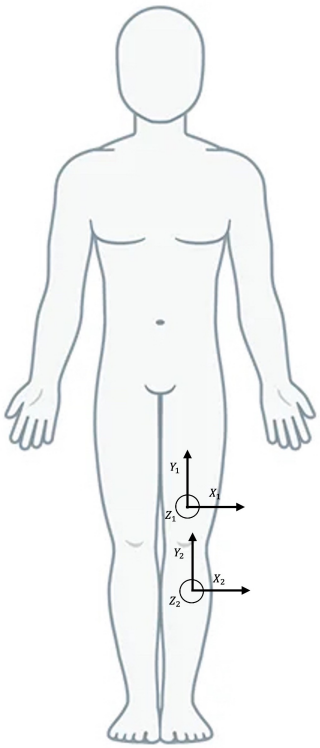

Figure 4.6: Graphical representation of the accelerometer 1 positioned on the thigh and the accelerometer 2 posed on the shin

same voltage reference. The DAQ is configured with a sampling rate of 2 $\frac{kS}{s}$ and each input channel used is set up in single ended mode. Next, a program is written on MATLAB which performs the configuration just described and runs a continuous acquisition. For every 10 samples generated by the accelerometers, a *listener* calculates the pitch, roll and yaw angles of each sensor.

Since the instrument we are designing has intrinsic limitations due to the method we used to calculate the pitch, roll and yaw angles, we have to create a procedure to acquire the data. In the following, we explain the procedure used to measure the angles of flexion, rotation and adduction currently used in the medical field, and then we illustrate the procedures we are developing.

4.4.1 Knee flexion/extension evaluation

Standard procedure To measure the flexion angle of a knee, an anatomical goniometer and a standard procedure are typically used. In order to carry out the passive ROM, the intervention of an external operator is necessary. The person on which the measurement should be taken is placed supine with his legs extended and, once the typical knee reference point is identified, which corresponds to

the lateral epicondyle of femur, i.e. the central outer point of the knee, the goniometer is placed over it. The fixed arm of the protractor is aligned with the greater trochanter (that is the upper bone protrusion of the femur) along the outside of the thigh and then the other arm of the goniometer is aligned with the lateral malleolus of the ankle. To measure knee extension, it is gently push towards the floor (using the leg muscles, not the hands) so that the knee is as straight as possible and it is taken the maximum angle reached. To measure knee flexion, it is bend as far as possible, sliding the foot towards the buttocks, keeping the arms and the axis of the protractor in position.

Adopted procedure The procedure we use is very similar to the one normally used. The patient is placed supine, on a flat surface parallel to the ground. To measure hyperextension, the knee should be gently pushed down while the other hand delicately lifts the calf. To measure flexion, the knee is raised, keeping the gluteus in contact with the surface on which the patient is lying and pressure is applied towards the patient's trunk, so that the knee flexes as much as possible.

Initially, the idea was to make the patient sit down and keep his leg elevated above the ground so that the person performing the passive mobilisation could make the necessary movements with the leg. However, we discovered, through experimental practice, that this sequence is not applicable since maximum leg flexion cannot be achieved and this technique was discarded.

4.4.2 Knee rotation evaluation

Standard procedure In order to assess the rotation of the knee, the person whose knee is to be measured is made to sit on a flat surface, with their legs raised off the floor. The knee to be measured is bent at 90°, completely rotated inwards and the centre of the protractor is placed in the middle of the malleolus, on the sole of the foot. Keeping the fixed arm of the protractor aligned with the sole of the foot, a torsion is made at the height of the tibia, rotating the mobile arm of the protractor by an angle equal to the rotation undergone by the knee. When making this measurement, it is essential not to rotate the ankle, otherwise the results are invalid.

Adopted procedure The patient is placed supine on a rigid surface parallel to the ground. Keeping the ankle rigid and the gluteus of the leg on which the measurement is made in contact with the support surface, the knee is rotated internally and externally. Care must be taken to prevent the ankle and hip from

rotating, which could invalidate the measurement.

4.4.3 Knee adduction/abduction evaluation

Standard procedure To calculate the angle of valgus/varus of the knee, it is necessary that the person on which the measurement is to be made is lying supine, with the legs extended. The centre of the goniometer is placed in the centre of the patella, with the fixed axis facing the head of the femur, while the mobile axis is directed towards the axis of the tibia. Press the shin gently inwards and outwards, taking care not to move the hip to avoid distorting the measurements.

Adopted procedure This type of measurement is completely different to the standard procedure. It does not actually measure the adduction/abduction performed by the knee, which are movements that not all physiotherapy manuals suggest to calculate. The procedure we developed makes it possible to calculate the valgus/varus angle by simply standing the patient upright and measuring how far the axis of the shin is inclined with respect to the axis of the thigh.

4.4.4 Range of motion evaluation with 4 accelerometers

As previously mentioned, tests are conducted using four accelerometers. The procedures for calculating the data of interest are exactly the same as for two accelerometers. The main difference between the experiments performed with a different number of sensors concerns the position where the sensors are applied and the computation of the angles of interest. Moreover, tests are conducted by placing the sensors in different configurations to evaluate which is the best. Note that the same assumptions have been made here as previously.

In the first experiment, the sensors are placed in pairs above and below the knee, laterally with respect to it (Figure 4.7 a). Two sensors are used to calculate the movements made by the thigh, while the other two are used to measure the movements made by the shin. For simplicity, consider the case only the sensors positioned on the thigh. Each accelerometer estimates the pitch, roll and yaw angles $(\rho_1, \phi_1, \theta_1)$ and $(\rho_2, \phi_2, \theta_2)$. In order to be able to use the data thus obtained, a simple average is made, without giving greater importance to one or to the other sensor, namely

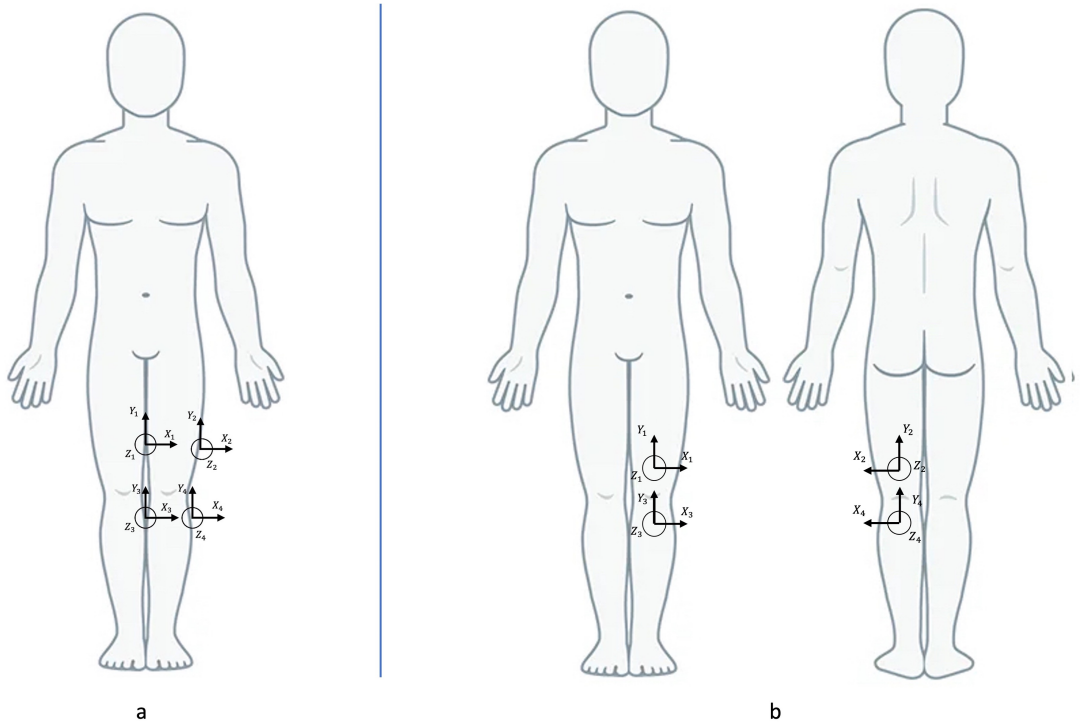

Figure 4.7: Graphical representation the accelerometers 1 and 2 positioned on the thigh and the accelerometer 3 and 4 posed on the shin in different configurations

$$\begin{aligned} \bar{\rho}_{12} &= \frac{\rho_1 + \rho_2}{2} \\ \bar{\phi}_{12} &= \frac{\phi_1 + \phi_2}{2} \\ \bar{\theta}_{12} &= \frac{\theta_1 + \theta_2}{2} \end{aligned} \tag{4.6}$$

The same procedure is followed with the sensors positioned on the shin, finding $(\bar{\rho}_{34}, \bar{\phi}_{34}, \bar{\theta}_{34})$. Finally, as explained in Section 4.3.1, flexion, rotation and adduction are calculated by simply subtracting the pitch, roll and yaw values computed on the shin with those evaluated on the thigh. Also manual measurements are made using an anatomical goniometer

The second experiment is executed by placing the sensors on both the front of the leg and the back (Figure 4.7 b). The procedure followed to calculate the angles of interest is exactly the same as that explained above.

4.5 Discussion of results

In this section, we analyse the outcomes obtained through the experimental tests described in the previous section and investigate the causes that led to these results. We decide to analyse the data obtained using two accelerometers separately from the data obtained using four sensors. To evaluate the goodness of the results, we use the **Absolute Error** (**AE**) and the **Root Mean Square Error** (**RMSE**), calculated as follows

$$\begin{aligned} \mathbf{AE} &= |x - \bar{x}| \\ \mathbf{RMSE} &= \sqrt{\frac{\sum_{i=1}^{N}(x - x_i)^2}{N}} \end{aligned} \tag{4.7}$$

where x is the expected value, i.e. the value calculated manually with the goniometer, $\bar{x}$ is the mean of the measurements, x_i is the $i - th$ measure and $i = 1, \ldots, N$, with N indicating the number of observations.

4.5.1 Two accelerometers

Measurements taken using two accelerometers are shown in Table 4.1. Remember that we initially made two assumptions: the first assumed that the single movements made by a knee took place in a single plane, while the second concerned the positioning of the sensors. Due to the first hypothesis, in the table of results we have reported only one of the two angles calculated every time a movement was measured, since the results should be similar using one parameter or the other.

What is immediately evident from the table is that the results obtained are considerably different from what we expected. Considering only flexion, the absolute average error of the six measurements performed was 11.27°, which compared to the results found in the literature is the highest ever. In particular, we can see that the absolute error increases as the angle to be measured increases. In Figure 4.8 we can observe the behaviour just described. Due to the small number of measurements taken, we used an error interpolation method to at least get an idea of the error trend. The interpolation was done by using either a quadratic function or a cubic function, as shown in the same figure, to predict the evolution of the error as the angle to be calculated increases. We observe that the cubic function allows a virtually perfect interpolation, having the index

of determination[2] $R^2 = 0.99$, while the quadratic function does not perfectly describe the trend of the data, having $R^2 = 0.92$. However, by choosing a function for interpolating data that has an index of determination too close to 1 we risk falling into the problem of overfitting, namely the problem that the choice of interpolating function is too precise, making it work correctly for our problem, but failing to predict future data correctly. Therefore, if we want to interpolate the data correctly, we should use a quadratic function. This means that the absolute error between the true value and the measured angle is approximately quadratic, which is obviously not acceptable according to our specifications. The same behaviour can be observed with reference to RMSE. This phenomenon is due to the fact that the pitch, roll and yaw angles are calculated as inverse trigonometric functions, which have important limits as they are nonlinear functions. An error of 43° is observed compared to a real angle of 120°, which is 35% of the value to be measured. This means that the measuring system we have developed is not at all reliable.

On the other hand, with regard to the angles of rotation and abduction, we found the measurements to be quite reliable, since we have average errors of, respectively, 2.86° and 0.56°. This is probably due to the fact that the angles measured during the estimation of rotation and adduction are very small angles compared to those measured during the calculation of flexion.

Figure 4.9 shows the measurement of the flexion angle carried out using the roll and yaw angles when the knee is at -5° and +120°. From these plots it is possible to observe a very important fact, namely that our first hypothesis is not valid. In fact, this data are obtained by perfectly following the procedure we have devised, but despite this we can see very clearly that the flexion angles calculated using roll are significantly different from the same angles calculated using yaw. This is probably due to the fact that the technique we used to calculate the flexion angle is not the ideal solution, since the data coming from the accelerometer are analysed independently, without merging the accelerations calculated by the sensor. In addition, it is also evident in Figure 4.9 that the data we acquired are not as stable as we would like, since even though we followed the procedure by keeping the sensors in a static position, they show a behaviour that is not

[2] R^2 is calculated from the residual, i.e. $res = x_{measured} - x_{fitted}$, and it is computed as

$$R^2 = \frac{1 - SS_{res}}{SS_{tot}}$$

where SS_{res} is the sum of the squared residuals from the regression, while SS_{tot} is the total sum of squares

Movement	True Angle	Meas Angle	AE	RMSE
Extension/Flexion	$-5°$	$-8.96°$	$3.96°$	$5.03°$
ϕ	$0°$	$4.02°$	$4.02°$	$5.88°$
	$+15°$	$10.84°$	$4.14°$	$6.49°$
	$+60°$	$55.85°$	$4.15°$	$6.19°$
	$+90°$	$81.72°$	$8.27°$	$9.01°$
	$+120°$	$76.92°$	$43.08°$	$62.08°$
Int/Ext Rotation	$-10°$	$-9.39°$	$0.60°$	$1.29°$
ρ	$0°$	$2.76°$	$2.76°$	$2.77°$
	$+30°$	$26.03°$	$5.21°$	$3.97°$
Adduction/Abduction	$-10°$	$-9.49°$	$0.51°$	$1.45°$
ρ	$+10°$	$10.66°$	$0.66°$	$1.18°$

Table 4.1: Experimental evaluation of passive knee ROM calculation using 2 accelerometers, positioned as in Figure 4.6

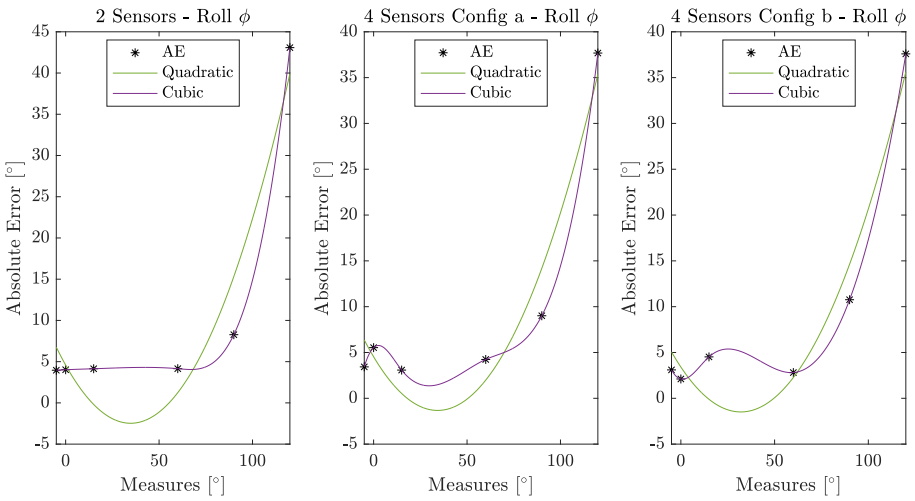

Figure 4.8: Plots of the absolute error for flexion with *roll* (ϕ) and corresponding fitting

asymptotically stable, but simply stable within a confidence interval that is too high for our standards. The same identical behaviour is found in the evaluation of knee rotation and abduction.

4.5.2 Four accelerometers

Measurements made using four accelerometers positioned as in Figure 4.7a are shown in Table 4.2, while those obtained by positioning the sensors as in Figure 4.7b are shown in Table 4.3. Analysing the results, it is clear that independently of configuration a or b, the number of sensors is irrelevant to the calculation of flexion, since even using four accelerometers we have an absolute average error of respectively $10.49°$ and $10.14°$ on the six measures performed. Although these

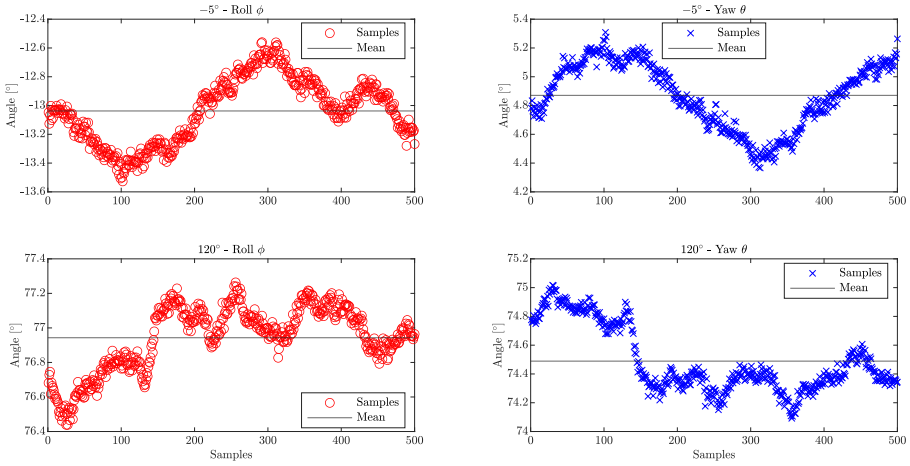

Figure 4.9: Plots of the flexion evaluation with *roll* (ϕ) and *yaw* (θ) for $-5°$ and $120°$

values are more than 1° lower than the average error calculated using two sensors, they are not small enough to consider the system valid or to justify the use of two more accelerometers. Even in this case we can see that the absolute error increases as the angle to be calculated grows (with an angle of 120° we have an error of 31.4° and 31.33°) and a similar behaviour is also noticeable in the RMSE, as we can clearly see from Figure 4.8 and the considerations that can be made about these cases are exactly the same as those mentioned in the previous paragraph. With respect to the absolute average error calculated on rotation and adduction, we have that for configuration a these values are 2.03° and 1.92°, while for configuration b these values are approximately 1.16° and 1.25°, values always consistent with the results achieved using only two sensors.

Although the data obtained are not as good as we would like, we can safely say that using four accelerometers is unnecessary and the optimal position to place the sensors on the leg is the one presented when we chose to use only two accelerometers.

4.5.3 Considerations about system designed

The acquisition system we have designed and just described has considerable limitations, in particular in the calculation of pitch, roll and yaw angles. These are computed as the inverse of the tangent, which has intrinsic limits: this function is nonlinear and the results returned are periodic by π, which means that if the patient were turned 180°, the same exact results would be obtained. This could be an advantage as errors due to the positioning of the sensors or how the

Movement	True Angle	Meas Angle	AE	RMSE
Extension/Flexion	$-5°$	$-8.43°$	$3.43°$	$6.28°$
ϕ	$0°$	$5.53°$	$5.53°$	$6.22°$
	$+15°$	$11.92°$	$3.08°$	$5.19°$
	$+60°$	$55.76°$	$4.24°$	$5.98°$
	$+90°$	$80.99°$	$9.01°$	$12.65°$
	$+120°$	$82.32°$	$37.68°$	$55.99°$
Int/Ext Rotation	$-10°$	$-9.1°$	$0.90°$	$1.37°$
ρ	$0°$	$3.42°$	$3.42°$	$6.12°$
	$+30°$	$31.77°$	$1.77°$	$6.14°$
Adduction/Abduction	$-10°$	$-9.24°$	$0.76°$	$1.99°$
ρ	$+10°$	$13.08°$	$3.08°$	$3.85°$

Table 4.2: Experimental evaluation of passive knee ROM calculation using 4 accelerometers, positioned as in Figure 4.7 a

Movement	True Angle	Meas Angle	AE	RMSE
Extension/Flexion	$-5°$	$-8.1°$	$3.1°$	$4.77°$
ϕ	$0°$	$2.11°$	$2.11°$	$7.63°$
	$+15°$	$10.48°$	$4.52°$	$8.69°$
	$+60°$	$57.19°$	$2.81°$	$3.14°$
	$+90°$	$79.24°$	$10.76°$	$16.48°$
	$+120°$	$82.41°$	$37.59°$	$49.49°$
Int/Ext Rotation	$-10°$	$-9.96°$	$0.04°$	$1.12°$
ρ	$0°$	$-1.98°$	$1.98°$	$3.76°$
	$+30°$	$28.52°$	$1.48°$	$3.32°$
Adduction/Abduction	$-10°$	$-10.96°$	$0.96°$	$2.89°$
ρ	$+10°$	$8.46°$	$1.54°$	$5.52°$

Table 4.3: Experimental evaluation of passive knee ROM calculation using 4 accelerometers, positioned as in Figure 4.7 b

patient is placed could be avoided, but it is inevitable to note that this leads to limitations in the calculation of the angles themselves, as the closer to the limit of the arcotangent function, the more incorrect the results are.

The results obtained are much worse than expected, due to several problems in addition to those already analysed. For example, one problem that has not been taken into account so far concerns the calibration of the sensors: in Section 4.2.3 we explained the procedure for calibrating the sensors and from then we always assumed that the measurements obtained were calibrated. However, that calibration procedure has some evident limitations, since in order to carry out the calibration correctly, it is necessary to perfectly position the sensor with two axes simultaneously parallel to the ground. This situation is not always feasible and if the sensor is placed with just a small inclination of a couple of degrees with respect to a flat surface, the offset and sensitivity values are not correct, compromising the calibration of all subsequent measurements. Generally, this condition is still considered acceptable, but in our case, the fact that inverse trigonometric functions are used, only accentuates the errors caused by an imperfect calibration.

Another error that has not been taken into account, but which could directly influence the measurements, may be due to a sensor supply that is not exactly $5\ V$, as the cables used during our experiments may have caused a voltage drop, thus generating incorrect voltage references.

A further problem with our measurement system, but which can easily be solved, consists in the fact that we have employed only accelerometers, which are very good sensors but return disturbed values, making the measures taken with these sensors measurements with a large uncertainty. To avoid this, it is sufficient to add additional sensors, such as gyroscopes, to correct the uncertainties and make the data more reliable and stable.

All the problems mentioned are common to all measurement systems, but in our case they combine to cause unreliable measurements, in addition to the fact that the technique used is not the best. The issues just mentioned can easily be solved by using better sensors, possibly digital, which do not require calibration procedures such as the one previously outlined.

Regarding the choice of the number of sensors, we can say, after analysing the available data, that the use of four accelerometers would be the best choice. However, we have seen that the results obtained are not very different between the use of two or four sensors. Therefore, the improvements we have seen do not justify the use of two additional acquisition instruments, because of the increase in

cost, which is almost double, because of the need to use a DAQ with a sampling frequency that allows simultaneous acquisition from 16 different channels, and because of the increase in calculation capacity needed to calculate the angles of bending, adduction, rotation.

With regard to the choice of sensor positioning, we have seen that the position used in Figure 4.6 can be considered valid, since the sensors are hypothetically positioned in the points where there is the greatest excursion of the angles to be calculated. In fact, let us suppose that we positioned both sensors on the same side of the leg instead of frontally and acquired the data with the measuring system so positioned. Since we are interested in the difference between the values calculated by the sensor placed on the thigh and the one on the shin, the configuration now proposed does not allow us to obtain the maximum obtainable value of relative rotation because we must remember that the human skin is an extremely flexible tissue and in this case, even if the shin rotates completely, the position where the sensor has been applied remains practically immobile.

In conclusion, the objective established for this project, namely to reconstruct the ROM of a knee, cannot be considered validated, due to the bad results. Moreover, this system is designed for a static system and it is necessary to develop an additional tool capable of evaluating a dynamic system. This experiment has highlighted that the use of accelerometers alone is not able to meet the system requirements we have imposed and we must radically change the whole acquisition system.

Chapter 5

Prototype design of the measurement system

In this section we expose the project on which this thesis is based. Starting from what has been described until this moment, a system consisting of Arduino Mega and IMUs is designed, whose aim is to reconstruct the dynamics of a knee. The device and sensors communicate via I2C communication protocol and the methods used in this chapter exploit what is presented in Section 3.3. In particular, quaternions are used to calculate the pose of the sensors and the Kalman filter is employed to remove uncertainties from the measurements. The prototype shown below is to be considered as an approximation of the final system; further studies need to be carried out to achieve more advanced results.

5.1 Analog model problems

The techniques used so far to reconstruct the dynamics of a knee are only applicable to analyse a static system. However, a skier is a dynamic system, and his movements, as well as his falls, are not predictable, even with the most advanced predictive techniques. In reality, a skier during a competition follows a pattern that is quite foreseeable, having to follow a precise trajectory. But it is during a fall that the athlete manifests movements not attributable to any specific model, and it is precisely this event we would like to forecast and avoid causing damage to the sportsman; otherwise, if this cannot be prevented, we would like to have an instrument that can quickly record and analyse the type of accident and the degree of injury to which the skier's knee has been subjected. Consequently, there is the need to improve the system we have already designed.

To enhance the project already developed, it is necessary to introduce additional sensors, as the accelerometers used cannot provide an acceptable reconstruction of a dynamic system. In addition, analog sensors have a variety of drawbacks, including the need of a sampler and quantizer to be able to analyse the data, which are usually external and bulky components, as seen in the previous chapter. Moreover, the techniques used until now to reconstruct the angles formed by a knee have considerable limitations, such as the impossibility of combining the various data, since in the measurement of the flexion and rotation angles it is necessary that the patient is lying down, while in the calculation of the valgus angle it is necessary that he is standing. Although what has been shown so far has proved to be reasonably valid in theory (as the results are not good), we need to change our approach from both an instrumentation and data analysis point of view in order to obtain results that meet our specifications. A final aspect that should not be neglected is that the analog system we have designed must be wired and connected to a PC in order to operate. Obviously ours was just an experiment, but to be considered valid it must be wireless or at least not require a physical connection to a data analysis device.

With this awareness in mind, we decid to leave aside the tools we had been using up to now and to look for new instruments that would enable us to achieve the desired objectives, knowing that in any case the results we might obtain would not be a point of arrival, but a basis for further studies. We start from our prior knowledge of dynamic systems, predictive models, data filtering and methods of representing a body in three-dimensional space and using our previous research into methods already used in the literature, observations of the sensors we could use and the practical feasibility study presented in Section 2.4, we try to design a prototype that is capable of meeting the specifications.

5.2 New instrumental setup

The first step in designing a new instrument which, to a first approximation, can be considered capable of meeting our requirements, is to understand whether what has been done so far can still be used, even minimally. We have already demonstrated that the use of accelerometers is not sufficient to meet our specifications, but it would be possible to use these sensors in combination with gyroscopes in order to make the data measured in this way more reliable. Gyroscopes are sensors that measure the angular velocity of a body. Accelerometers and gyroscopes

are often combined with magnetometers to form the *Inertial Measuremnt Unit* or *IMU*. These sensors are digital, which means that our design must be converted from an analog to a digital system.

The next step is to search on the market for an IMU that meet the following requirements:

- Each sensor has to be triaxial, since it is essential to analyse the movements made on the three axes;
- The accelerometer must measure at least ± 5 g, which is considered to be a significantly high value for the acceleration that a skier can be subjected to;
- *I2C* type communication, because it is a communication protocol widely used and easily accessible;
- It must be light, small and compact, as it needs to be applied to the body of an athlete without being uncomfortable;
- It must be compatible with Arduino, which is a system typically used to design prototypes.

After extensive research, our choice is felt on the sensor *Adafruit TDK InvenSense ICM-20948 9-DoF IMU* as it meets all our requirements and is compatible with both Arduino and Python.

The platform for reading and analysing the data coming from the sensors is chosen to be an Arduino, which is one of the best and most widely available devices for prototyping. Following careful research and examination of the specifications of each device that Arduino presents in its catalogue, we decided to use an *Arduino Mega 2560 Rev3*.

5.2.1 Arduino Mega 2560 Rev3

The Arduino Mega 2560 Rev3 (Figure 5.1) accommodates the *ATmega2560* microcontroller. The device has 54 digital input/output pins, 16 analogue inputs, 4 UARTs (hardware serial ports), a USB connection, a power jack, an ICSP header and a reset button. It can be powered by a USB cable or a special 5 V charger and the program to be executed by the microcontroller is flashed through the same USB power cable.

The *ATmega2560* processor works at a frequency of 16 MHz, it has 256 kB of flash memory, 4 kB of EEPROM and 8 kB of internal SRAM.

We chose this Arduino because it has a higher calculation capacity than other devices, it is cheap and has *SPI*, *I2C*, *USB* and *UART* communication channels. It is also compact and allows us, in the event of substantial modifications to our project, to add additional sensors without any problems. Finally, the operating temperature of the device is between $-40°C$ and $80°C$, which would hypothetically allow us to use it even in a mountain environment in winter.

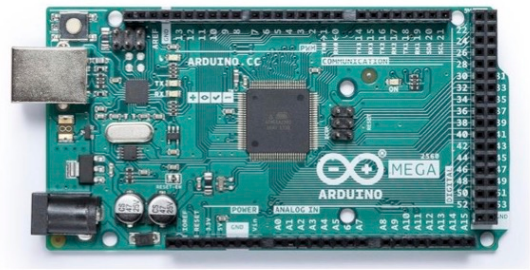

Figure 5.1: Arduino Mega 2560 Rev3

5.2.2 Adafruit TDK InvenSense ICM-20948 9-DoF IMU

The *Adafruit TDK InvenSense ICM-20948 9-DoF IMU* (Figure 5.2) allows measurements on the 9 axes using 16 *bit* analogue to digital converters that trasform the raw analogue data into data accessible via the *I2C* or *SPI*. The IMU is the *ICM20948* and it is composed by:

- 3 axis programmable gyroscope of ± 250 $dps, \pm 500$ $dps, \pm 1000$ dps and ± 2000 dps;
- 3 axis programmable accelerometers of ± 2 $g, \pm 4$ $g, \pm 8$ g and ± 16 g;
- 3 axis compass with a range $[-4900; +4900]$ μT

It has two *Stemma QT* connectors, which allow the sensor to be easily connected to an external device such as an Arduino. It has a 3 V or 5 V power supply pin; the voltage is then reduced to 1.8 V by an internal conditioning circuit to enable the *ICM20948* to be supplied. A pair of pins handles the *I2C* communication protocol (*SCL, SDA*), while it uses 4 pins for the *SPI* communication protocol. The Adafruit provides two Arduino libraries for acquiring, managing and processing data. In addition, the *ICM20948* has a *Digital Motion*

Processor or *DMP*, that allow computation of motion processing algorithms from the host processor, improving system power performance. It acquires data from accelerometers, gyroscopes, and magnetometers, processesing the data. Its purpose is to offload the timing requirements and processing power from the host processor.

Finally, the operating temperature of the sensor is between $-40°C$ and $85°C$, making it suitable for the application for which we would like to use it.

Figure 5.2: Back and front view of an Adafruit TDK InvenSense ICM-20948 9-DoF IMU

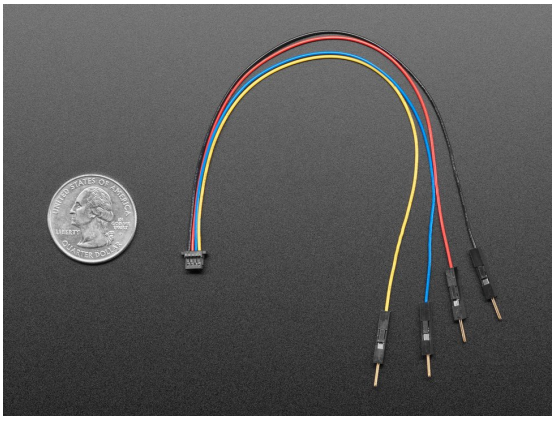

Figure 5.3: Adafruit STEMMA QT/Qwiic JST SH 4-pin Premium Male Head-ers Cable - 150mm Long

5.2.3 I2C protocol

The *I2C* or *Inter Integrated Circuit* is a synchronous serial communication bus that allows connection between lower speed peripheral and processors in short distance. This communication protocol uses only bidirectional lines: the *Serial Data Line* (*SDA*) and the *Serial Clock Line* (*SCL*), pulled up with resistors, with typical voltages of 5 V or 3.3 V (Figure 5.4). The *I2C* has a 7 *bit* address, so 128 different possible addresses, typically called *nodes*. However, 16 addresses are reserved, so the maximum number of devices connected on the same bus is 112.

A bus has two types of nodes:

- *Master node*, which generates the clock signal;
- *Slave node*, which synchronises itself to the clock signal.

Several masters can be present on the same bus and there are four distinct ways of operation with the *I2C* protocol:

- A master transmits: controls the clock and sends data to the slaves;
- A master receives: controls the clock but receives data from the slave;
- The slave transmits: the device does not control the clock but sends data on the bus;
- The slave receives: the device does not control the clock and receives data from the bus.

To start a communication, the master sends the start bit, followed by the address of the slave it wants to communicate with and a bit that is used to transfer information to the slave, i.e. to indicate if the master want to receive or send data. If the called slave exists, it takes control of the data line and sends an ACK signal. After that, the slave starts to transmit or receive data, depending on the operation it has to perform. When the data exchange is finished, a stop bit is sent and, if the master is receiving data, it sends an ACK for each byte received.

In our case, the Arduino acts as master, while the sensors operate as slaves: in fact, IMUs do not have an internal clock, so they have to synchronise with the clock generated by the Arduino Mega to transmit data. Note that the sensors are connected to the Arduino using four cables (Figure 5.3): the red cable is the 5 V or 3.3 V power supply, the black cable is the common ground, the yellow cable is the SCL and the blue one is the SDA.

5.3 Connection and system setup

Before to start to design a measurement device capable of reconstructing a dynamic system, we have to correctly connect the components and we have to open a communication channel between all actors. Figure 5.5 shows the schematic that allows the sensors to be connected to the correct pins on the Arduino Mega.

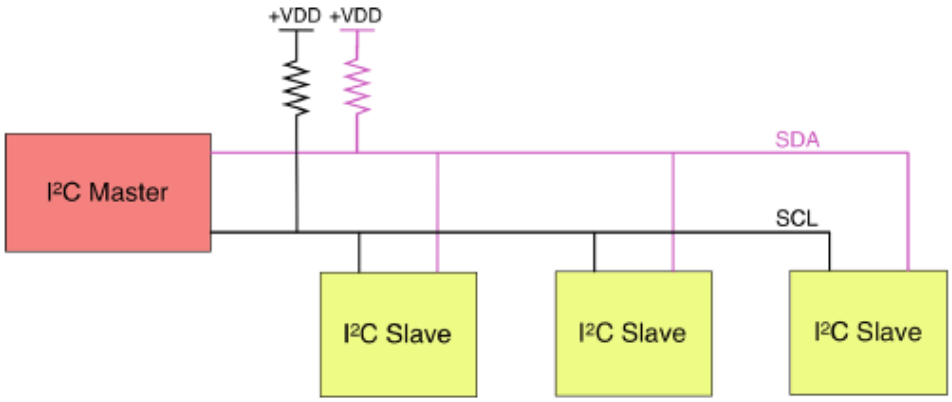

Figure 5.4: I2C Protocol with one master and three slaves

Although the pins used to connect the sensors to the I2C of the Arduino Mega are different, the bus on which they communicate is the same, operating in serial mode as shown in Section 5.2.3. As a result, each time the program running on the Arduino Mega asked the sensors to provide data, they operated in an alternating mode, ensuring that the connection to the SDA line is established one at a time, disconnecting after the communication is completed.

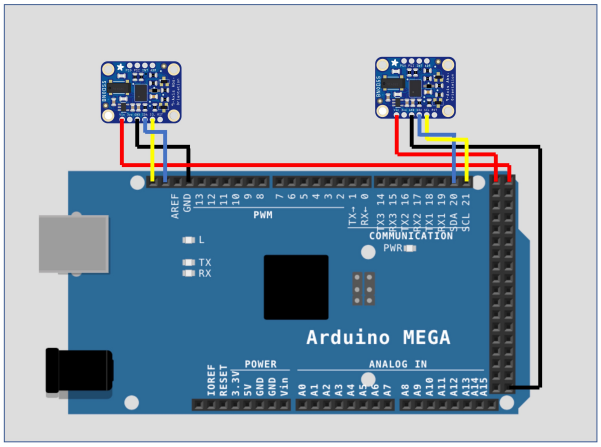

Figure 5.5: Graphical representation of Arduino Mega and sensors connection, where colors are used as follows: Red - +5 V, Black - GND, Yellow - SCL, Blue - SDA

To create a communication channel between master and slaves, we use a program written in Arduino that exploited the public library *Wire*, the library provided by Adafruit *Adafruit_ICM20948* and the IMU datasheet. As reported in [18], the address to which the IMU responds is set equal to 0x69, established by the manufacturer. When the master asks to receive information from the device with address 0x69, if it is connected to the line, it responds by sending an ACK and then the requested information. However, if two IMUs are simultaneously

connected to the same SDA, since they have the identical address, they will both respond, not allowing us to discriminate the sender. This aspect is very uncomfortable, as it is essential to be able to understand which sensor is sending the data. It is therefore necessary to change the address of at least one of the two sensors connected to the same I2C line. There are three ways to solve this problem, as explained in the datasheet [2]: in the first case, we can connect the SDO pin of the sensor to a digital port of the Arduino, then set this port as HIGH, so that the sensor can change its address from the default 0x69 to the new 0x68. In a very similar way, we can connect the CS pin of the sensor to a ground reference of the Arduino, ensuring the address change. However, these techniques have the disadvantage of having to introduce a new cable, making it even more inconvenient to use this wearable device, even though it is only a first prototype. The last technique consists of shorting the jumper indicated by the ADR serigraphy on the back of the sensor, visible in Figure 5.2, to allow the change from the old to the new address 0x68. This method is the easiest to implement and it does not introduce any new physical elements into the measurement system. After careful consideration, we decided to use the third technique on a sensor, soldering a small piece of tin onto the jumper, short-circuiting it.

To check the success of the address change and the possibility of correctly distinguishing the two IMUs connected to the Arduino and we proceed to install the Listing 5.1 in the memory of the Mega.

```
1 #include <Adafruit_ICM20948.h>
2 #include <Wire.h>
3
4 void setup()
5 {
6   Wire.begin();           //Start communication
7   Serial.begin(115200);   //Set baud rate ...
       communication
8 }
9
10 void loop()
11 {
12   byte error, address;
13   int nDevices = 0;
```

```
14
15   for(address = 1; address < 127; address++ ) ...
        //Scan of all possible addresses
16   {
17     Wire.beginTransmission(address);
18     error = Wire.endTransmission();
19
20     if (error == 0)   //If any device answer, print ...
          the corresponding address
21     {
22       Serial.print("I2C device found at address 0x");
23       if (address<16)
24         Serial.print("0");
25       Serial.print(address,HEX);
26       Serial.println("  !");
27       nDevices++;
28     }
29   }
30   if (nDevices == 0)    //If no devices answer
31     Serial.println("No I2C devices found\n");
32   else
33     Serial.println("");
34   delay(1000);          //Repeat every second
35 }
```

Listing 5.1: Arduino program for I2C scanning

After running the code, we display the result using the *Serial Monitor* integrated in the Arduino IDE, obtataining

```
1       I2C device found at address 0x68  !
2       I2C device found at address 0x69  !
```

5.3.1 Knee range of motion evaluation with prototype

After establishing a connection between the Arduino and the sensors, we can try to replicate what has been done so far on the analogical system also on this

prototype. We chose to do it for various reasons, in particular to become familiar with the new acquisition system and to see if there are better results with this system than those seen in the previous chapter.

The following stage after the start of communication between the various actors is to configure the IMUs, since they are composed of three different instruments, which can be switched on and off independently of each other and both the accelerometer, the gyroscope and the magnetometer can accept different ranges, depending on the application of these instrumentation. For this experiment we decide to use a range for the accelerometer equal to $\pm 8g$, while the gyroscope is set with a full scale of 1000 DPS, and magnetometers are turned off. Finally, we calculate the pitch, roll and yaw angles for each sensor and then we compute the flexion, rotation and adduction angles, as presented in Section 4.3.1. The full Arduino code can be found in Appendix C.

The purpose of this test is not to verify that our algorithm worked, as we have discussed this aspect extensively before, but to establish if what we are doing is correct or not. Consequently, we do not perform any proper experiments, but we simply position the sensors manually so that they assume precise angles and we verify the results. We place the sensors as if they are measuring a flexion of $0°, 45°, 90°$ and $120°$, then a rotation of $-10°$ and $30°$ and finally an abduction of $-10°$ and $10°$. Before performing these trials, we expected the results to be more or less similar to the actual data and indeed this is the case, as shown in Table 5.1. We do not review these aspects further, having analysed them in depth in the previous chapter, but we can consider ourselves satisfied with the results achieved. Nevertheless, it should be noted that the results obtained with these sensors are much better than those obtained using the analog devices, and this is probably due to the fact that the digital IMUs chosen are of a higher quality, providing more accurate measurements.

Once we have demonstrated that this measurement system is correctly configured following the procedures just described, we proceed to develop more complicated algorithms to reconstruct the dynamics of a knee.

5.4 Implementation of the measurement system prototype

Using the new measurement system, it is possible to implement algorithms that are more sophisticated than those seen up to now, since sensors of different types

Movement	True Angle	Meas Angle
Extension/Flexion	0°	1.96°
ϕ	+45°	46.60°
	+90°	88.46°
	+120°	84.92°
Int/Ext Rotation	−10°	−12.1°
ρ	+30°	29.44°
Adduction/Abduction	−10°	−11.65°
ρ	+10°	11.28°

Table 5.1: Experimental evaluation of ROM using the digital prototype

are available. At the same time, there is the problem of the method used to merge data from accelerometers and gyroscopes, as we decided to avoid using magnetometers. In order to carry out the task of sensor fusion, we took advantage of what was explained in Section 3.3, in particular we initially make a comparison between the Kalman and the complementary filter and then we explored the world of quaternions. Before proceeding to explain all this, it is essential to define the positioning of the sensors. Exploiting what we saw in Section 4.4.4, we decide, having only two IMUs available, to place them exactly as in Figure 4.6, as this placement proved to be effective.

It is necessary to add two clarifications: firstly, even during these experiments, we have avoided attaching the sensors to a kneepad, as it only serves the purpose of acting as a frame for positioning the sensors, but does not affect the measurements at all. Secondly, it must be pointed out that no sensor calibration procedure was planned when using this measurement system, as after several experiments we found that the data in the datasheet are consistent with the measures taken. Since there is no need to modify the sensitivity and offset parameters of the accelerometer and gyroscope, the following values are to be considered calibrated with the values in the datasheet.

5.4.1 Kalman filter and complementary filter sensor fusion

The new measuring system is similar to the analogical already exposed and consists of two different sensors, positioned respectively on the thigh and on the shin. Now, in order to determine the dynamics of the knee it is essential to know precisely the orientation of each sensor and then calculate the relative orientation between one and the other. We have already analysed in detail that the use of accelerometers alone allows us to have a good knowledge of the initial position

that the IMU assumes in the absence of movements; however, what we want to monitor is a dynamic system, which moves continuously. It is for this reason that we have introduced gyroscopes, which are sensors able to measure the angular velocity undergone by a body and, consequently, can help us achieve our goal of knowing the orientation of our measurement system at all times.

The use of gyroscopes alone would not allow us to reach our aim because of an intrinsic problem of these instruments, which causes a progressive deviation of the calculated measurement from the real value. In fact, we could think that integrating over the time the measurements coming from the gyroscopes, we might know the orientation of the body at each instant. However, integrating this quantity also integrates the measurement error made by the instrument itself and this leads to the fact that the overall error increases continuously.

To solve the problems of both sensors it is therefore necessary to use them simultaneously. Accelerometers can be used to know the initial position of the body and gyroscopes can be used to calculate the movements made and thus update the orientation of the body. It is also possible to reset the gyroscope angle after a certain amount of time and maintain knowledge of body orientation using the accelerometers, thus avoiding the problem of drift and maintaining valid results.

The most widely used sensor fusion algorithm for combining data from accelerometers and gyroscopes is the Kalman filter. In our case, we define a Kalman filter for each angle (so three filters in total) which is initialised with the respective pitch, roll and yaw positions calculated as explained in Section 4.3, and then, at each instant, updates the orientation using data from gyroscopes. For a better comprehension, consider a single angle. Using the same terminology as in the Section just cited, we define the state vector $\boldsymbol{x}(k)$ as

$$\boldsymbol{x}(k) = \begin{bmatrix} \mu(k) \\ \dot{\mu}_b(k) \end{bmatrix} \tag{5.1}$$

where $\mu(k)$ is the angle and $\dot{\mu}_b(k)$ is the bias at the $k-th$ instant, based upon the measurements from the accelerometer and gyroscope. The bias is intended as the amount the gyro has drifted and means that we can get the true rate by subtracting the bias from the gyro measurement. The measurement $\boldsymbol{y}(k)$ is given by the current state $\boldsymbol{x}(k)$ multiplied by the C matrix plus the measurement noise

$\boldsymbol{v}_y(k)$. The system we use is therefore equal to

$$\begin{aligned} \boldsymbol{x}(k) &= A\boldsymbol{x}(k-1) + B\boldsymbol{u}(k) + \boldsymbol{v}_x(k) \\ \boldsymbol{y}(k) &= C\boldsymbol{x}(k) + \boldsymbol{v}_y(k) \end{aligned} \tag{5.2}$$

where we define

$$A = \begin{bmatrix} 1 & -\Delta k \\ 0 & 1 \end{bmatrix} \qquad B = \begin{bmatrix} \Delta k \\ 0 \end{bmatrix} \qquad C = \begin{bmatrix} 1 & 0 \end{bmatrix} \tag{5.3}$$

Note that we use the symbol Δk to denote the interval between two consecutive discrete time instant. The control input $\boldsymbol{u}(k)$ that we chose is the gyroscope measurement in radiant per second ($\frac{rad}{s}$) at time k, also called the rate $\dot{\mu}(k)$. This makes sense as we compute the angle $\mu(k)$ when we multiply the rate $\dot{\mu}(k)$ by the delta time Δk and since we cannot calculate the bias directly based on the rate we set the second element of the matrix $\boldsymbol{B}$ equal to 0.

The noise of the measurement $\boldsymbol{v}_x(k)$ is Gaussian distributed with a zero mean and covariance matrix Q equal to

$$Q = \begin{bmatrix} Q_\mu & 0 \\ 0 & Q_{\dot{\mu}_b} \end{bmatrix} = \begin{bmatrix} \sigma_{angle} & 0 \\ 0 & \sigma_{bias} \end{bmatrix} \tag{5.4}$$

where σ_{angle} is the noise measurement of the sensor, while σ_{bias} is the bias to which the gyroscope is subjected. Also the noise of the measurement $\boldsymbol{v}_y(k)$ is Gaussian distributed with a zero mean and covariance matrix R, set equal to

$$R = var[\boldsymbol{v}_y(k)] = \sigma_{MU} \tag{5.5}$$

where σ_{MU} is the measurement uncertainty. R and Q are time invariant. The values for σ_{angle}, σ_{bias} and σ_{MU} can be calculated using the datasheet [18].

If we consider the Kalman filter implemented to evaluate the pitch angle, the initial conditions are known and equal to

$$\boldsymbol{x}_\rho(0) = \begin{bmatrix} \rho_0 \\ 0 \end{bmatrix} \tag{5.6}$$

The same state vector has been introduced considering the roll and yaw angles. Note that ρ_0 (as also ϕ_0, θ_0) is the initial pitch, i.e. calculated as soon as the measuring system is switched on and while stationary. Therefore the initial un-

certainty $P(0)$ is very small and has been chosen equal to

$$P(0) = \begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix} = \begin{bmatrix} 0.001 & 0 \\ 0 & 0.001 \end{bmatrix} \tag{5.7}$$

After initialising it, the Kalman filter implemented by us is based on 7 steps:

Step 1 The current state is computed as

$$\hat{\boldsymbol{x}}(k|k-1) = A\hat{\boldsymbol{x}}(k-1|k-1) + B\dot{\mu}(k) = \tag{5.8}$$

$$\begin{bmatrix} \mu(k|k-1) \\ \dot{\mu}_b(k|k-1) \end{bmatrix} = \begin{bmatrix} \mu(k-1|k-1) + \Delta k(\dot{\mu}(k|k-1) - \dot{\mu}_b(k-1|k-1)) \\ \dot{\mu}_b(k-1|k-1) \end{bmatrix} \tag{5.9}$$

so the a priori estimate of the angle is $\hat{\mu}(k|k-1)$ is equal to the estimate of the previous state $\hat{\mu}(k-1|k-1)$ plus the unbiased rate times the delta time Δk. Since we can not directly measure the bias the estimate of the a priori bias is just equal to the previous one.

Step 2 We compute the previous error covariance matrix as

$$P(k|k-1) = AP(k-1|k-1)A^T + Q(k) \tag{5.10}$$

$$= \begin{bmatrix} P_{00} + \Delta k(\Delta k P_{11} - P_{01} - P_{10} + Q_\mu) & P_{01} - \Delta k P_{11} \\ P_{10} - \Delta k P_{11} & P_{11} + Q_{\dot{\mu}_b}\Delta k \end{bmatrix} \tag{5.11}$$

Step 3 The innovation is calculated as

$$\boldsymbol{e}(k) = \boldsymbol{x}(k) - C\hat{\boldsymbol{x}}(k|k-1) \tag{5.12}$$

$$= \mu(k) - \mu(k|k-1) \tag{5.13}$$

Step 4 The innovation covariance is computed as

$$\Lambda(k) = CP(k|k-1)C^T + R(k) \tag{5.14}$$

$$= P_{00}(k|k-1) + var[\boldsymbol{v}_y(k)] \tag{5.15}$$

Step 5 Then we compute the Kalman gain

$$L(k) = P(k|k-1)C^T\Lambda(k)^{-1} \tag{5.16}$$

$$= \frac{\begin{bmatrix} P_{00} \\ P_{10} \end{bmatrix}(k|k-1)}{\Lambda(k)} \tag{5.17}$$

Step 6 Finally the prediction stage estimates

$$\hat{\boldsymbol{x}}(k|k) = \hat{\boldsymbol{x}}(k|k-1) + L(k)\boldsymbol{e}(k) \tag{5.18}$$

$$= \begin{bmatrix} \mu \\ \dot{\mu}_b \end{bmatrix}(k|k-1) + \begin{bmatrix} L_0\,\boldsymbol{e} \\ L_1\,\boldsymbol{e} \end{bmatrix}(k) \tag{5.19}$$

Step 7 In conclusion, we compute the a posteriori error covariance matrix

$$P(k|k) = (I - L(k)C)P(k|k-1) \tag{5.20}$$

$$= \begin{bmatrix} P_{00} & P_{01} \\ P_{10} & P_{11} \end{bmatrix}(k|k-1) - \begin{bmatrix} L_0\,P_{00} & L_0\,P_{01} \\ L_1\,P_{00} & L_1\,P_{01} \end{bmatrix} \tag{5.21}$$

With regard to the complementary filter, we simply calculate it using α equal to 0.93, thus giving a very high importance to the measurement taken by the gyroscope, while a much lower importance is given to the angle computed using the values coming from the accelerometer. The equations that return the orientation calculated with the complementary filter are therefore

$$\begin{aligned} \rho(k) &= \angle_\rho(k) = 0.93 \cdot (\angle_\rho(k-1) + m_{\rho,gyro} \cdot \Delta k) + 0.07 \cdot \rho_{acc}(k) \\ \phi(k) &= \angle_\phi(k) = 0.93 \cdot (\angle_\phi(k-1) + m_{\phi,gyro} \cdot \Delta k) + 0.07 \cdot \phi_{acc}(k) \\ \theta(k) &= \angle_\theta(k) = 0.93 \cdot (\angle_\theta(k-1) + m_{\theta,gyro} \cdot \Delta k) + 0.07 \cdot \theta_{acc}(k) \end{aligned} \tag{5.22}$$

where $\Delta k = 0.06\ s$, corresponding to the time between two consecutive samples is the same interval of the Kalman filter.

Now consider only the equation for calculating the pitch angle. The quantity $m_{\rho,gyro}$, that is the measurement of the angular velocity provided by the gyroscope, is multiplied by the interval Δk in order to compute the pitch angle calculated by the gyroscope. Instead $\rho_{acc}(k)$ is the pitch angle evaluated by using (4.1). Since both the gyroscope and the accelerometer allow us to determine the orientation angle of the sensor, but since the angle computed by the gyroscope

is more reliable for the reasons explained above, we have chosen to use $\alpha = 0.93$. This value of α was chosen after some experimental tests, as this quantity represented the best compromise between the measurements coming from the different sensors.

These filters are applied to each sensor and to evaluate the relative orientation between them, the resulting values of pitch, roll and yaw of the IMU on the shin are subtracted with those of the sensor on the thigh, obtaining the pitch, roll and yaw angles that define the relative rotation between the two IMUs. In formulas, using the notation in Figure 4.6, we define the relative rotation between the two sensors $(\rho_{12}, \phi_{12}, \theta_{12})$ as

$$\begin{aligned} \rho_{12} &= \rho_2 - \rho_1 \\ \phi_{12} &= \phi_2 - \phi_1 \\ \theta_{12} &= \theta_2 - \theta_1 \end{aligned} \tag{5.23}$$

The full Arduino code that compute the Kalman filters and the complementary filters can be found in the Appendix D.

5.4.2 Quaternions sensor fusion

To calculate the quaternions, the capabilities of the ICM[1] are exploited, in particular the Digital Motion Processor. The IMU used has a DMP, which offloads the calculation of the motion processing algorithms from the host processor. This allows the user to have tools available that would otherwise require external programming. Therefore by utilising this functionality, we use the quaternions calculated directly by the IMU, we processe them, and we extract the information we need.

Through the function `myICM_0.initializeDMP()`[2] we activated the DMP on IMU, then we use a function to obtain six degree of freedom quaternions, namely, which are calculated using measurements from both accelerometers and gyroscopes. In order to avoid the use of data that are not part of the acquisition but are left inside the acquisition buffer, the FIFO memory is first activated and then both the DMP and the FIFO memory are reset. These operations are done for each sensor used. Note that it is necessary to normalise the quaternions provided by the ICM, because as the datasheet indicates, they are not unit

[1] Recall that this is the name of the IMU

[2] Note that the object `myICM_0` is the object that we creat in Arduino to manage the IMU

quaternions. Using the JPL nomenclature and the same one used in Figure 4.6, we define the quaternions computed by the IMU on the thigh and shin as

$$\begin{aligned} \boldsymbol{q}_1 &= [\eta^1 \ \epsilon_i^1 \ \epsilon_j^1 \ \epsilon_k^1] \\ \boldsymbol{q}_2 &= [\eta^2 \ \epsilon_i^2 \ \epsilon_j^2 \ \epsilon_k^2] \end{aligned} \tag{5.24}$$

Once we have acquired the quaternions calculated directly by the processor, it is possible to elaborate them as explained in Section 3.3.3. Again, since we need the relative rotations between the thigh and shin sensors, we calculate pitch, roll and yaw, but this time instead of doing this for each sensor and then subtracting the results, we follow a different procedure. Knowing that quaternions represent rotation in three-dimensional space, we subtract the quaternion relative to the sensor on the shin with the one on the thigh, obtaining the quaternion that expresses the relative rotation between the two IMUs. We then proceed to derive the pitch, roll and yaw of the difference between the quaternions with the equations (3.28)-(3.30), thus obtaining what we are looking for. This process allow us to save computationally, as we simply have to calculate the angles of interest once, without having to do this for every sensor at every time.

In formulas, since we have at our disposal the imaginary part of the quaternions provided by the DMP, namely $(\epsilon_{i,1},\ \epsilon_{j,1},\ \epsilon_{k,1})$ and $(\epsilon_{i,2},\ \epsilon_{j,2},\ \epsilon_{k,2})$, we compute the difference between this two vector $(\epsilon_{i,12}, \epsilon_{j,12}, \epsilon_{k,12})$ and then we normalize the result, obtaining the unit quaternion expressing the relative rotation between the two sensors $\boldsymbol{q}_{12}$. The normalisation is made taking into account the fact that the norm of a unit quaternion is equal to 1, therefore

$$\boldsymbol{q}_{12}^2 = \eta_{12}^2 + \epsilon_{i,12}^2 + \epsilon_{j,12}^2 + \epsilon_{k,12}^2 = 1 \Longrightarrow \eta = \sqrt{1 - \epsilon_{i,12}^2 - \epsilon_{j,12}^2 - \epsilon_{k,12}^2} \tag{5.25}$$

An interesting feature of this algorithm is that when the system is switched on, the positions of the two IMUs are taken as reference positions. Consequently, the quaternion expressing the relative rotation between the two sensors is immediately calculated and each subsequent rotation is referred as a rotation undergone by the IMUs with respect to their initial relative position.

The full Arduino code that compute the quaternion sensor fusion can be found in the Appendix E.

5.5 Discussion of results

In order to be able to evaluate the goodness of the algorithms just described, it is necessary to carry out practical experiments. We manually place the sensors in known positions and we evaluate the results calculated using the Kalman filter, the complementary filter and the quaternion filter, comparing them with the values they should assume.

We use the **Absolute Error** (**AE**) and the **Root Mean Square Error** (**RMSE**) to make the static comparisons. To compute the dynamic behaviour of the measurment system we exploit the **Rise Time** (t_r), the **Settling Time** (t_s) and the **Overshoot** (M_p), defined as:

- t_r as the time the signal takes to go from 10% to 90% of the total excursion;
- t_s as the time it takes for the signal to reach its final state so that the error is less than 2%, where error is the difference between the signal and its final state;
- M_p as the percentage of the maximum signal peak value.

5.5.1 Kalman filter and complementary filter results

The first experiment we performed was a static experiment, meaning we placed the sensors in a known position and we evaluated the results obtained with both the Kalman filter and the complementary filter. Table 5.2 shows the calculated values. What is immediately apparent if we compare these data with those obtained using the analog system is that now we obtain the desired angles with a very small error. This remarkable improvement is probably due both to the use of gyroscopes, because they stabilise the measurement, and to the fact that the sensors used are much better than their analogue correspondences.

Another very interesting result that can be observed from the data obtained is that the complementary filter and the Kalman filter have more or less the same measurement errors, so that for static measurements they both work correctly.

At this point we could have said that both filters work very well, but our aim is to design a system capable of monitoring a dynamic system, not a static one. We therefore performed experiments capable of evaluating the dynamic behaviour of the acquisition system, evaluating the quantities typical of dynamic systems. Table 5.3 shows the pitch, roll and yaw values calculated using the Kalman filter and the complementary filter for a dynamic signal ranging from 0°

True Angle	CF	EA	RMSE	KF	EA	RMSE
Pitch ρ						
0°	0.10°	0.10°	0.10°	0.07°	0.07°	0.07°
45°	46.41°	1.41°	1.45°	46.43°	1.43°	1.54°
60°	59.69°	0.31°	0.36°	59.65°	0.35°	0.49°
90°	89.50°	0.50°	0.50°	89.48°	0.52°	0.52°
Roll ϕ						
0°	−0.06°	0.06°	0.06°	0.07°	0.07°	0.07°
45°	44.77°	0.23°	0.25°	44.78°	0.22°	0.35°
60°	59.91°	0.09°	0.14°	59.94°	0.58°	0.23°
90°	89.24°	0.76°	0.76°	89.23°	0.77°	0.77°
Yaw θ						
0°	−0.02°	0.02°	0.02°	−0.02°	0.02°	0.02°
45°	45.12°	0.12°	0.66°	45.08°	0.08°	0.88°
60°	60.89°	0.89°	0.99°	60.95°	0.95°	1.22°
90°	89.88°	0.12°	0.12°	89.89°	0.11°	0.11°

Table 5.2: Result of the static test with the Kalman filter and the complementary filter, where **CF** is the complementary filter and **KF** is the Kalman filter

to 90°. Analysing these data, we could think that the complementary filter has a better performance than the Kalman filter. In particular, we can seen that the overshoot of the complementary filter is always the smaller of the two (except for the pitch angle). However, by analysing Figure 5.6, we can see that even though the data are better, the signal calculated using a complementary filter is not able to follow the raw data correctly, unlike the signal calculated by the Kalman filter, which follows the raw data with good approximation. This behaviour also manifests itself in the calculation of roll and yaw.

5.5.2 Quaternions results

Due to the method of calculating quaternions using DMP, there is no comparison with the raw data representing the real motion of the measuring instrument. Therefore, no analysis is performed on the results obtained with the quaternions, but it is nevertheless possible to observe that even when the device is stationary, the calculated pitch, roll and yaw data are not stable, but oscillated around the value representing the true position. This is probably due to the fact that no method was designed to compensate for gyro drift when the quaternions are calculated. As a simple example, a graphical representation of the angles calculated using unit quaternions is shown in Figure 5.7. This graph also shows how we han-

	t_r $[s]$	t_s $[s]$	M_p [%]
Pitch ρ			
Raw Data	11.99	45.44	1.07
Complementary	29.94	74.87	86.48
Kalman	12.58	47.52	0.26
Roll ϕ			
Raw Data	33.16	99.82	35.52
Complementary	26.97	98.50	4.99
Kalman	28.99	99.70	23.19
Yaw θ			
Raw Data	21.72	100.74	9.51
Complementary	30.37	98.49	0
Kalman	22.67	100.63	6.49

Table 5.3: Result of the dynamic test with the Kalman filter and the complementary filter

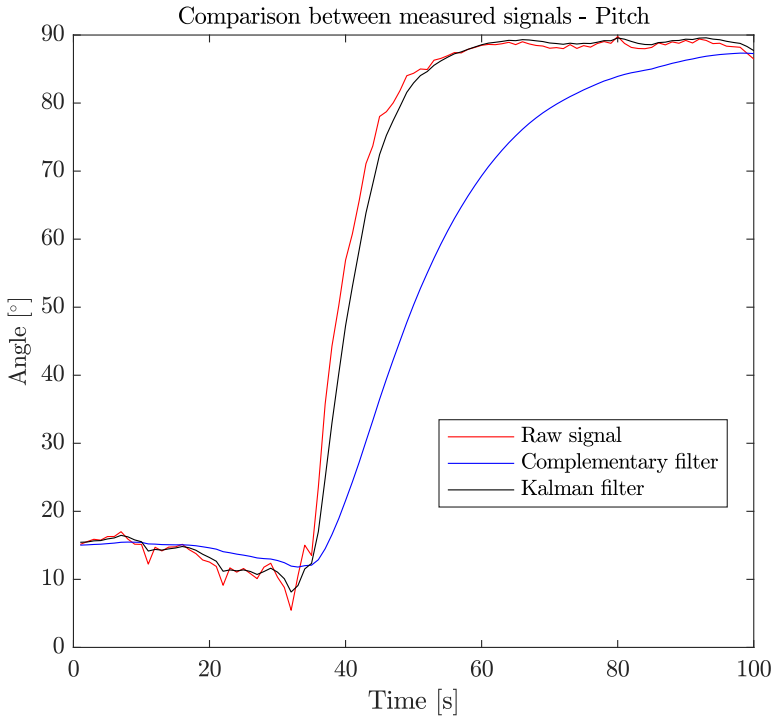

Figure 5.6: Comparison of the pitch between raw data, signal computed with Kalman filters and signal computed with complementary filter

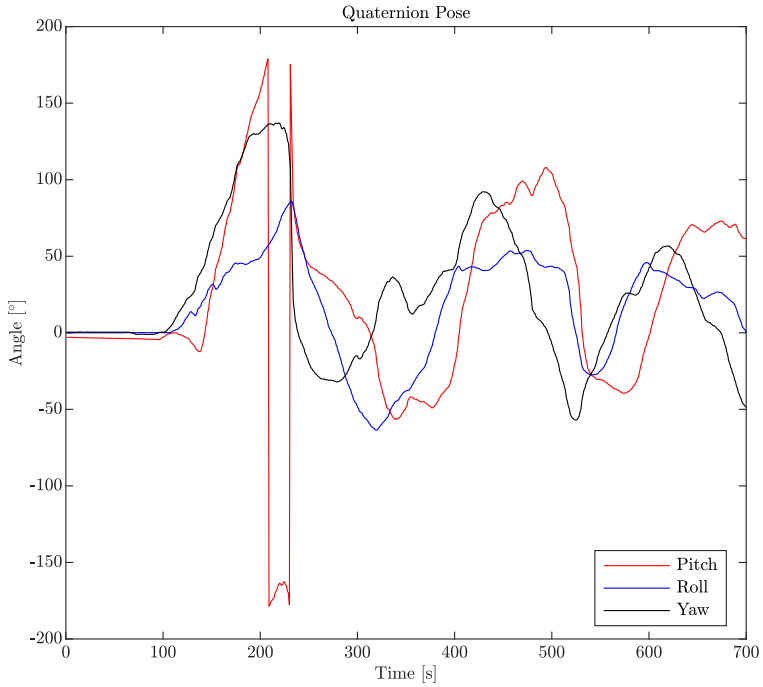

Figure 5.7: Graphical representation of the pose estimated using quaternions

dled the case where one of the angles exceeds the limit of 180°: when this angle is reached, since the unit quaternions can represent a rotation and the inverse of the same quaternion represents the same rotation, but in the opposite direction, we decided to change the sign of the angle, so as to have a measurement range of 360°.

5.5.3 Considerations about prototype designed

The digital prototype designed achieves the minimum standards of the specifications we set at the beginning of the thesis. The use of gyroscopes greatly improves the performance of the measurement system, as they stabilise the measures and provide the necessary information to complete the previously designed analogical system. In addition to the improvement of the static behaviour of the model, they also make it possible to create a tool capable of monitoring a dynamic system.

Through the use of the Kalman filter, which allows to combine the measurements made with accelerometers and gyroscopes, we are able to follow the real relative orientation between the sensor placed on the thigh and the one placed on the shin. This technique could be improved more by better defining the quantities expressing the measurement uncertainties, i.e. the noise and measurement

covariance matrices $R(k)$ and $Q(k)$. In addition, the use of the Kalman filter enabling us to clean the raw data from any noise and uncertainties, thus allowing us to have more precise measurements of the relative orientation between the two IMUs.

The complementary filter, despite having a very low computational cost, does not provide satisfying results, as a simple weighted average of the measurements from the two different types of sensors is not sufficient to make the system reliable. Its use is not recommended for our purpose, as it is essential that the prototype is reliable and fast, which are requirements not met by the signal. However, we could think of improving this algorithm by introducing a method that allows the raw signal to be followed more quickly, since as can be seen from the graphs, the signal coming out of the complementary filter is totally free of any noise and uncertainty, making it perfect for clean and, hypothetically, error-free measurement data.

Unfortunately, it is not possible to define the goodness of angles calculated using quaternions, but in theory their use is most recommended, as they can easily provide quantities that express orientation.

In conclusion, the project shown in this chapter is to be considered a good basis for future studies. A possible improvement on what has been shown so far could be the implementation of a Kalman filter for quaternions, in order to realise an algorithm with low computation, but at the same time very reliable.

Chapter 6

Conclusions

6.1 Considerations on the systems

The aim of this thesis was to solve one of the problems of designing a mechanism to prevent knee injuries in skiers. Since this objective is very large, we chose to focus on the design of a mechanism capable of monitoring the dynamics of the joint. Initially, a literature search was carried out to establish the fact that the knee is the most exposed part of a skier. After confirming this hypothesis, it was necessary to understand the dynamics that cause rupture of the anterior cruciate ligament in skiers. Having understood that this type of injury occurs in combination with rotation and hyperextension or hypereflexion of the knee, we decided to use inertial sensors to allow us to follow the dynamics of the joint. The best choice for positioning the sensors is to place one on the thigh, just above the patella, while another is placed on the shin. The optimal number of sensors to use is four, but the benefits of this quantity are not high enough to recommend them over the two used in the experimental trials. We therefore conducted a further literature search to assess the state of the art on reconstructing knee dynamics. We took our inspiration from two papers in particular and designed a first measurement system using only analog accelerometers to reconstruct the range of motion of the knee. We devised a procedure that would theoretically allow us to perform this task, but the results we obtained were not as good as we would have expected. As a consequence, we decided to discard this measurement system in order to develop a better one using digital sensors. This decision led us to develop a measuring system that uses data from accelerometers and gyroscopes to enable us to monitor a dynamic system in real time. In particular, we used the potential of the Kalman filter, the complementary filter and quaternions to

perform this task. The results obtained have permitted us to state that the initial aim that led to the elaboration of this thesis has been achieved, even if the data obtained are not to be considered as a point of arrival, but rather as a starting point for further developments. In fact, it is clear that the goals reached with the study presented in this paper are not good enough to ensure the use of the measurement system developed by us in a hostile and dynamic place such as a ski slope during a race. It is also evident that, using the results obtained in this thesis, reaching the final target is much easier than when this study was started, thanks to the mistakes made with the first measurement system and the considerable progress achieved using the prototype based on Arduino and digital IMUs.

6.2 Future Work

The work described in this thesis can, and indeed must, be improved in order to find an application outside the controlled environment of a laboratory. First of all, a new embedded system must be designed to perform the same functions as outlined in Chapter 5.4, in order to make the measurement system more robust and to enable field tests to be carried out. It is necessary to improve the algorithms designed until now, making them more robust and reliable, perhaps by implementing a Kalman filter for quaternions. If considered necessary, a procedure for calibrating the measurement system could be devised, without it being too complicated or troublesome for the wearer of the measurement system. It would also be advisable to make the communication between the sensors and the processor wireless, so as to avoid using cables that could disturb the user or cause errors in data transmission. It would also be recommendable to introduce a local backup system, which would allow the recording of data relating to the orientation of the knee, so that, in the event of an unavoidable accident, the magnitude of the injury would be known immediately, allowing doctors and health workers to act in the best possible way. Connected to this last point, we could also think of adding a telemetry system to send data to an external device that allows continuous monitoring of the skier's performance. It would be interesting to see if introducing magnotometers into the system would guarantee an improvement in the measurements taken, and it would also be interesting to use additional sensors, such as bioimpedance or fibre optics, to help the measurement system assess the actual orientation between the thigh and shin. In connection with the design

of the active safety system for skiers, it is essential to carry out further studies on the maximum displacements and rotations that a knee can be subjected to without injury. Once the limiting conditions are reached, it is necessary to activate a warning mechanism when an extreme situation is reached. Ideally, the final measurement system should be designed to evaluate dangerous situations for the knee and, if a dangerous situation is detected, to allow the activation of a useful mechanism to strengthen the knee, avoiding the extreme condition in which the injury occurs and, at the same time, to communicate danger information to a ski releasing mechanism, so as to guarantee the removal of the constraint on the foot, ensuring the full effectiveness of the injury prevention system.

Appendix A

Anatomical planes and axes in human body

In the human body there are three cardinal planes and three cardinal axes: *sagittal, tranverse* and *longitudinal*. These planes and axes of motion are typically depicted in the context of a person standing in the anatomic position, as in the Figure A.1. Regarding the cardinal planes, we can say that:

- The **longitudinal or sagittal plane** runs parallel to the the sagittal suture of the skull, dividing the body into right and left sections. Flexion and extension movements are performed along this plane.
- The **coronal or frontal plane** runs parallel to the coronal suture of the skull, dividing the body into front and back sections. Adduction and adduction movements are performed along this plane.
- The **transverse or horizontal plane** courses parallel to the horizon and divides the body into upper and lower sections. Internal (medial) and external (lateral) rotational movements are performed along this plane.

Instead, for what concerns the cardinal axes, we can say that:

- The **sagittal axis** passes horizontally, from posterior to anterior and is formed by the intersection of the sagittal and transverse planes;
- The **transverse axis** passes horizontally from left to right and is formed by the intersection of the frontal and transverse planes;
- The **longitudinal axis** passes vertically from inferior to superior and is formed by the intersection of the sagityal and frontal planes

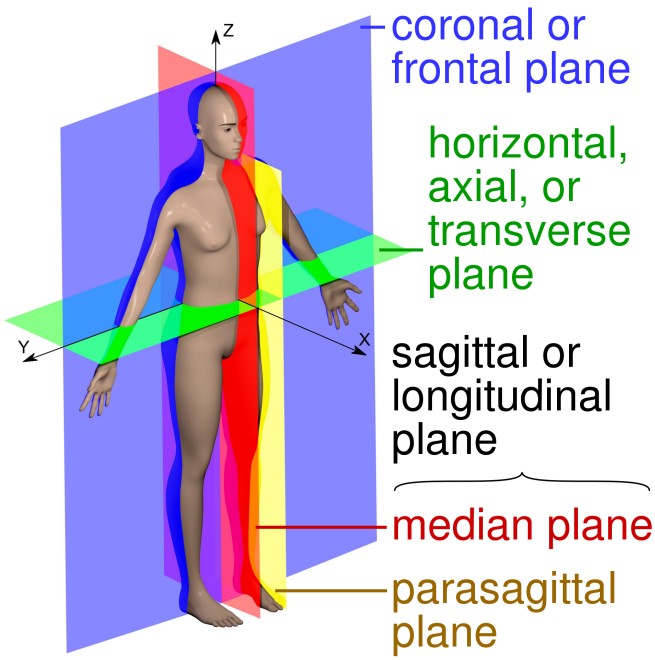

Figure A.1: The three cardinal planes and the three cardinal axis of the body are shown referred to a person in the anatomic position

Sometimes the **parasagittal plane** is also used, which has the same characteristics as the sagittal, but ends at the shoulder.

Appendix B

MATLAB algorithm for knee flexion evaluation

This section provides the MATLAB code used to calculate the knee flexion angle (Listing B.1). Similar programs have been created to calculate the angle of adduction and rotation of the knee, substituting the values of roll and pitch and yaw following the indications given in Section 4.3.1.

```
%% Initialization
s = daq.createSession('ni');       %Session creation

Fs = 2000;                         %2 [kS] Sample frequency
time = 0.005;                      %[s]
Nsamples = Fs * time;              %Samples
s.DurationInSeconds = time;        %Acquisition time
s.Rate = Fs;

%DAQ channels configurations
%First accelerometer thigh
ch1 = s.addAnalogInputChannel('Dev1','ai1','Voltage');
ch1.Name = 'x_axis';
ch1.TerminalConfig = 'SingleEnded';
ch1.Range = [-5 +5];
ch2 = s.addAnalogInputChannel('Dev1','ai9','Voltage');
ch2.Name = 'y_axis';
ch2.TerminalConfig = 'SingleEnded';
ch2.Range=[-5 +5];
ch3 = s.addAnalogInputChannel('Dev1','ai2','Voltage');
ch3.Name = 'z_axis';
```

```
ch3.TerminalConfig = 'SingleEnded';
ch3.Range = [-5 +5];
%Second accelerometer shin
ch4 = s.addAnalogInputChannel('Dev1','ai10','Voltage');
ch4.Name = 'x_axis';
ch4.TerminalConfig = 'SingleEnded';
ch4.Range = [-5 +5];
ch5 = s.addAnalogInputChannel('Dev1','ai3','Voltage');
ch5.Name = 'y_axis';
ch5.TerminalConfig = 'SingleEnded';
ch5.Range=[-5 +5];
ch6 = s.addAnalogInputChannel('Dev1','ai11','Voltage');
ch6.Name = 'z_axis';
ch6.TerminalConfig = 'SingleEnded';
ch6.Range = [-5 +5];

%% Acquisition
s.IsContinuous = 1;
s.addlistener('DataAvailable', @RealTimeTrack);
s.startBackground();
figure

function RealTimeTrack(src,event)
    persistent i;
    if(isempty(i))
        i = 1;
    else
        i = i + 1;
    end
%   Loading of calibration values
    cal_values_1 = load('calibrated_accelerometer1');
    cal_values_2 = load('calibrated_accelerometer2');
    data = event.Data;
%   Adjustment of measurements using calibration data
    acc_1 = CALIBRATION_ACC(data(:,1:3), ...
        cal_values_1.calibrated_values(:,1), ...
        cal_values_1.calibrated_values(:,2));
    acc_2 = CALIBRATION_ACC(data(:,4:6), ...
        cal_values_2.calibrated_values(:,1), ...
        cal_values_2.calibrated_values(:,2));
    N = length(acc_1);

%   Roll [rad]:angle of Y-axis wrt the ground
    phi_1 = atan2(acc_1(N,2),sqrt(acc_1(N,1).^2 + ...
```

```
        acc_1(N,3).^2));
62      phi_2 = atan2(acc_2(N,2),sqrt(acc_2(N,1).^2 + ...
        acc_2(N,3).^2));
63  %   Yaw [rad]: angle of Z-axis wrt the gravity
64      theta_1 = atan2(sqrt(acc_1(N,1).^2 + ...
        acc_1(N,2).^2),acc_1(N,3));
65      theta_2 = atan2(sqrt(acc_2(N,1).^2 + ...
        acc_2(N,2).^2),acc_2(N,3));
66  %   Evaluation of the knee flexion
67      flexion_extension_ZY_pitch = rad2deg(phi_2 - phi_1);
68      flexion_extension_ZY_yaw = rad2deg(theta_2 - theta_1);
69
70  %   Plot
71      subplot(2,1,1)
72      plot(i,flexion_extension_ZY_roll, 'ro')
73      title('Knee Flexion/Extension with Roll ')
74      xlabel('Samples')
75      ylabel('Angle [Deg]')
76      hold on
77      subplot(2,1,2)
78      plot(i,flexion_extension_ZY_yaw, 'ro')
79      title('Knee Flexion/Extension with Yaw ')
80      xlabel('Samples')
81      ylabel('Angle [Deg]')
82      hold on
83      drawnow;
84  end
```

Listing B.1: MATLAB program to compute knee flexion and extension with analog system

Appendix C

Arduino algorithm for knee flexion evaluation

This section provides the Arduino code used to calculate the knee flexion angle (Listing C.1). Similar programs have been created to calculate the angle of adduction and rotation of the knee, substituting the values of roll and pitch and yaw following the indications given in Section 4.3.1. The libraries were taken from [3] and modified according to our requirements.

```
#include <Adafruit_ICM20X.h>
#include <Adafruit_ICM20948.h>
#include <Wire.h>

Adafruit_ICM20948 icm1, icm2;   //Building of the ...
    objects

void setup() {
   Serial.begin(115200);        //Set baud rate
  while (!Serial)
    delay(10);

  Serial.println("Adafruit ICM20948 test!");
  if (!icm1.begin_I2C1() || !icm2.begin_I2C2()) {
    Serial.println("Failed to find ICM20948 chip");
    while (1) {
      delay(10);
```

```
17      }
18    }
19    Serial.println("ICM20948 Found!");
20
21    //Sensors configuration
22    icm1.setAccelRange(ICM20948_ACCEL_RANGE_8_G);
23    icm2.setAccelRange(ICM20948_ACCEL_RANGE_8_G);
24    icm1.setGyroRange(ICM20948_GYRO_RANGE_1000_DPS);
25    icm2.setGyroRange(ICM20948_GYRO_RANGE_1000_DPS);
26
27 }
28
29 void loop() {
30    sensors_event_t accel1, accel2;
31    sensors_event_t gyro1, gyro2;
32    sensors_event_t mag1, mag2;
33    sensors_event_t temp1, temp2;
34    icm1.getEvent(&accel1, &gyro1, &temp1, &mag1);  ...
        //Thigh sensor reading
35    icm2.getEvent(&accel2, &gyro2, &temp2, &mag2);  ...
        //Shin sensor reading
36
37    //Roll
38    float phi_1 = atan2(accel1.acceleration.y, ...
39     sqrt(pow(accel1.acceleration.x,2)+ ...
40     pow(accel1.acceleration.z,2)));
41    float phi_2 = atan2(accel2.acceleration.y, ...
42     sqrt(pow(accel2.acceleration.x,2)+ ...
43     pow(accel2.acceleration.z,2)));
44    //Yaw
45    float theta_1 = ...
        atan2(sqrt(pow(accel1.acceleration.x,2)+ ...
46     pow(accel1.acceleration.y,2)),accel1.acceleration.z);
47    float theta_2 = ...
        atan2(sqrt(pow(accel2.acceleration.x,2)+ ...
48     pow(accel2.acceleration.y,2)),accel2.acceleration.z);
```

```
49
50   //Evaluation of knee flexion
51   float flexion_roll = phi_2-phi_1;
52   float flexion_yaw = theta_2-theta_1;
53
54   Serial.print("\t\tFlex roll: ");
55   Serial.print(flexion_roll);
56   Serial.print(" \tFlex yaw: ");
57   Serial.print(flexion_yaw);
58   Serial.println("");
59 }
```

Listing C.1: Arduino program to compute knee flexion and extension with digital system

Appendix D

Arduino algorithm for Kalman filter and complementary filter

This section provides the Arduino code used to compute the Kalman filter and the complementary filter, as shown in Section 5.4.1. The libraries were taken from [3] and modified according to our requirements

```
#include <Adafruit_ICM20X.h>
#include <Adafruit_ICM20948.h>
#include <Adafruit_Sensor.h>
#include <Wire.h>
#include <Kalman.h>

Adafruit_ICM20948 icm1, icm2;
sensors_event_t accel1, accel2;
sensors_event_t gyro1, gyro2;
sensors_event_t temp1, temp2;
sensors_event_t mag1, mag2;
Kalman kalmanX1, kalmanX2, kalmanY1, kalmanY2, ...
    kalmanZ1, kalmanZ2;

double accX1, accX2, accY1, accY2, accZ1, accZ2;
double gyroX1, gyroX2, gyroY1, gyroY2, gyroZ1, gyroZ2;
int16_t tempRaw1, tempRaw2;
double gyroXangle1, gyroXangle2, gyroYangle1, ...
    gyroYangle2, gyroZangle1, gyroZangle2;
```

```
18 double compAngleX1, compAngleX2, compAngleY1, ...
     compAngleY2, compAngleZ1, compAngleZ2;
19 double kalAngleX1, kalAngleX2, kalAngleY1, ...
     kalAngleY2, kalAngleZ1, kalAngleZ2;
20 uint32_t timer;
21
22 void setup() {
23   Serial.begin(115200);
24   while (!Serial)
25     delay(10);
26   if (!icm1.begin_I2C1() || !icm2.begin_I2C2()) {
27     Serial.println("Failed to find ICM20948 chip");
28     while (1) {
29       delay(10);
30     }
31   }
32   Serial.println("ICM20948 Found!");
33
34   // Set kalman and gyro starting angle
35   icm1.setAccelRange(ICM20948_ACCEL_RANGE_8_G);
36   icm1.setAccelRange(ICM20948_ACCEL_RANGE_8_G);
37   icm2.setGyroRange(ICM20948_GYRO_RANGE_1000_DPS);
38   icm2.setGyroRange(ICM20948_GYRO_RANGE_1000_DPS);
39   icm1.getEvent(&accel1, &gyro1, &temp1, &mag1);
40   icm2.getEvent(&accel2, &gyro2, &temp2, &mag2);
41   accX1 = accel1.acceleration.x;
42   accX2 = accel2.acceleration.x;
43   accY1 = accel1.acceleration.y;
44   accY2 = accel2.acceleration.y;
45   accZ1 = accel1.acceleration.z;
46   accZ2 = accel2.acceleration.z;
47
48   float pitch_1 = atan2(pow(accX1,2), ...
       sqrt(pow(accY1,2) + pow(accZ1,2))) * RAD_TO_DEG;
49   float roll_1 = ...
       atan2(pow(accY1,2),sqrt(pow(accX1,2) + ...
```

```
      pow(accZ1,2))) * RAD_TO_DEG;
50  float yaw_1 = atan2(sqrt(pow(accX1,2) + ...
      pow(accY1,2)), pow(accZ1,2)) * RAD_TO_DEG;
51  float pitch_2 = atan2(pow(accX2,2), ...
      sqrt(pow(accY2,2) + pow(accZ2,2))) * RAD_TO_DEG;
52  float roll_2 = atan2(pow(accY2,2), ...
      sqrt(pow(accX2,2) + pow(accZ2,2))) * RAD_TO_DEG;
53  float yaw_2 = atan2(sqrt(pow(accX2,2) + ...
      pow(accY2,2)), pow(accZ2,2)) * RAD_TO_DEG;
54
55  kalmanX1.setAngle(pitch_1); // Set starting angle
56  kalmanY1.setAngle(roll_1);
57  kalmanZ1.setAngle(yaw_1);
58  gyroXangle1 = pitch_1;
59  gyroYangle1 = roll_1;
60  gyroZangle1 = yaw_1;
61  compAngleX1 = pitch_1;
62  compAngleY1 = roll_1;
63  compAngleZ1 = yaw_1;
64
65  kalmanX2.setAngle(pitch_2);
66  kalmanY2.setAngle(roll_2);
67  kalmanZ2.setAngle(yaw_2);
68  gyroXangle2 = pitch_2;
69  gyroYangle2 = roll_2;
70  gyroZangle2 = yaw_2;
71  compAngleX2 = pitch_2;
72  compAngleY2 = roll_2;
73  compAngleZ2 = yaw_2;
74
75  timer = micros();
76 }
77
78 void loop() {
79  // Update all the values
80  icm1.getEvent(&accel1, &gyro1, &temp1, &mag1);
```

```
81  icm2.getEvent(&accel2, &gyro2, &temp2, &mag2);
82
83  accX1 = accel1.acceleration.x;
84  accY1 = accel1.acceleration.y;
85  accZ1 = accel1.acceleration.z;
86  gyroX1 = gyro1.gyro.x;
87  gyroY1 = gyro1.gyro.y;
88  gyroZ1 = gyro1.gyro.z;
89  accX2 = accel2.acceleration.x;
90  accY2 = accel2.acceleration.y;
91  accZ2 = accel2.acceleration.z;
92  gyroX2 = gyro2.gyro.x;
93  gyroY2 = gyro2.gyro.y;
94  gyroZ2 = gyro2.gyro.z;
95
96  double dt = (double)(micros() - timer) / 1000000; ...
       // Calculate Δ time
97  timer = micros();
98
99  float pitch_1 = atan2(pow(accX1,2), ...
       sqrt(pow(accY1,2) + pow(accZ1,2))) * RAD_TO_DEG;
100 float roll_1 = ...
       atan2(pow(accY1,2),sqrt(pow(accX1,2) + ...
       pow(accZ1,2))) * RAD_TO_DEG;
101 float yaw_1 = atan2(sqrt(pow(accX1,2) + ...
       pow(accY1,2)), pow(accZ1,2)) * RAD_TO_DEG;
102 float pitch_2 = atan2(pow(accX2,2), ...
       sqrt(pow(accY2,2) + pow(accZ2,2))) * RAD_TO_DEG;
103 float roll_2 = atan2(pow(accY2,2), ...
       sqrt(pow(accX2,2) + pow(accZ2,2))) * RAD_TO_DEG;
104 float yaw_2 = atan2(sqrt(pow(accX2,2) + ...
       pow(accY2,2)), pow(accZ2,2)) * RAD_TO_DEG;
105
106 double gyroXrate1 = gyroX1;
107 double gyroYrate1 = gyroY1;
108 double gyroZrate1 = gyroZ1;
```

```
109   double gyroXrate2 = gyroX2;
110   double gyroYrate2 = gyroY2;
111   double gyroZrate2 = gyroZ2;
112
113   // This fixes the transition problem when the ...
         accelerometer angle jumps between -180 and 180 ...
         degrees
114   if ((roll_1 < -90 && kalAngleY1 > 90) || (roll_1 ...
         > 90 && kalAngleY1 < -90)) {
115     kalmanY1.setAngle(roll_1);
116     compAngleY1 = roll_1;
117     kalAngleY1 = roll_1;
118     gyroYangle1 = roll_1;
119   } else
120     kalAngleY1 = kalmanY1.getAngle(roll_1, ...
           gyroYrate1, dt);
121     kalAngleZ1 = kalmanZ1.getAngle(yaw_1, ...
           gyroZrate1, dt);
122
123   if (abs(kalAngleY1) > 90)
124     gyroXrate1 = -gyroXrate1;
125   kalAngleX1 = kalmanX1.getAngle(pitch_1, ...
         gyroXrate1, dt);
126   kalAngleZ1 = kalmanZ1.getAngle(yaw_1, gyroZrate1, ...
         dt);
127
128   if ((roll_2 < -90 && kalAngleY2 > 90) || (roll_2 ...
         > 90 && kalAngleY2 < -90)) {
129     kalmanY2.setAngle(roll_2);
130     compAngleY2 = roll_2;
131     kalAngleY2 = roll_2;
132     gyroYangle2 = roll_2;
133   } else
134     kalAngleY2 = kalmanY2.getAngle(roll_2, ...
           gyroYrate2, dt);
135     kalAngleZ2 = kalmanZ2.getAngle(yaw_2, ...
```

```
        gyroZrate2, dt);
136
137   if (abs(kalAngleY2) > 90)
138     gyroXrate2 = -gyroXrate2;
139   kalAngleX2 = kalmanX2.getAngle(pitch_2, ...
        gyroXrate2, dt);
140   kalAngleZ2 = kalmanZ2.getAngle(yaw_2, gyroZrate2, ...
        dt);
141
142   // Calculate gyro angle without any filter
143   gyroXangle1 += gyroXrate1 * dt;
144   gyroYangle1 += gyroYrate1 * dt;
145   gyroZangle1 += gyroZrate1 * dt;
146   gyroXangle2 += gyroXrate2 * dt;
147   gyroYangle2 += gyroYrate2 * dt;
148   gyroZangle2 += gyroZrate2 * dt;
149
150   // Calculate the angle using a Complimentary filter
151   compAngleX1 = 0.93 * (compAngleX1 + gyroXrate1 * ...
        dt) + 0.07 * pitch_1;
152   compAngleY1 = 0.93 * (compAngleY1 + gyroYrate1 * ...
        dt) + 0.07 * roll_1;
153   compAngleZ1 = 0.93 * (compAngleZ1 + gyroZrate1 * ...
        dt) + 0.07 * yaw_1;
154   compAngleX2 = 0.93 * (compAngleX2 + gyroXrate2 * ...
        dt) + 0.07 * pitch_2;
155   compAngleY2 = 0.93 * (compAngleY2 + gyroYrate2 * ...
        dt) + 0.07 * roll_2;
156   compAngleZ2 = 0.93 * (compAngleZ2 + gyroZrate2 * ...
        dt) + 0.07 * yaw_2;
157
158   // Reset the gyro angle when it has drifted too much
159   if (gyroXangle1 < -180 || gyroXangle1 > 180)
160     gyroXangle1 = kalAngleX1;
161   if (gyroYangle1 < -180 || gyroYangle1 > 180)
162     gyroYangle1 = kalAngleY1;
```

```
163   if (gyroZangle1 < -180 || gyroZangle1 > 180)
164     gyroZangle1 = kalAngleZ1;
165   if (gyroXangle2 < -180 || gyroXangle2 > 180)
166     gyroXangle2 = kalAngleX2;
167   if (gyroYangle2 < -180 || gyroYangle2 > 180)
168     gyroYangle2 = kalAngleY2;
169   if (gyroZangle2 < -180 || gyroZangle2 > 180)
170     gyroZangle2 = kalAngleZ2;
171
172   double flexion = pitch_2-pitch_1;
173   double rotation = roll_2-roll_1;
174   double adduction = yaw_2-yaw_1;
175
176   double flexion_kalman = kalAngleX2 - kalAngleX1;
177   double rotation_kalman = kalAngleY2 - kalAngleY1;
178   double adduction_kalman = kalAngleZ2 - kalAngleZ1;
179
180   double flexion_compl = compAngleX2-compAngleX1;
181   double rotation_compl = compAngleY2 -compAngleY1;
182   double adduction_compl = compAngleZ2 - compAngleZ1;
183
184   Serial.print("\t\tRoll: "); ...
        Serial.print(flexion); Serial.print("\t");
185   Serial.print("\t\tComlp_X: "); ...
        Serial.print(flexion_compl); Serial.print("\t");
186   Serial.print("\t\tKalm_X: "); ...
        Serial.println(flexion_kalman);
187
188   Serial.print("\t\tPitch: "); ...
        Serial.print(rotation); Serial.print("\t");
189   Serial.print("\t\tComlp Y: "); ...
        Serial.print(rotation_compl); Serial.print("\t");
190   Serial.print("\t\tKalm Y: "); ...
        Serial.println(rotation_kalman);
191
192   Serial.print("\t\tYaw: "); ...
```

```
        Serial.print(adduction); Serial.print("\t");
193   Serial.print("\t\tComlp Z: "); ...
        Serial.print(adduction_compl); Serial.print("\t");
194   Serial.print("\t\tKalm Z: "); ...
        Serial.println(adduction_kalman);
195
196   Serial.print("\r\n");
197   delay(50);
198 }
```

Listing D.1: Algorithm that exploits Kalman filter and complementary filter for sensor fusion

Appendix E

Arduino algorithm for quaternion

This section provides the Arduino code used to compute the quaternion and the relative position between the two sensors, as shown in Section 5.4.2. The libraries were taken from [3] and modified according to our requirements

```
#include "ICM_20948.h"
#define SERIAL_PORT Serial
#define WIRE_PORT Wire
#define AD0_VAL0 0
#define AD0_VAL1 1

ICM_20948_I2C myICM_0;
ICM_20948_I2C myICM_1;

void setup()
{

  SERIAL_PORT.begin(115200);
  delay(100);

  while (SERIAL_PORT.available())
    SERIAL_PORT.read();

  SERIAL_PORT.println(F("Press any key to ...
     continue..."));

```

```
21  while (!SERIAL_PORT.available())
22    ;
23  WIRE_PORT.begin();
24  WIRE_PORT.setClock(400000);
25
26  bool initialized = false;
27  while (!initialized)
28  {
29    myICM_0.begin(WIRE_PORT, AD0_VAL0);
30    myICM_1.begin(WIRE_PORT, AD0_VAL1);
31    SERIAL_PORT.println(myICM_0.statusString());
32    SERIAL_PORT.println(myICM_1.statusString());
33    if (myICM_0.status != ICM_20948_Stat_Ok || ...
         myICM_1.status != ICM_20948_Stat_Ok)
34    {
35      delay(500);
36    }
37    else
38    {
39      initialized = true;
40    }
41  }
42
43  bool success_0 = true;
44  success_0 &= (myICM_0.initializeDMP() == ...
       ICM_20948_Stat_Ok);
45  success_0 &= ...
       (myICM_0.enableDMPSensor(INV_ICM20948_SENSOR_...
46  GAME_ROTATION_VECTOR) == ICM_20948_Stat_Ok);
47  success_0 &= ...
       (myICM_0.setDMPODRrate(DMP_ODR_Reg_Quat6, 0) ...
       == ICM_20948_Stat_Ok);
48  success_0 &= (myICM_0.enableFIFO() == ...
       ICM_20948_Stat_Ok); // Enable the FIFO
49  success_0 &= (myICM_0.enableDMP() == ...
       ICM_20948_Stat_Ok);  // Enable the DMP
```

```
50   success_0 &= (myICM_0.resetDMP() == ...
       ICM_20948_Stat_Ok);    // Reset DMP
51   success_0 &= (myICM_0.resetFIFO() == ...
       ICM_20948_Stat_Ok);    // Reset FIFO
52
53   bool success_1 = true;
54   success_1 &= (myICM_1.initializeDMP() == ...
       ICM_20948_Stat_Ok);
55   success_1 &= ...
       (myICM_1.enableDMPSensor(INV_ICM20948_SENSOR_...
56   GAME_ROTATION_VECTOR) == ICM_20948_Stat_Ok);
57   success_1 &= ...
       (myICM_1.setDMPODRrate(DMP_ODR_Reg_Quat6, 0) ...
       == ICM_20948_Stat_Ok);
58   success_1 &= (myICM_1.enableFIFO() == ...
       ICM_20948_Stat_Ok); // Enable the FIFO
59   success_1 &= (myICM_1.enableDMP() == ...
       ICM_20948_Stat_Ok);  // Enable the DMP
60   success_1 &= (myICM_1.resetDMP() == ...
       ICM_20948_Stat_Ok);    // Reset DMP
61   success_1 &= (myICM_1.resetFIFO() == ...
       ICM_20948_Stat_Ok);    // Reset FIFO
62   SERIAL_PORT.println(F("DMP enabled!"));
63 }
64
65 void loop()
66 {
67   icm_20948_DMP_data_t data_0;
68   icm_20948_DMP_data_t data_1;
69   myICM_0.readDMPdataFromFIFO(&data_0);
70   myICM_1.readDMPdataFromFIFO(&data_1);
71   if (((myICM_0.status == ICM_20948_Stat_Ok) || ...
       (myICM_0.status == ...
       ICM_20948_Stat_FIFOMoreDataAvail)) && ...
       ((myICM_1.status == ICM_20948_Stat_Ok) || ...
       (myICM_1.status == ...
```

```
       ICM_20948_Stat_FIFOMoreDataAvail)))
72   {
73     if ((data_0.header & DMP_header_bitmap_Quat6) > ...
          0 & (data_1.header & ...
          DMP_header_bitmap_Quat6) > 0)
74     {
75       double q1_0 = ((double)data_0.Quat6.Data.Q1) ...
           / 1073741824.0;
76       double q2_0 = ((double)data_0.Quat6.Data.Q2) ...
           / 1073741824.0;
77       double q3_0 = ((double)data_0.Quat6.Data.Q3) ...
           / 1073741824.0;
78       double q1_1 = ((double)data_1.Quat6.Data.Q1) ...
           / 1073741824.0;
79       double q2_1 = ((double)data_1.Quat6.Data.Q2) ...
           / 1073741824.0;
80       double q3_1 = ((double)data_1.Quat6.Data.Q3) ...
           / 1073741824.0;
81       double q0_0 = sqrt(1.0 - ((q1_0 * q1_0) + ...
           (q2_0 * q2_0) + (q3_0 * q3_0)));
82       double q0_1 = sqrt(1.0 - ((q1_1 * q1_1) + ...
           (q2_1 * q2_1) + (q3_1 * q3_1)));
83       double q1_2 = q1_1 - q1_0;
84       double q2_2 = q2_1 - q2_0;
85       double q3_2 = q3_1 - q3_0;
86       double q0_2 = sqrt(1.0 - ((q1_2* q1_2) + ...
           (q2_2 * q2_2) + (q3_2 * q3_2)));
87       double q2sqr_2 = q2_2 * q2_2;
88
89       // pitch (difference between q1 and q2 on ...
           x-axis)
90       double t0_2 = +2.0 * (q0_2 * q1_2 + q2_2 * ...
           q3_2);
91       double t1_2 = +1.0 - 2.0 * (q1_2 * q1_2 + ...
           q2sqr_2);
92       double pitch = atan2(t0_2, t1_2) * 180.0 / PI;
```

```
93
94        // roll (difference between q1 and q2 on y-axis)
95        double t2_2 = +2.0 * (q0_2 * q2_2 - q3_2 * ...
              q1_2);
96        t2_2 = t2_2 > 1.0 ? 1.0 : t2_2;
97        t2_2 = t2_2 < -1.0 ? -1.0 : t2_2;
98        double roll = asin(t2_2) * 180.0 / PI;
99
100       // yaw (difference between q1 and q2 on z-axis)
101       double t3_2 = +2.0 * (q0_2 * q3_2 + q1_2 * ...
              q2_2);
102       double t4_2 = +1.0 - 2.0 * (q2sqr_2 + q3_2 * ...
              q3_2);
103       double yaw = atan2(t3_2, t4_2) * 180.0 / PI;
104
105       SERIAL_PORT.print(F("Pitch: ")); ...
              SERIAL_PORT.print(pitch, 1);
106       SERIAL_PORT.print(F("Roll: "));  ...
              SERIAL_PORT.print(roll, 1);
107       SERIAL_PORT.print(F("Yaw: "));   ...
              SERIAL_PORT.println(yaw, 1);
108     }
109   }
110   if (myICM_0.status != ...
          ICM_20948_Stat_FIFOMoreDataAvail)
111   {
112     delay(10);
113   }
114 }
```

Listing E.1: Arduino program that exploits quaternions for sensor fusion

References

[1] Adafruit. *Adafruit STEMMA QT/Qwiic JST SH 4-pin Premium Male Headers Cable - 150mm Long*, 2020.

[2] Adafruit. *Adafruit TDK InvenSense ICM-20948 9-DoF IMU*, 2020.

[3] Adafruit. Adafruit libraries icm20948. `https://github.com/adafruit/Adafruit_ICM20X.git`, 2021. [Online; accessed].

[4] Analog Devices. *3-Axis ±3 g, Accelerometer ADXL335*, 2009. Rev. B.

[5] Arduino. *Arduino MEGA 2560 Rev3*, 2021. Rev. 3.

[6] S. Bakhshi, M. H. Mahoor, and B. S. Davidson. Development of a body joint angle measurement system using imu sensors. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 6923–6926. IEEE, 2011.

[7] A. Bally, M. Boreiko, F. Bonjour, and C. Brown. Modeling forces on the anterior cruciate knee ligament during backward falls while skiing. *Skiing Trauma and Safety: Seventh International Symposium*, pages 267–276, 1989.

[8] T. Bere, T. W. Flørenes, T. Krosshaug, H. Koga, L. Nordsletten, C. Irving, E. Muller, R. C. Reid, V. Senner, and R. Bahr. Mechanisms of anterior cruciate ligament injury in world cup alpine skiing: A systematic video analysis of 20 cases. *The American Journal of Sports Medicine*, 39(7):1421–1429, 2011.

[9] T. Bere, K.-M. Mok, H. Koga, T. Krosshaug, L. Nordsletten, and R. Bahr. Kinematics of anterior cruciate ligament ruptures in world cup alpine skiing: 2 case reports of the slip-catch mechanism. *The American Journal of Sports Medicine*, 41(5):1067–1073, 2013.

[10] G. Bravo-Illanes, R. T. Halvorson, R. P. Matthew, D. Lansdown, C. B. Ma, and R. Bajcsy. Imu sensor fusion algorithm for monitoring knee kinematics in acl reconstructed patients. In *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 5877–5881. IEEE, 2019.

[11] A. Cenedese and R. Antonello. Networked control for multiagent systems lecture notes, 2020.

[12] G. Cooper, I. Sheret, L. McMillian, K. Siliverdis, N. Sha, D. Hodgins, L. Kenney, and D. Howard. Inertial sensor-based knee flexion/extension angle estimation. *Journal of biomechanics*, 42(16):2678–2685, 2009.

[13] J. Favre, F. Luthi, B. Jolles, O. Siegrist, B. Najafi, and K. Aminian. A new ambulatory system for comparative evaluation of the three-dimensional knee kinematics, applied to anterior cruciate ligament injuries. *Knee Surgery, Sports Traumatology, Arthroscopy*, 14(7):592–604, 2006.

[14] J. Fischer, O. Loureiro, P. Leyvraz, and A. Bally. Injury mechanism of the anterior cruciate ligament in alpine skiing: Case studies. *Skiing Trauma and Safety*, 10:270–279, 1996.

[15] S. L. Hame, D. A. Oakes, and K. L. Markolf. Injury to the anterior cruciate ligament during alpine skiing: A biomechanical analysis of tibial torque and knee flexion angle. *The American Journal of Sports Medicine*, 30(4):537–540, 2002.

[16] A. Hermann, J. Ostarhild, Y. Mirabito, N. Bauer, and V. Senner. Stretchable piezoresistive vs. capacitive silicon sensors integrated into ski base layer pants for measuring the knee flexion angle. *Sports Engineering*, 23(1):1–10, 2020.

[17] A. Hermann and V. Senner. Knee injury prevention in alpine skiing. a technological paradigm shift towards a mechatronic ski binding. *Journal of Science and Medicine in Sport*, 24(10):1038–1043, 2021.

[18] InvenSense. *TDK ICM-20948*, 2020.

[19] M. J. Jordan, P. Aagaard, and W. Herzog. Anterior cruciate ligament injury/reinjury in alpine ski racing: a narrative review. *Open access journal of sports medicine*, 8:71–83, 2017.

[20] M. Järvinen, A. Natri, and S. Laurila. Mechanisms of anterior cruciate ligament ruptures in skiing. *Knee Surg, Sports traumatol*, 2:224–228, 1994.

[21] M. Klous, E. Müller, and H. Schwameder. Three-dimensional lower extremity joint loading in a carved ski and snowboard turn: A pilot study. *Computational and Mathematical Methods in Medicine*, 2006.

[22] R. S. McGinnis, S. Patel, I. Silva, N. Mahadevan, S. DiCristofaro, E. Jortberg, M. Ceruolo, and A. Aranyosi. Skin mounted accelerometer system for measuring knee range of motion. In *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 5298–5302. IEEE, 2016.

[23] J. Moon, H. Kim, J. Lee, and S. B. Panday. Effect of wearing a knee brace or sleeve on the knee joint and anterior cruciate ligament force during drop jumps: A clinical intervention study. *The Knee*, 25(6):1009–1015, 2018.

[24] National Instrument. *NI DAQ USB-6211 Multifunction I/O Device*, 2017. Rev. C.

[25] R. Negrin, B. Uribe-Echevarria, and N. Reyes. Do knee braces prevent ski knee injuries? *Asian Journal of Sports Medicine*, 8(4), 2017.

[26] D. A. Neumann. *Kinesiology of the musculoskeletal system: Foundations for rehabilitation*. 3 edition, 2016.

[27] E. Papi, Y. N. Bo, and A. H. McGregor. A flexible wearable sensor for knee flexion assessment during gait. *Gait and posture*, 62:480–483, 2018.

[28] G. Picci. *Filtraggio statistico (Wiener, Levinson, Kalman) e applicazioni*. Progetto Libreria, 2007.

[29] M. Posch, A. Schranz, and M. Lener. In recreational alpine skiing, the acl is predominantly injured in all knee injuries needing hospitalisation. *Knee Surg Sports Traumatol Arthrosc*, 29:1790–1796, 2021.

[30] B. Ren, J. Liu, and J. Chen. Simulating human–machine coupled model for gait trajectory optimization of the lower limb exoskeleton system based on genetic algorithm. *International Journal of Advanced Robotic Systems*, 17(1), 2020.

[31] T. Seel, J. Raisch, and T. Schauer. Imu-based joint angle measurement for gait analysis. *Sensors*, 14(4):6891–6909, 2014.

[32] V. Senner, F. I. Michel, and S. Lehner. *Ski equipment-related measures to reduced knee injuries: Review of the potential for further technical improvements in recreational alpine skiing.* bfu knowledge base. bfu - Swiss Council for Accident Prevention, Berne, 2013.

[33] J. Shealy, R. Johnson, C. Ettlinger, I. Scher, and R. Greenwal. Role of helmets in mitigation of head injuries: Epidemiologic study of head injuries to skiers. *Skiing Trauma and Safety*, 20:1–15, 2015.

[34] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo. Robotics: modelling, planning and control, 2010.

[35] W. I. Sterett, K. K. Briggs, T. Farley, and J. R. Steadman. Effect of functional bracing on knee injury in skiers with anterior cruciate ligament reconstruction: a prospective cohort study. *The American journal of sports medicine*, 34(10):1581–1585, 2006.

[36] M. C. Tarka, A. Davey, G. C. Lonza, C. M. O'Brien, J. P. Delaney, and N. K. Endres. Alpine ski racing injuries. *Sports Health*, 11(3):265–271, 2019.

[37] Unknown. I traumatismi del ginocchio. `http://web.tiscali.it/riabilitaziosportiva/immagini/gin_traumatismi.html`, 2010. [Online; accessed].

[38] L. S. Vargas-Valencia, F. B. Schneider, A. G. Leal-Junior, P. Caicedo-Rodriguez, W. A. Sierra-Arevalo, L. E. Rodriguez-Cheu, T. Bastos-Filho, and A. Frizera-Neto. Sleeve for knee angle monitoring: An imu-pof sensor fusion system. *IEEE journal of biomedical and health informatics*, 25(2):465–474, 2020.

[39] T. Walter and J. Higgings. A comparison of complementary and kalman filtering. *IEEE Trans. Aerospace and Electronic Systems*, 11(3), 1975.

[40] Wikipedia. Anatomical terms of motion 1. `https://en.wikipedia.org/wiki/Anatomical_terms_of_motion#/media/File:FlexsionExtension2017.jpeg`, 2013. [Online; accessed].

[41] Wikipedia. Anatomical terms of motion 2. https://en.wikipedia.org/wiki/Anatomical_terms_of_motion#/media/File:Body_Movements_I_(cropped_rotation).jpg, 2013. [Online; accessed].

[42] Wikipedia. Left knee joint from behind, showing interior ligaments. https://en.wikipedia.org/wiki/Anterior_cruciate_ligament#/media/File:Gray348.png, Before 1858. [Online; accessed].

[43] Wikipedia. Lower extremity of right femur viewed from below. https://en.wikipedia.org/wiki/Femur#/media/File:Gray246.png, Before 1858. [Online; accessed].

[44] Wikipedia. Right knee joint, from the front, showing interior ligaments. https://en.wikipedia.org/wiki/Anterior_cruciate_ligament#/media/File:Gray347.png, Before 1858. [Online; accessed].

[45] M. Zorko, B. Nemec, and J. Babič. The waist width of skis influences the kinematics of the knee joint in alpine skiing. *Journal of sports science and medicine*, 14(3):606–619, 2015.