Universidade do Minho

Escola de Engenharia

Ana Rita Simões Pereira

First Advances in Near Fall Detection and Prediction when using a Walker

Dissertação de Mestrado

Mestrado Integrado em Engenharia Biomédica

Ramo Eletrónica Médica

Trabalho realizado sob a orientação de

Professora Doutora Cristina P. Santos,

Universidade do Minho

Abril de 2020

DIREITOS DE AUTOR E CONDIÇÕES DE UTILIZAÇÃO DO TRABALHO POR TERCEIROS

STATEMENT OF INTEGRITY

I further declare that I have fully acknowledged the Code of Ethical Conduct of the University of Minho.

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

RESUMO

As quedas representam uma grande preocupação para a sociedade. Várias lesões associadas às quedas necessitam de cuidados médicos e, no pior dos casos, uma queda pode levar à morte. Estas consequências traduzem-se em custos elevados para a população. A fim de ultrapassar estes problemas, várias abordagens têm sido endereçadas para deteção, previsão e prevenção das quedas.

Os andarilhos são muitas vezes prescritos a sujeitos que apresentam um risco de queda maior. Desta forma, é essencial desenvolver estratégias para aumentar a segurança do utilizador perante uma situação de perigo iminente. Neste sentido, esta dissertação visa desenvolver uma estratégia que permita a deteção de uma quase queda (NF) e a sua direção, assim como a deteção incipiente de uma NF (INF). Para além disso, tem o objetivo de detetar dois eventos de marcha, o *heel strike* (HS) e o *toe-off* (TO).

As estratégias definidas, neste trabalho, basearam-se nos dados recolhidos através de um sensor inercial posicionado no tronco inferior e de sensores de força colocados nas palmilhas. Após a aquisição dos dados, a metodologia adotada para identificar as situações anteriormente referidas foi baseada em algoritmos de *machine learning*. Com o intuito de obter o modelo com melhor desempenho, várias combinações de diferentes classificadores foram testadas com três métodos de seleção de *features*.

No que concerne à deteção da NF, os resultados alcançados apresentaram um *Matthews Correlation Coefficient* (MCC) de 79.99% sendo possível detetar uma NF 1.76±0.76s antes do seu final. Com a implementação do algoritmo de pós-processamento, grande parte dos falsos positivos foram eliminados, sendo possível detetar todas as NF 1.48±0.68s antes do seu final. Em relação aos modelos construídos para distinguir a direção da NF, o melhor modelo apresentou uma precisão (ACC) de 59.97%.

A metodologia seguida neste trabalho não foi bem sucedida na deteção INF. O melhor modelo apresentou um MCC=23.87%.

Relativamente à deteção dos eventos, HS e TO, o melhor modelo atingiu um MCC=86.94%. Com a aplicação do algoritmo de pós-processamento parte das amostras mal classificadas foram eliminadas, no entanto, foi introduzido um atraso na deteção do HS e do TO. Com o pós-processamento foi possível obter um MCC=88.82%, não incluindo o atraso imposto pelo pós-processamento.

PALAVRAS-CHAVE

Andarilho, Eventos da Marcha, *Machine Learning*, Quase queda, Quase Queda Incipiente, Seleção de *features*

ABSTRACT

Falls are a major concern to society. Several injuries associated with falls need medical care, and in the worst-case scenario, a fall can lead to death. These consequences have a high cost for the population. In order to overcome these problems, a diversity of approaches for detection, prediction, and prevention of falls have been tackled.

Walkers are often prescribed to subjects who present a higher risk of falling. Thus, it is essential to develop strategies to enhance the user's safety in an imminent danger situation. In this sense, this dissertation aims to develop a strategy to detect a near fall (NF) and its direction as well as the detection of incipient near fall (INF) while the subject uses a walker. Furthermore, it has the purpose of detecting two gait events, the heel strike (HS) and the toe-off (TO).

The strategies established, in this work, were based on the data gathered through an inertial sensor placed on the lower trunk and force sensors placed on the insoles. Following data collection, the methodology adopted to identify the situations aforementioned was based on machine learning algorithms. In order to reach the model with best performance, many combinations of different classifiers were tested with three feature selection methods.

Regarding the detection of NF, the results achieved presented a Matthews Correlation Coefficient (MCC) of 79.99% being possible to detect a NF 1.76±0.76s before its end. With the implementation of the post-processing algorithm, a large part of the false positives was eliminated being able to detect all NF 1.48±0.68s before its end. Concerning the models built to distinguish the direction of the NF, the best model presented accuracy of 58.97% being unable to reliably distinguish the three fall directions.

The methodology followed, in this work, was unsuccessful to detect an INF. The best model presented MCC=23.87%, in this case.

Lastly, with respect to the detection of HS and TO events the best model reached MCC=86.94%. With the application of the post-processing algorithm, part of misclassified samples was eliminated, however, a delay in the detection of the HS and TO events was introduced. With the post-processing it was possible to reach MCC=88.82%, not including the imposed delay.

KEYWORDS

Feature Selection, Gait events, Incipient Near Fall, Machine Learning, Near Fall, Walker

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

ACC	Accuracy
ASBGo SW	Adaptive System Behaviour Group Smart Walker
Bi- LSTM	Bidirectional Long Short-Term Memory
BiRD Lab	Biomedical Robotic Devices Laboratory
COM	Center Of Mass
COP	Center Of Pressure
DT	Decision Tree
FN	False Negative
FP	False Positive
FSR	Force Sensor Resistor
FTS	Force/Torque Sensors
GA	Gait Analysis
HMM	Hidden Markov model
HS	Heel Strike
IMU	Inertial Measurement Unit
INF	Incipient Near Fall
KNN	k- Nearest Neighbours
LDA	Linear Discriminant Analysis
LRF	Laser Range Finder
LSTM	Long Short-Term Memory
MCC	Mathew Correlation Coefficient
MRMR	Max- Relevancy Min-Redundancy
NF	Near Fall

PC	Principal Components
PCA	Principal Component Analysis
PD	Parkinson Disease
PREC	Precision
QDA	Quadratic Discriminant Analysis
RMSE	Root Mean Square Error
RQ	Research Question
SENS	Sensitivity
SPEC	Specificity
SVM	Support Vector Machine
TN	True Negative
TO	Toe Off
TP	True Positive
USA	United States of America
UWB	Ultra-WideBand
WHO	World Health Organization

CHAPTER 1 – INTRODUCTION

This dissertation presents the work developed during the academic year of 2018/2019 in order to complete the fifth year of Integrated Master's Biomedical Engineering within the scope of the master's in medical electronics. This work was performed at Biomedical Robotic Devices Laboratory (BiRD Lab) of the Center of MicroEletroMechanical Systems at the University of Minho, Portugal.

The work developed in this dissertation arose through the project Adaptive System Behaviour Group Smart Walker (ASBGo SW), which has been developed on the BiRD Lab for rehabilitation and physical therapy. With this work, it is intended to enhance the user's safety, reducing the number of fall-related injuries. In this sense, the main objectives established were the detection of a Near Fall (NF) and an Incipient Near Fall (INF), and the division of the gait cycle into two phases while using a walker, i.e., an alternative system to ASBGo SW. For this purpose, an Inertial Measurement Unit (IMU) and Force Sensor Resistors (FSR) were used to record gait and lower trunk information related to the user. An approach focused on machine learning algorithms will be addressed to achieve the dissertation's main goals presented in section 1.3.

Thereby, in this dissertation the detection of a NF and an INF while using a walker are conducted with the purpose of building a prevention strategy in the future capable of changing the walker's trajectory.

1.1 Motivation

Walking is an activity performed as a means of locomotion and has a huge importance in a person's daily life [1]–[3]. This activity can be done in several ways and several directions [2], and can be an essential indicator of health [1], [3]. Over the years, human gait has been extensively studied and diverse monitoring devices have been developed. Different systems of the body are involved in human gait, namely the nervous, cardiorespiratory and musculoskeletal systems. The gait disorders may be associated with dysfunctions in the brain, neuromuscular and musculoskeletal problems. The occurrence of this problem rises with age: 10% between 60-69 years and >60% for people over 80 years. The vertiginous gait is an example of gait disorder and people who presented this type of gait have a predisposition to fall to one side. The festination gait is characteristic of patients with Parkinson's diseases (PD) and increases the risk of falling forward. Gait impairments can lead to an increase risk of falling and consequently cause injuries [4]. Many other factors can increase the likelihood of falling, such as medication, gender, diabetes, depression, Alzheimer's disease, visual problems, body mass index, sedentarism, race, and socioeconomic status [5].

Falling is actually a huge concern nowadays. They are the second main cause of death caused by accident or unintentional injury worldwide, according to the World Health Organization (WHO) [6]. About 646 000 people die each year with a large proportion of adults over 65 years old, and 37.3 million falls are severe enough to require medical attention [6].

The cost associated with non-fatal falls and fatal falls represents an economic burden to society. In 2012, in the United States of America (USA) a fatal fall costs on average $25,487 and direct costs were estimated $616.5 million were spent. Concerning non- fatal falls, in 2012, a fall cost $9463 and in total, the costs were $30.3 billion. In 2015, the costs associated with fatal falls and non-fatal falls increased to $637.2 million and $31.3 billion, respectively. In 2012, falls were more frequent in women than men in both fatal and non-fatal falls, and for both genders the occurrence of fatal falls increased with age [7].

Falls can result in severe injuries for the subject, and 31% of falls required medical assistance. Head trauma, soft tissues injuries and fractures are examples of fall related consequences [8]. In addition to physical injuries, falls are also related to anxiety, depression and fear of falling [8]. In healthcare context, there are fall risk assessment measured based on functional tests and personal information as history of fall and medication. TimeUp&Go, Berg Balance and STRATIFY are examples of assessment tools to evaluate the risk of fall [9]–[11].

There is an evident concern to decrease the costs and the injuries related to falls in order to improve the quality of life. Thus, distinct systems have been developed in order to detect a fall [12], [13] as well as the pre-impact fall [14], and to prevent it [13]. In this context, wearables and non-wearables sensors have been used to gather information from subjects. Different methodologies have been implemented, such as methodologies based on machine learning and threshold algorithms [12]–[14]. As mentioned, the impact of the user on the ground when it falls can cause physical and psychological damage. In order to mitigate the damage caused by the fall there is an interest to develop an approach that prevents the person from falling to the ground, as the work developed by [15].

Normally assistive devices, such as canes and walkers are prescribed for people who have balance problems and need help to maintain it and for locomotion. These devices can decrease the load on the lower limbs, relieve joint pain, and help people who have gait problems, such as muscle weakness, arthritis or hemiplegia. There is controversy as to whether these devices increase the risk of falling or whether it is just an indicator of risk of falling and balance problems [16]. However, we can conclude that people who use walking aids are prone to falling or due to the use of these devices or just because of their physical condition.

1.2 Problem Statement

Fall-related injuries and the costs associated with them are a major concern to society, since they can be severe requiring medical care and the associated costs are high. Therefore, the exploration of fall detection and prevention has been the focus of interest with the purpose of improving the person's quality of life [13].

Walkers are assistive devices used by individuals who have some mobility problem, and inherently have an associated risk of falling. In this sense, it is important develop a strategy that can detect a fall and, mainly, prevent it. Therefore, literature research related to the sensors and the approaches implemented to detect and prevent falls is imperative.

The detection of NF is the first step to be taken in order to develop a strategy that prevents the subject from falling to the ground in a dangerous situation when using a walker. In this way collect subject's information during the locomotion is important. Through gait and lower trunk information several features were calculated and several combinations of feature selection methods with machine learning algorithms were explored, in order to detect a NF. Detecting the direction of the NF is the second step, this information is important to develop a more focused and robust approach to avoid the NF. The time in advance to detect a dangerous situation is essential, in this sense the recognition of an INF is an interesting case of study to be explored. In this regard, a methodology similar to the detection of NF was addressed. Based on machine learning algorithms, two gait events, toe off (TO) and heel strike (HS), will be detected to extract more information about the users' gait.

All the aspects mentioned show relevance in the detection of NF and this study is a contribution to increase the knowledge in this research area.

1.3 Goals and Research Questions

The main objectives of this dissertation are to develop a strategy to distinguish a NF and an INF from normal walking, and to detect two gait events while using a walker, TO and HS. In order to achieve these objectives, it is necessary to acquire knowledge about the sensors that can be employed and their locations as well as the methodologies developed for the detection, prediction, and prevention of fall and NF events. Additionally, it is essential to understand the basic concepts of human gait.

Thereby, with this dissertation, it is necessary to achieve the following goals:

- **Goal 1:** To analyze and extract pertinent information related to falls and NF. It is important to know what are the types of fall and the risk that contribute to a fall, i.e., related to the subject – intrinsic - and related to the environment - extrinsic. It is also crucial to understand the variety of sensors and their locations on the subject's body as well as the methodologies implemented concerning fall and NF detection, prediction and prevention.
- **Goal 2:** To perform a literature search, similar to goal 1, focused on fall-related strategies implemented in smart walkers. After a state of the art, it intends to recognize the potentials, issues, challenges and future directions.
- **Goal 3:** To prepare an existing system allowing the resolution of goals 4, 5 and 6. This goal considers the outcome information from goals 1 and 2.
- **Goal 4:** To implement a strategy to distinguish the normal walking from a NF situation and, subsequently, classify the NF direction (right, left and forward). It is intended to define which is the most relevant feature set for detecting the NF and its direction as well as the most suitable machine learning algorithm.
- **Goal 5:** To distinguish the normal walking from an INF using machine learning algorithms and feature selection methods. It is intended to analyze the minimum number of features required and which is the best combination of classifier and feature selection method.
- **Goal 6:** To detect two gait events, namely, TO and HS, during normal human walking, only through the signals achieved with an IMU. The approach implemented is based on machine learning algorithm as in the goals 4 and 5.

The following Research Questions (RQ) are expected to be answered in the present work:

- **RQ1:** Which sensors have the highest potential for detecting a NF and what is the most viable strategy for the problem under study? This RQ is addressed in Chapter 4.
- **RQ2:** What is the minimum number of features necessary, and what is the best combination of feature selection method and classifier algorithm to detect a NF and its direction while using a walker? This RQ is addressed in Chapter 5.
- **RQ3:** Is it possible to detect an INF while using a walker? This RQ is addressed in Chapter 6.

- **RQ4:** What is the best combination of classifier, feature selection method, and the number of features that presented the best performance in detecting gait events while using a walker? This RQ is addressed in Chapter 7.

1.4 Contributions

The main contributions of this work are:

- A survey of existing strategies implemented in smart walkers to detect and prevent a fall.
- A NF detection strategy also capable of determining the NF direction while the subject uses a walker at different velocities.
- A study to explore the reliability of the INF detection based on different machine learning algorithms.
- An automatic method to detect the HS and TO gait events of each foot through a feature set computed from inertial signals.

1.5 Publications

From the work developed during this year, it was achievable public two conference articles.

- A. Pereira, N. F. Ribeiro and C. P. Santos, "A Survey of Fall Prevention Systems Implemented on Smart Walkers", 2019 IEEE 6th Portuguese Meeting on Bioengineering (ENBENG), Lisbon, Portugal, 22-23 February 2019.

- A. Pereira, N. F. Ribeiro and C. P. Santos, "A Preliminary Strategy for Fall Prevention in the ASBGo Smart Walker", 2019 IEEE 6th Portuguese Meeting on Bioengineering (ENBENG), Lisbon, Portugal, 22-23 February 2019.

1.6 Thesis Outline

This dissertation is organized as follows. Chapter 2 presents a state of the art addressed to the four points: i) the different types of fall; ii) the intrinsic and extrinsic fall-related risk factors; iii) the consequences of falls; and iv) the technologies approaches to detect, predict and prevent a fall or a NF. This comprises the sensors used, the sensor's location, features computed and the performance of the systems.

In chapter 3, a detailed description of ASBGo SW is presented. This chapter also presents a survey centered on different approaches implemented on smart walkers to detect or prevent a fall. A brief description of approaches, sensors used, and their locations are provided.

Chapter 4 describes the system and the methodology followed to detect a NF. A detailed explanation of the sensors used, the pre-processing methodology, the feature computation, the feature selection methods, and the different machine learning algorithms are provided. This chapter also presents the volunteers characteristics and the activities description performed during the trials while using a walker.

In chapter 5, it shows all procedures stages performed to distinguish a NF from normal walking. These stages involve the process of data labelling, the features computation, the model construction, the classifiers performances achieved and, lastly, the techniques implemented to improve the outcomes. This chapter is divided in two parts: distinguish between a NF and normal walking, and distinguish three different NF directions (forward, right and left).

Chapter 6 presents the results and the discussion of the outcomes achieved to distinguish an INF from normal walking. In chapter 7, the gait events, HS and TO of each foot are detected based on inertial signals and compared with the ground truth. In both chapters the methodology implemented was similar to the one used in chapter 5.

Chapter 8 presents the general conclusion of the work developed during this dissertation, the answers to the RQ, and what are the future challenges to be addressed.

CHAPTER 2 – STATE OF THE ART

Falls are responsible for numerous deaths worldwide being the second cause of death by accident or unintentional injury [6]. The consequences linked to fatal and non-fatal falls represent an economic burden on society [7]. Hence the fall related effects lead to a reduction in subjects' quality of life. In order to comprehend more concretely the problematic of falls, this chapter begins with a contextualization of falls, specifically: the definition of a fall and a NF; the types of fall and in which phases can be divided; the factors that contribute to increase the risk of falling; and the consequences that can result from a fall. It is followed by an overview of fall-related systems, divided into two main parts: systems that use wearable and non-wearable sensors. The chapter ends with a discussion of all the knowledge acquired throughout these sections.

2.1 Fall and Near Fall Definition

First of all, it is necessary to define and distinguish a fall from a NF. Thereby, according to the WHO a fall is "an event which results in a person coming to rest inadvertently on the ground or floor or other lower level" [6]. While a NF is, generally, defined as a result of the loss of balance that can be recovered, i.e., the person does not fall to the ground, for instance, slip, trips, and missteps [17]. Nonetheless, Maidan et. al [18] proposed a new definition of a NF. These authors consider that for a NF to happen, two of the following actions should be taken in order to compensate the loss of balance: lowering of the center of mass (COM); trunk inclination; an unexpected change in the movement of the arms and/or legs; a sudden variation in the size of the stride; and an unforeseen alteration in the speed of the stride.

2.2 Types and Phases of a Fall

The falls can occur in various ways and in different scenarios. Yu et al. [19] considered four kinds of fall based on scenarios, i.e., fall from sitting (e.g. when the subject is sitting in the chair and starts to fall), bed, walking and standing on the floor, and standing on support. On the other hand Noury et al. [20] classified the falls in three different forms, i.e., forward, backward, sideward. These falls can happen, for instance due to transfers "Stand to sit", stumble on an obstacle, backward slip and walking.

Falls can be discriminated into different phases, however there is not only one form of division. Noury et al. [21] divide the fall into four phases. The first phase, called pre-fall phase is the period the subject is executing activities of daily live comprising fast movements. The second phase designated as the critical phase corresponds the period since an abrupt movement happen until to vertical impact on the ground (300-500ms). The third phase, post-fall phase, the subject gets immobile. Lastly, the recovery phase, the faller gets up. However, Hsieh et al. [22] only considered three phases: the free-fall phase, the impact phase, and the rest phase. The free-fall phase is characterized by rapid movements; the impact phase consists of the subject shock on the ground; and lastly, the rest phase, the subject remains immobilized.

2.3 Fall Risks

It is elementary to realise which factors contribute to fall risks in order to reduce the number of falls and prevent them. Falls result due to a number of factors that can be separated into two groups: intrinsic and extrinsic [23], [24]. The intrinsic risk factors are related to the patients while the extrinsic risk factors are associated to the surrounding environment. The first group mentioned includes, e.g. visual problems, weakness in the lower extremities, and cognitive impairments among the elderly population [23]. Concerning the second group, lack of light, inappropriate shoes and slippery surfaces that lead to slip are some of the factors that increase the fall risk [23]. Furthermore, the assistive devices may also contribute to an increased risk of a fall in the aging population [23], [25]. In this section, further factors which play a crucial role in the fall risks will be introduced. As a result of this, it is intended to identify who is more prone to suffer a fall.

In the older adult population, the fall is a considerable concern. People over 80 years old have a higher risk of suffer multiple falls than older adults with 65-79 years old [24], and subjects between 75 and 85 years old experience falls more frequently [26]. Moreover, the possibility of a fall causing an injury doubled for subject with 85 years old or more and for women [26]. The fall rate and the severity level of the injury are higher for women than for men [25]–[27]. Given the fact of females have a loss of lower bone mass quicker than males, and, with age, females are more likely to have osteoporosis, which can be an explanation for a higher incidence of fall in this gender [25], [27]. In addition, women with ageing have a reduction in muscle mass which may also contribute to the risk of falling [25]. Beyond the age and the gender, the race is another risk factor. In the USA white men are more likely to fatal-fall and the rate of fatal-falls decrease for white women followed by black men, and black women [5]. Elderly people with low educational levels [5], [25], and older women with lack of social relationships have a higher fall risk [5].

The number of medications may contribute to increase the fall risk due to adverse effects and also by interaction of different medicines [25]. The intake of more than four medications considerably contribute to the fall risk [5], [25], and it is linked to the fear of falling [5]. Some of the drugs associated with falls are, e.g. antidepressants, antihypertensives, antipsychotic and sedatives [28], [29]. The drugs can provoke balance problems, mental confusion, decrease in blood pressure and urinary incontinence that rise the fall possibility [23]. As a result of the number of times that the person needs to go to the bathroom and because of loss of balance when the person goes hastily to the bathroom can increase the fall danger [25].

Physical problems, such as muscle weakness that causes changes in balance as well as cognitive problems from dementia, and foot problems (e.g. toe deformity) are considered biological risk factors [5]. Muscle weakness and gait problems are conditions linked to osteoarthritis that can contribute to an increased risk of falling in patients who suffer from this disease [25]. Subjects with neurological impairments are more prone to fall than subjects without any neurological impairments. Homann et al. [30] revealed that 50% of ambulatory neurological patients experienced at least one fall in one year. Patients with PD and who suffer stroke are five and six times, respectively, more prone to fall than control group. In Figure 1, it is possible to observe the most common neurological diseases with highest frequency of fall [30].

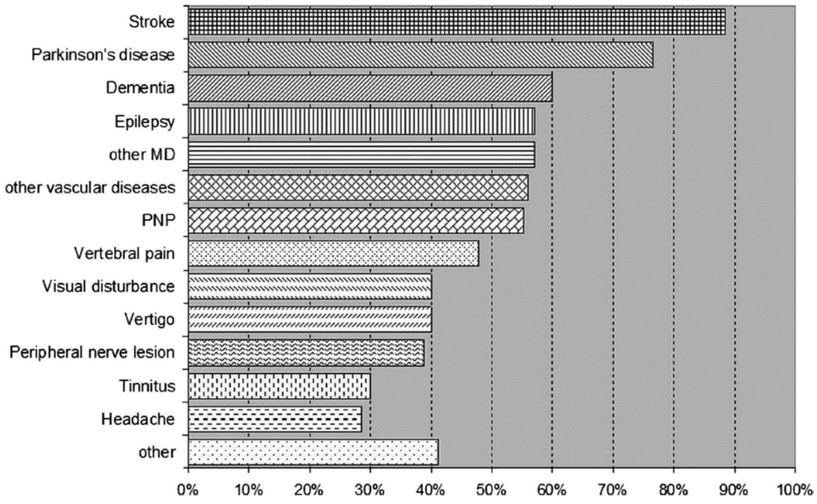


Figure 1 - Difference in frequency of having at least one fall within a period of a year for patients with neurological disorders [30].

Usually, PD patients fall forward. This type of fall is common related to wrist fractures since in a fall situation the person try to support themselves with the hand [31]. However, PD patients have more hip fractures. This can be explained by abnormal arm movement that prevents PD patients from supporting themselves with the hand. In Figure 2, it is possible to observe how PD patients and the elderly react when they fall [31].

Different conditions associated with PD contribute to an increased risk of falling [31]. The freezing of gait episodes, i.e., incapacity to step during a short period of time or short steps [32], leads to a variability in foot strike and stride length which can potentiate the fall event [31], [33]. Movements like turning, performed transfers (for instance, rising from a chair) and execute dual tasking can be difficult to PD patients, also leading to a higher risk of falling [31].

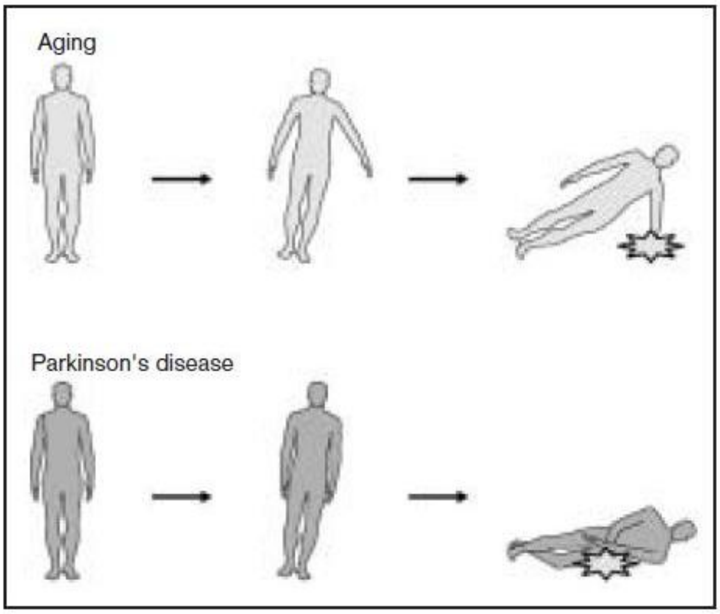


Figure 2 - Different fall mechanisms that occur with aging and with person with Parkinson disease [31].

As previously referred, stroke is one of the most prone neurological diseases to fall. One of the common effects in people who suffered a stroke is hemiplegia, which means that one side of the body remains paralyzed [28]. The hemiparesis is other effect of a stroke that provokes weakness on one side of the body. These two effects can conduct to a fall [29].The weakness of one of the lower limbs (i.e., hip, knee, or ankle) may lead to a loss of balance contributing to the occurrence, e.g. of a trip [28]. However, the upper limbs when in postural instability (e.g. NF) in particular the arms, are important to stabilize the balance and prevent people from falling [28], [34]. The stroke can also cause a loss of proprioception of one of the side of limbs, which can increase the possibility of a fall [28], [35]. Depending on the part of the brain affected, the stroke can cause for example: partial or total visual problems leading people not to notice the obstacles; walking disorders; and loss of muscle tone making the person unable to hold their body weight. All these factors can potentiate a fall event [28], [34]. The medication prescribed for patients who have experienced a stroke, and post fall syndrome, also contribute to the increase risk of falling [28], [29]. In these patients the fall can happen in several forms, since it depends on the part of the brain affected[28].

2.4 Consequences of a Fall

Several consequences can occur when a person fall, and some injuries can be severe. Head injuries, fractures, dislocations, lacerations, and hematomas are some examples of consequences of falls [36], [37]. After a fall, the victim cannot be able to stand up and can experience, for instance, dehydration, hypothermia, pressure injuries and pneumonia [36]. Once a subject experiences a fall may be afraid to fall again, have less confidence while walking and even develop depressions [36], [37].

Falls have a significant impact in the elderly community, representing high costs, thus, this is a concern that has been studied for several years, being a current theme, since it is necessary to minimize the effects [38].

2.5 Fall-related Systems

Much work has been done to minimize the number of falls and related injuries. Fall risk assessment tools and a few systems have been developed over the years in this respect. The fall risk assessment is performed to identify the subject's risk level [9].

Regarding clinical fall risk assessment tools, according to [9], it is possible to divide them in functional assessment tools and nursing assessment tools. The functional assessment tools, in general, are performed by physical therapists. The Timed Up & Go, Berg Balance test, elderly fall screening test, and Tinetti performance are examples of functional assessment tools. The nursing assessment tools access information about the patient, such as age, history of falls, mental status or cognitive impairment, and incontinence. A few examples of the tests performed are STRATIFY, Morse fall scale, and Hendrich fall risk model. Depending on the setting and the person responsible for performing the evaluation, one of the assessments may be more suitable. In the acute care setting, the nursing assessment scales may be more indicated since an easy and quick assessment is essential. In the case of outpatient setting, the functional assessment may be more correct. Regarding the extended care setting, the screening may not be advantageous since these patients, in general, have a high risk of fall and should be implemented a fall prevention plan [9].

There are tools to assess the risk of fall as were mentioned above. However, throughout the years, systems with sensors have been developed to detect a fall or to detect a NF in order to minimize post-fall consequences. Systems based on wearable sensors (for instance, accelerometers, gyroscope, and force sensors) and non-wearable sensors (for instance, cameras and thermal sensors) have been studied in this respect. In the next section, some of these systems will be presented.

2.5.1 Non-Wearable Systems

Non-wearables sensors are not in contact with the subject, for instance, the infrared-based depth sensors, thermal sensors, and pressure sensors. In Table 1, it can be observed the features computed, the sensors local attachment, the results, and the characteristics of the subjects who participated in the tests per study.

In articles [39]–[41] systems based on the Kinect depth camera were proposed. It is notable that due to the infra-red led of the depth camera, the illumination of the environment does not influence anything the system results. The aforementioned works used a Microsoft Kinect SDK. In [39], first, a binary image was achieved and the noise reduced, then with the canny filter the contour of the image was obtained. After that, the tangent vector angle of each white pixel in the contour was determined to detect the fall event in real-time based on thresholds. In [40], the authors used depth and colour information, the initial step was detect a moving object from the video by ground event segmentation and subsequently calculate the features. Through the support vector machine (SVM), the event was classified and if a fall is detected, a message is sent. In [41] the features were extracted based on joints data and the skeleton furnished by the camera. The approach performed, in this work, to detect a fall event was based on the adaptive threshold.

In articles [42]–[47] authors proposed an approach focused on a camera. In [42], the ellipse approximation and motion history images techniques were implemented to extract the features. Once again, the methodology followed to detect the fall event was based on threshold algorithms. In this particular case, the best outcome was achieved by combining the two techniques. In [43], the system developed concentrated on analysis of the human shape and head detection. Through the features extracted was possible detect a fall event. It is noteworthy, that in this work, the human shape was approximated to an ellipse, alike the study advanced by [42]. Regarding the fall event identification, it was determined according to certain established conditions. In [44], similarly to other papers mentioned above, the authors used the ellipse approximation and motion history images techniques. However, before applying these techniques, the background and foreground were segmented with the Gaussian mixture model. With the several features proposed in this work, it was possible to identify the fall event based on the threshold algorithms. In [45], the approach followed was centered on the accumulated image map and motion vector. Then, to identify a falling situation, the authors employed the k-nearest neighbor (KNN) classification algorithm. In [46], the objects were detected by the threshold approach. Next, the object was approximated by an ellipse and a rectangle in order to extract the features. The system detects whether or not a fall occurs through the rule-based approach. In [47], the authors used inclination angle and aspect ratio features, since it was computationally less intensive. The researchers applied threshold approach to detect a fall event.

Concerning the articles [48]–[50] the approaches developed to detect falls were not based in vision methods like the previous studies. In [48], a total different sensor was used, namely, the thermal sensor. This sensor allowed to notice the heat of any subject without contact. In order to detect the fall event using the temperature information, three different neuronal networks were tested, long short-term memory (LSTM), gated recurrent unit, and bidirectional LSTM (Bi- LSTM). The best result was achieved with Bi-LSTM. The system proposed in [49] was based on ultrasonic array sensors. The approach developed, in this study, decides whether or not a fall event happens based on distance variation calculated through ultrasonic sensors information. The last non-wearable system, [50], presented in this section was developed based on Ultra-WideBand (UWB) technology. In order to detect a fall event, the authors employed the random forest algorithm.

Table 1- Characteristics of non-wearable fall detection systems

Sensors	Best Sensor Location	Features	Subjects	Types of fall	Best result	Article
- Microsoft Kinect	- Cupboard	- Tangent vector angle	- 3 subjects - Height: 170– 185 cm - Ages: 24 – 40 years - Weight: 60 -90 kg	- Falling horizontal - Falling declining	- Sensitivity: 94.9 % - Specificity: 100 % - Accuracy: 97.1 %	[39]
	- Room	- Velocity - Acceleration - Width height ratio - Fall motion	*	*	*	[40]
	*	- Head velocity - Hip vertical velocity - Hip horizontal velocity - The height difference between hip and head	- 10 subjects - Ages: 25-33 years	*	F1-measure: 0.944	[41]

* Not specified

Table 1 – Continued

Sensors	Best Sensor Location	Features	Subjects	Types of fall	Best result	Article
- Video camera	- Room	- Standard deviation of orientation - Standard deviation of eccentricity - Motion velocity	*	- Sideward fall - Front fall	- Accuracy: fall 86.66% - Accuracy: non-fall 90%	[42]
	- Room	- Ellipse center - Ellipse orientation - Length of major and minor semi-axes of the ellipse - Head detection: edge points of the head; distance between the corresponding edge points on every consecutive line - Bounding box ratio.	*	*	- Accuracy 94.0%	[43]

* Not specified

Table 1 – Continued

Sensors	Best Sensor Location	Features	Subjects	Types of fall	Best result	Article
- Video camera	- Room	- Acceleration - Angular acceleration - Orientation standard deviation - Ratio standard deviation from the ellipse - Intensity of each pixel in motion history image	*	*	*	[44]
	- Room	- Motion Co-Occurrence features - Accumulated image map	*	*	*	[45]
	- Outdoor environment and home environment	- Aspect ratio - Angle	*	*	- Sensitivity: 90.0 % - Specificity: 98.93% - Accuracy: 92.5%	[46]

* Not specified

Table 1 – Continued

Sensors	Best Sensor Location	Features	Subjects	Types of fall	Best result	Article
- Video camera	- Room	- Aspect ratio - Inclination angle	*	- Sideway fall - Forward fall - Backward fall	*	[47]
- Thermal sensor	- Corner of room	- Temperature from each pixel	- 4 subjects - Age: 25-37 years - Height: 147-182 cm - Weight: 47-79kg	- Backward fall - Forward fall - Sideward fall	- Accuracy (95%CI): 93% (88.4%-95.9%) - Sensitivity (95%CI): 93% (85.5%-97.0%) - Specificity (95%CI): 93% (86.4%-96.9%)	[48]
- Ultrasonic array sensors	- On top and wall of the room	- Distance	*	*	- Accuracy: 92%	[49]
- Ultra-Wide Band	- Ceiling	- Time of arrival - Velocity	1 subject	- Forward fall - Sideward fall	*	[50]

* Not specified

2.5.2 Wearable Systems

Table 2, 3, and 4 present the characteristics of wearable systems like Table 1 for non-wearable sensors. Most of the wearable systems presented in Table 2 used at least an accelerometer or a gyroscope to detect fall events. Nonetheless, other sensors like magnetometers, reflective markers, electromyography probes, FSR are also used for fall detection.

López et al. [51] developed an approach to identify falls based on accelerometer signals. The data acquired through the sensor exhibited a different pattern when a person was falling compared to other activities performed by the subject. In this sense, the authors developed an algorithm based on thresholds which enabled the detection of a fall event in real-time.

In article [52], the authors used an accelerometer and gyroscope incorporated in a jacket. From the signals provided by the sensors, two characteristics were computed with a sliding window, namely, acceleration signal vector magnitude and Kalman filtered attitude angles. In order to detect the fall event eight classifiers were explored. Nonetheless, the best performance was achieved with KNN classifier. It is important to mention that the authors conducted two studies, one that considered the fall direction (i.e. backward and sideward falls) and the other that did not consider it. However, the best outcome was reached ignoring the fall direction.

Quadros et al. [53] developed a system based on IMU. In order to realize the best approach to classify a fall, three different methods were tested: the threshold-based method; threshold-based method with Maggwick's decomposition; and machine learning methods. Regarding machine learning methods, the KNN, Linear Discriminant Analysis (LDA), logistic regression, Decision Tree (DT) and SVM were implemented. Despite the two first methods reached accuracy values of 89.1% and 91.1%, respectively, the best performance was achieved with KNN classifier with 99.0% (Table 2).

Araújo et al. [54] proposed a system based on a smartwatch, more specifically on accelerometer signals. In this work, the authors proposed a threshold algorithm method to detect a fall. The algorithm implemented can be divided in four main stages. Firstly, it was detected a free fall phase. Subsequently, it was verified if occurred impact on the ground. Then, the standard deviation was calculated and analysed. Lastly, the state of the user was recognized in order to distinguish a fall from other activities.

Tao et al. [55] proposed a shoe system with eight FSR in each insole to detect a fall and the fall direction. In this study the sensors needed to be first calibrated. Then, the researcher to reduce the number of sensors implemented the Principal Component Analysis (PCA). Different values of pressure corresponding to a different action. No pressure means lying on the floor, low-pressure corresponding to sitting, and high pressure means that the subject is walking or standing. In this article, an artificial neural network was used to classify the pressures mentioned above. However, to detect a fall direction, it was employed the nearest neighbour method that learned from tilted standing data, once the data to detect the fall direction was reduced.

Montanini et al. [56] developed a system based on the FSR and accelerometer signals placed on shoes. The FSR signals were used to identify the gait cycle phases and the accelerometer to give information on foot orientation. Concerning data acquisition, a laboratory experiment was performed to simulated falls and activities of daily life with healthy subjects. Moreover, real-life data was collected from two elderly people performing activities of daily life. The algorithm developed to detect a fall event was based on thresholds.

The articles [57]–[59], developed systems to detect a pre-impact fall. Sivaranjani et al. [57] proposed an approach based on gyroscope and accelerometer sensors incorporated in the jacket. The jacket inflates when a pre-impact fall phase was detected through a threshold-based algorithm implemented. With this system, the authors intended to protect the hip and the head of the subject and reduce the impact on the ground. The approach developed by Rescio et al. [58] was based on surface electromyography signals to detect the pre-fall phase. For data gathering a movable platform was utilized to induce a fall. With the data acquired ten time-domain features were calculated. However, with the intention of reducing the feature set, the Markov Random Field selector based on Fisher-Markov was implemented. Afterward, the LDA classifier was selected to identify the pre-fall phase. It is worth mentioning that an analysis of the size of the window and the most appropriate frequency was also carried out. Furthermore, a post-processing was implemented with a filter by vote (50ms temporal window) to improve the result performance. In article [59], a movable floor surface was used to create slip-induced falls. This system was based on the IMU and six cameras to measure the 3-D position of the reflective markers. The participants, while doing the test, used a harness for protection. In order to detect fall events prior to impact phase, a threshold-based approach was addressed.

Systems for NF detection have also been developed in addition to the systems mentioned above. The approaches implemented in these systems are described below.

Aziz et al. [60] developed a system based on acceleration and angular velocity signals to detect a NF. In this study, the participants watched videos with real fall situations and performed a similar falls or NF. In order to establish the effect of the number of sensors and their location on the outcomes, the researchers studied different combinations. The sensors were attached on thighs, ankles, head, waist and chest. The best combination was determined through the results achieved by SVM with radial basis function.

Iluz et al. [61] proposed a system based on inertial data to detect missteps in subjects with PD. Thus, the authors developed a novel algorithm based on thresholds. In this study, data were collected in the laboratory with PD patients using a harness to ensure their safety. In order to simulate the missteps were placed obstacles in the path. In addition, real data was also collected for three days, however this data due to the lack of notes were considered “suspected missteps”. Besides the novel algorithm developed, this study also implemented learning algorithms (e.g. K-means) and machine learning algorithms (e.g. Ada boost) with different features from those mentioned in Table 4. However, the results reached with these approaches were not satisfactory. This system has shown potential to help identify subjects with high risk of falling.

Chehada et al.[62] developed a system based on acceleration data. In this research, only one accelerometer was used for data gathering. However, seven different positions (i.e. left ankle, left pocket, left wrist, right ankle, right pocket, right wrist and chest) on the body were tested to determine which is the best location. The tests were conducted with people blindfolded and listen to loud music. In order to simulate the near fall, the authors placed an obstacle in the path. A Gaussian model with threshold algorithm was implemented to detect stumbles and the best result was achieved with the accelerometer placed on chest.

Kareal et al. [63] developed a system based on accelerometer signals to detect stumbles. For the purpose of simulating stumbling, the participant had a rope attached to each ankle to disturb both legs. The tests were performed in a treadmill equipped with a harness and emergency stop to ensure the participants safety. The approach implemented in this study was based on wavelet with threshold.

Table 2 – Characteristics of wearable fall detection systems

Article	Sensors	Best Sensor Location	Features	Subjects	Types of fall	Best Results
[51]	- Accelerometer	- Torso	- Magnitude of the acceleration	- 10 subjects - Age: 25-56 years - Weight: 65-80 kg - Height: 169-185 cm	- Forward fall - Backward fall - Sideward fall - Vertical fall	- Sensitivity:93.2% - Specificity: 87.5%
[52]	- Accelerometer - Gyroscope	- Torso	- Attitude angle - Acceleration signal vector magnitude	- 20 subjects - Age: 20-26 years	- Backward fall - Sideward fall	- Accuracy: 95.8% - Average sensitivity: 95.8% - Average specificity: 99.2%
[53]	- Accelerometer - Gyroscope - Magnetometer	- Wrist	Related to: - Acceleration - Displacement - Velocity - Spatial orientation angles	- 22 subjects - Age: 26.09 ± 4.73 years - Height: 1.68 ± 0.11 m - Weight: 67.82 ± 12.24 kg	- Forward fall - Backward fall - Sideward fall - Fall after rotating the waist clockwise and counterclockwise	- Accuracy: 99.0 % - Sensitivity: 100% - Specificity: 97.9%

Table 2 – Continued

Article	Sensors	Best Sensor Location	Features	Subjects	Types of fall	Best Results
[54]	- Accelerometer	- Wrist (smartwatch)	- Square root of the sum of the square of the axis of the accelerometer - Standard deviation	- Age: 30 years	- Fall with support - Fall without support	- Sensitivity: 92.9% - Specificity: 95.5% - Accuracy: 94.4%
[55]	- FSR	- Foot (5 metatarsal heads, hallux, and the heel)	- Force	*	- Forward fall - Backward fall - Sideward fall	- Correct classification rate: 75%
[56]	- FSR - Accelerometer	FSR: - Heel, the 1st, and 5th metatarsal heads Accelerometer: - Shoe	- Gait cycle phases (heel contact; flat foot contact; heel off; and limb swing) - Pitch - Roll	Laboratory experiment: - 17 subjects - Average age: 28 (±9.3) - Average height for males: 176 (±8.4) cm - Average height for females: 164 (±5.3) cm	- Backward fall - Forward fall - Sideward fall	- Accuracy: 97.1% (Laboratory tests)

* Not specified

Table 2 – Continued

Article	**Sensors**	**Best Sensor Location**	**Features**	**Subjects**	**Types of fall**	**Best Results**
				- Average weight for males: 72(±8.4) kg - Average weight for females: 62(±11.1) kg Real-Life experiment: - 2 elderly subjects - Age: 67 years - Height for female: 170 cm - Height for male: 183 cm - Weight for female: 67 kg - Weight for male: 99 kg		

Table 3 - Characteristics of wearable pre-impact fall systems

Article	Sensors	Sensor Location	Features	Subjects	Types of fall	Best Results
[57]	- Accelerometer - Gyroscope	*	- Acceleration - Angular velocity	*	*	*
[58]	-Electromyography probes	- Lower limb (Gastrocnemius lateralis and tibialis anterior muscles)	- Integrated EMG - Co-Contraction index - Willison Amplitude	- 15 subjects - Age: 32.6 ± 9.3 years - Weight: 68.3 ± 9.2 kg - Height: 1.74 ± 0.4 m	- Forward fall - Backward fall - Sideward fall	- Specificity: 89.5 % - Sensitivity: 91.3% - Lead time before the impact 770ms
[59]	- IMU - Reflective markers	- Close to sternum	- Trunk sagittal extension angle - Trunk sagittal angular velocity	- 10 elderly - Age: 75 ± 6 years - Weight: 74.1 ± 9.1 kg - Height: 174 ± 7.5 cm	- Backward fall (slip)	- Sensitivity: 100% - Specificity: 95.65% - Response time: 255ms

* Not specified

Table 4 - Characteristics of wearable NF systems

Article	Sensors	Sensor Location	Features	Subjects	Types of NF	Best Results
[60]	- Accelerometer - Gyroscope	- Left Foot - Right Foot - Right Thigh - Waist - Head	- Mean - Variance	- 10 subjects - Age: 22-32 years	- Slip - Trips - Incorrect transfers - Missteps - Hit and bump by another subject	- Sensitivity: 100% - Specificity: 100%
[61]	- Accelerometer - Gyroscope	- Lower Back	- Peak difference - Maximum amplitude - Step number - Entropy - Frequencies above threshold	- 40 PD subjects - Age: 62.16 ± 10.02 years	- Missteps	Laboratory tests: - Specificity: 98.6%
[62]	- Accelerometer	- Chest	- Maximum values of magnitude acceleration vector	- 9 subjects	- Stumble	- Sensitivity: 94% - Precision: 99%

Table 4 – Continued

Article	Sensors	Sensor Location	Features	Subjects	Types of NF	Best Results
			- Non linear energy operator of magnitude acceleration vector - FFT of magnitude acceleration vector			
[63]	-Accelerometer	- Sacrum	- Wavelet decomposition	91 subjects: - 41 young - 50 elderly -Age: 24±4 years and 67± 5 years	- Stumble	- Sensitivity: 98.4% - Specificity: 99.9%

2.6. Discussion

Throughout the research conducted in this chapter, it was possible to establish the existence of numerous factors that increase the risk of falling. Among them, neurological diseases, age, gender, race, medication and physical conditions [5], [23]–[30]. Severe injuries can arise from a fall, for instances, fractures, head injuries, and death in the worst case [6], [36], [37]. Hence, there is a considerable interest by researchers in developing systems to minimize fall-related problems such as those mentioned above.

The apparatus used to detect falls or NF can be divided into two categories: non-wearables and wearables. Concerning non-wearable technology, several works were encountered which resorted to the use of cameras [39]–[47]. Nevertheless, other types of sensors were used, such as thermal sensors and UWB [48], [50]. Of all the studies mentioned in the Table 1, it was observed that the majority of the sensors are placed in a room [39], [40], [42]–[45], [47]–[49].This leads to one of the problems associated with non-wearable sensors. Typically, the sensors are restricted to a specific locale and indoor environment. Apart from this problem, there are others, especially with regard to cameras. First, the confidentiality problem since personal information of the participants is recorded. Second, these sensors often have occlusion problems [64].

With regard to wearable sensors, several works have been developed. While in the category of non-wearable only systems for detect falls were presented, in this category the systems were divided into three groups. Namely, fall detection (i.e. the subject impacts on the ground), pre-fall phase detection and NF detection (i.e. the subject can restore balance) systems. These three groups are scrutinized in Tables 2, 3, and 4, respectively. There is at least one aspect in common between these groups: the use of an accelerometer sensor and/or gyroscope sensor in the majority of studies [51]–[54], [56], [57], [59]–[63]. Besides these sensors, other studies also used magnetometer [53], FSR [55], [56], and electromyographic probes [58]. Concerning the position of the sensors on the body differs from study to study. The sensors are placed in one or more of the following locations: lower limbs, upper limbs, head and torso [51]–[63]. For the purpose of determining the best possible sensor placement, two studies have investigated this issue [60], [62]. Both wearable and non-wearable sensors have some disadvantages, such as follows: i) the subject needs to remember to place the sensor on the body; ii) these devices are dependent on an external supply; and iii) if the user performs a sudden movement, the device can detect incorrectly a fall [64].

In both categories, wearable and non-wearable, the whole fall and NF data was simulated by the participants. The problem with simulated falls lies in the fact that their pattern may be different from a real fall. This situation may lead to better results than would be achieved with real falls [14].

In the majority of the articles mentioned in Tables 1 to 4 the experimental data were performed with young and healthy adults [39], [41], [48], [51]–[54], [56], [58], [60]. It is presumed that these systems will not be tested in the elderly because of their safety. One of the problem of falls or NF simulated by young adults is related to postural control characteristics, since these are different in the elderly when confronted with perturbation [65]. However, three studies included tests with the elderly [59], [63], and one with PD patients [61].

Throughout the fall-related system section of this chapter, the most commonly implemented approaches were based on machine learning [45], [48], [50], [52], [53], [55], [58], [60], [61] and thresholds algorithms [43], [44], [46], [47], [49], [51], [54], [56], [57], [59], [61]. Other algorithms were also applied such as the wavelet [63] and the Gaussian model [62].The performance achieved with these algorithms overall was high. The sensitivity and specificity values found in the studies reported range from 91.3%-100% and 85%-100%, respectively.

In conclusion, there are some problematic aspects associated with the fall, which need to be addressed in order to develop more robust systems. These include, for instance, gathering data from people with associated fall risk factors and real falls. The development of approaches that allow the detection of the fall before the impact is also an important aspect to be addressed. In order to be able to develop strategies to avoid an eminent fall and, thus, minimize the consequences.

CHAPTER 3 – SMART WALKERS

The main objective of this chapter is to conduct a survey of the strategies implemented in smart walkers to detect and prevent a fall. Initially, a contextualization of the different types of walkers and their final purpose is accomplished. Following an overview of the ASBGo SW, which includes its accessories and the functionalities provided. Finally, a comprehensive survey is performed of the fall-related strategies previously studied when using a walker in literature. Moreover, a research of commercial smart walkers and patents that have implemented a fall-related strategy is also carried out.

3.1. Introduction

Assistive devices have an essential role for people with mobility impairment. These devices have some benefits, such as enhancing the balance, decreasing the load on the lower limbs and assist propulsion. For elderly people, these kind of devices are fundamental due to the bad consequences of neurological and age-related diseases [66]. Thus, the recommendation of the assistive devices should be done very carefully, and it is necessary to take into consideration other conditions. Such as, vision, cognitive function and muscle strength [67].Walkers are examples of assistive devices, and they are widely used to partial body weight support and to improve the dynamic and static stability. There are different types of walker: standard, two-wheeled, rollator and hands-free [67]. These devices are shown in Figure 3. Depending on the user's condition, each one of these devices can be properly prescribed. For subjects who have difficulty in lifting the walker of the floor, a two-wheeled walker can be more suitable than a standard walker [67], [68]. The rollator (four- wheeled) is advised when subjects do not need constant bodyweight support but is important a large walking base [67], [69]. Finally, the hands-free allows the reduction of weight supported by the upper extremities and promoted at the same time support of the corporal weight [67]. Over time, these assistive devices suffered great evolutions, mainly because of electronic incorporation, i.e., human-machine interface, sensors, and control [67][69].

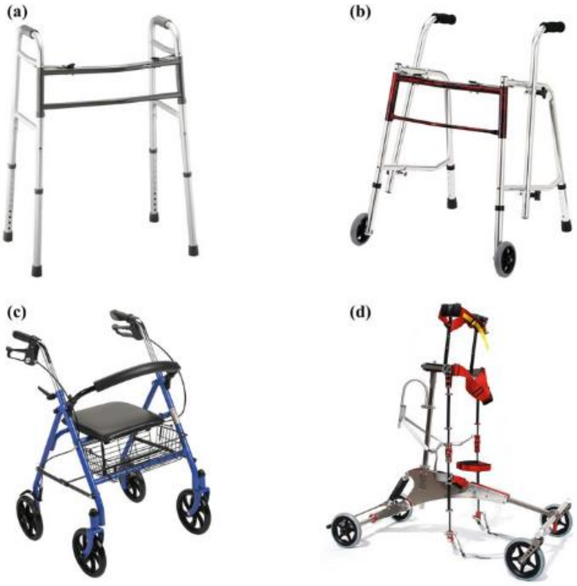


Figure 3 – Different walkers: (a) Standard; (b) Two-Wheeled; (c) Rollator; (d) Hands-Free [67].

3.2. ASBGo SW

The ASBGo SW was developed by the Adaptive System Behaviour Group. This assistive device has rehabilitation and physical therapy purpose, intending to aid patients with gait disorders, and improve their physical conditions. The ASBGo SW provides a technological alternative to standard walkers with higher levels of safety and quality rehabilitation. A threshold-based fall prevention strategy was already implemented in the ASBGo SW [70]. This assistive device can be divided into two parts: the upper part and the lower part.

The ABSGo SW has a handlebar with two handle grips that allow driving the device at the upper part. The smart walker is also equipped with a wooden table with forearms support (possible to adjust using velcro tape) to sustain the patient's weight. In order to adjust the height of smart walker, the upper part fits into two adjustable electrical columns. A touchscreen is located on the center of the wooden table for better human-machine interaction. An emergency button is available in front of the monitor to immediately block the smart walker in case of dangerous situations. Furthermore, this device is equipped with a Kinect, an infrared sensor, and an IMU. Figure 4 depicts the upper part of the ASBGo SW.

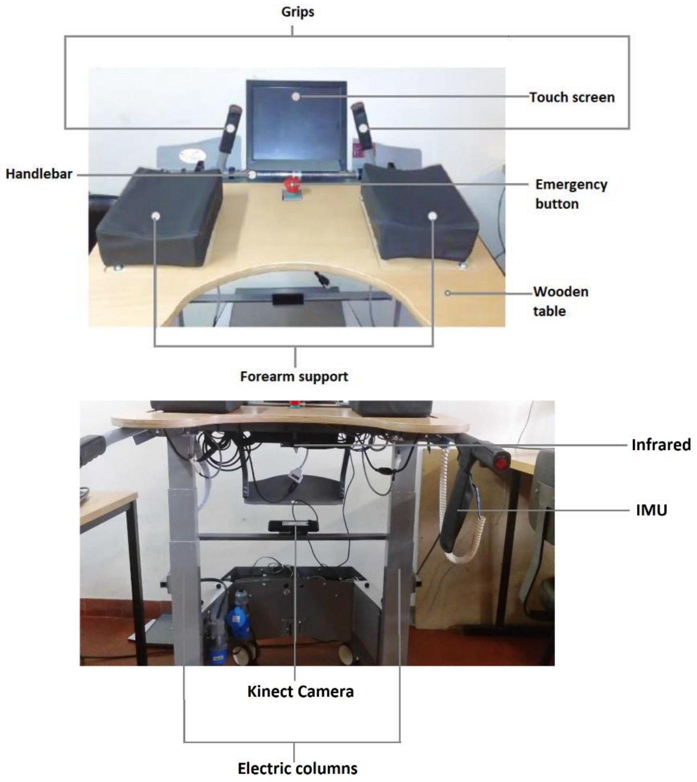


Figure 4 - Upper part of ASBGo SW.

The lower part is equipped with four wheels; encoders to record movement information (velocity and distance); motors on rear wheels which are responsible for the walker movement; manual brakes to lock and unlock the motors; and a laptop responsible for collecting, processing and saving data. The device also has two rechargeable batteries to provide power to the whole system. The lower part of the ASBGo SW is shown in Figure 5.

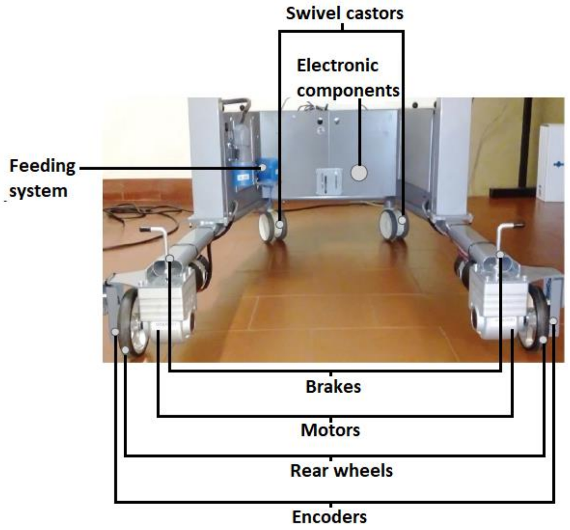


Figure 5 - Lower part of ASBG SW.

Table 5 shows all the sensors, overall, and their purpose. These sensors allow the computation of different features related to the patients and, consequently, relevant functionalities.

Table 5 - Purpose of all sensors present on ASBGo SW

Sensors	**Purpose**	**Previous works**
Angular potentiometer	Detect the angular displacement variation performed by the handlebar (forward/stop).	[71]
IMU (wearable)	Detect gait events and assessment of fall risk.	[72]
Infrared	Measure the distance between the user and the device.	[73][70]
Linear potentiometer	Detect the linear displacement variation performed by the handlebar (turn left/right).	[71]
Strain gauge	Detect if the patient has the forearms on the forearm support.	[70]
Ultrasonic	Obstacles detection (Not available at the moment).	[74]

The functionalities of the ASBGo SW were developed in order to improve patient rehabilitation. The acquired patients' information is related with their stability, balance, and gait. A multitasking game to evaluate the capacity of the patient to do more than one task and information related to their lower limbs is also available. More information about the functionalities can be found in Table 6.

Table 6 - Functionalities of ASBGo SW

Functionalities	**Sensors**	**Purpose**
Balance	IMU	Evaluate balance. Three static tests (static stance, static semi-stance tandem, and static semi-tandem) and one dynamic test.
Biofeedback	Kinect camera (Not available until the moment)	Show the lower limbs on the screen for visual feedback and gait correction stimulation.
Forearm support	Strain gauge	Record force values.
Gait analysis	IMU (foot) Kinect camera (Not available until the moment)	Compute spatiotemporal parameters.
Multitasking game	*	Evaluate reaction time and the capacity of the patients to maintain a natural gait.

Table 6 - Continued

Functionalities	Sensors	Purpose
Stability	IMU	Evaluate the stability. Four tests are performed: front tilt, rear tilt, lateral tilt to the left, and lateral tilt to the right.

* Not specified

Additionally, different modes of driving can be chosen when patients use the smart walker, such as local driving, remote driving, autonomous driving, and treadmill. The physiotherapist can therefore choose the most appropriate mode, depending on the patients' difficulties. The local mode allows the selection of parameters of driving, such as the velocity and the curvature of the walker. In this mode the user guides the walker. On the other hand, in remote control, the physiotherapist has a joystick to control the walker's trajectory in the environment. A treadmill mode can also be select. In this mode, the user, while uses the treadmill, can benefit the ASBGo SW functionalities at the same time. Finally, the autonomous mode is not implemented in the latest version of the ASBGo SW. This mode allows the physiotherapist or the user to choose the desired coordinate position.

With respect to fall detection in ASBGo SW, an infrared sensor was placed on the smart walker in order to measure the distance from the user to the walker. Based on this distance, an algorithm was developed to detect a forward fall because, in these events, the distance will decrease abruptly. In this case, the walker stops. Regarding backward falls two FSR were placed on each handlebar, and if the user does not have his hand on the handlebar the walker will stop. The same happens when the user does not have the forearm in forearm support. In this case, it detects by using FSR. Note that the walker does not move backwards, thus if the user pulls the walker, the walker stops [70].

3.3. Current Smart Walkers

3.3.1. Search Strategy

On October 22nd, 2018, a comprehensive survey was conducted on Scopus and on the Web of Science. The following keywords were used: ("Walking support" AND fall), ("Smart walker" AND fall), ("Smart rollators"), and ("Walking-aid" AND fall). This survey has the purpose to know which smart walkers are available in the literature and the fall-related strategies applied.

The points of interest during the analysis of the systems were the following: i) the sensors used; ii) the sensors location on the walker; iii) the strategy implemented to prevent and detect a fall or a NF event; iv) the algorithm developed.

3.3.2. Search Results

A few smart walkers were found through the search performed, which included strategies implemented to detect or prevent a fall. The search results, i.e., the smart walkers and fall-related strategies are described follows.

RT Walker is a passive device used in three studies [75]–[77]. This assistive device has rear wheels with powder brakes, which enable the change of the torque considering the current applied. All the above-mentioned studies reported the stop of the walker when they detect that the user is falling. In article [75], two laser range finder (LRF) were used. One was located at the same height as the user's hip to measure the distance along the vertical direction between the walker and the user. The other laser was placed at the base of the walker and measure the distance between the user's leg and the walker. Based on the information of LRF was generated the 7-link human model. Next, a stability region was determined based on support polygon formed by the walker and the feet of the user. The system detects that the user is falling when the center of gravity is out of the region of stability.

In article [76], the device had two stereo cameras in order to track the head, hands, shoulders, and hip to get the 3D upper body model. The 3D coordinates of the parts of the body were used to classify the normal state walking, sitting, standing and falling. In article [77], a depth camera was used to extract the upper body centroid position. In order to detect human action (standing, walking, sit, fall right, fall left, fall backward, fall, and fall forward), two approaches were used, namely, multivariate normal distribution function and Hidden Markov Model (HMM). The first approach mentioned detected 96.25% of the falls, and the second one 98.75% of the falls.

Xu et al. [78] developed an approach to prevent the user of the walking-aid robot from falling. Two human motion intentions were studied, the upper and lower limbs of the user. The FSR were positioned on the handle to monitor the user's upper limbs, and the LRF was placed on the lower part of the walker to monitor the user's leg movement. A state of normal and abnormal gait was distinguished in this work. In the second state mentioned different falls could occur, namely falling forward, to the left and to the right. The SVM was the approach used to classify the state of gait and, consequently, detect a possible fall. If the user is falling the walker stops moving.

Irgenfried et al. [79] developed a device that uses a 6D-force/torque sensors (FTS) for connecting the walker with the handlebar. A mathematical model of the human body was applied to help identify possible fall situations in FTS signal. With the purpose to test the system the participants simulated stumbles. The experimental results revealed a peak in the sensor values that can be used to detect a possible fall. In order to prevent a fall, the authors suggest stopping or slowing down the walker.

Huang et al. [80] used a different approach. In this work, wearable and non-wearable sensors were used to detect possible falls, while in the works previously presented only non-wearable sensors were employed. A tri-axial accelerometer, a tri-axial magnetometer, and a tri-axial gyroscope were positioned on the waist, both thighs, and both shanks to measure the acceleration and the angular velocity. The force sensors used on the handlebar enable to obtain the forward and lateral force and rotation torque of the walker. The center of pressure (COP) was extracted, and then, the authors calculated the relative position between the midpoint of the feet and the COP. This information is important to determine if a fall happen along the horizontal direction. In order to detect the vertical falls, was calculated the height of the human waist. The fuzzy threshold was the approach implemented to detect the fall. Different types of fall were performed: due to weakness in the legs (vertical direction); falling forward (horizontal direction); and falling to the left side (horizontal direction). When the possible fall is detected, the walker brakes and stops.

Mou et al. [81] and Azqueta-Gavaldon [82] developed a walker targeted to PD patients and the elderly. The first authors used a LRF in order to analyze the gait. Furthermore, the force sensors were placed on the handle to determine the follow actions: turning, stop, push, pull, and going backward. Lastely, through an adaptative HMM was possible to classify the three kinds of gaits from two type of sensors (festinating gait, freezing of gait and normal gait). For this purpose, the information acquired through the force sensors and the LRF was considered. In order to prevent users from falling the walker stops when the sudden push is detected.

Azqueta-Gavaldon et al. [82] developed a walker that monitors the walking movement of the user. The rollator used in this study has a depth camera placed at the same height as the rollator seat. This sensor measures the distance between the user's leg and the rollator. In order to test the system, three different possible falling situations were tested, freezing of limbs, stumble, and loss of balance (all forward falls). It is important to highlight that the tests were performed by healthy subjects. However, the system was conceived for subjects with lower reflexes, so it is important to test the system with these patients instead of healthy people. When the distance between the user and the rollator is higher than a threshold, the rollator stops to prevent the fall (the delay in brake activations is 80-90ms). The overall accuracy was 95% and the precision was 93% of the braking system.

It is possible to note that those different sensors were used in order to detect a possible fall event. The force sensors were used in the handlebar to monitor the upper limbs as well as the LRF placed on the upper part of the walker. On the other hand, the LRF placed on the lower part of the walker allowed monitoring the lower limbs of the user. The cameras were used to track the head, shoulder, lower limbs and hips, and to calculate the body centroid. The wearable sensors were used only in one study, and they were placed on the waist, thighs, and shanks. An overview of the sensors used as well as their positions on the walker and user is shown in Table 7 and Figure 6.

Table 7 Sensors present on the smart walkers

Article	**Force Sensors**	**LRF**	**Accelerometer; Gyroscope; Magnetometer**	**Stereo Camera**	**Kinect**	**Depth Camera**
[75]		1,2				
[76]				3		
[77]					4	
[78]	5	6				
[79]	7					
[80]	8		9,10,11			
[81]	12	13				
[82]						14

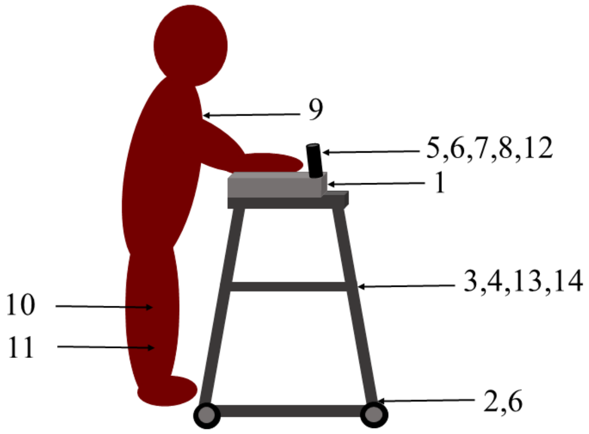


Figure 6 – Scheme of the location of the sensors on the smart walkers and the user according to Table 7.

3.4. Commercial Smart Walkers

A search was performed on Google in order to find which smart walkers are commercially available nowadays. In this search, only the walkers with the characteristics mentioned in the previous section were included. Only two smart walkers were found, namely RT.1 [83] and RT.2 [84], shown in Figure 7 and 8, respectively .

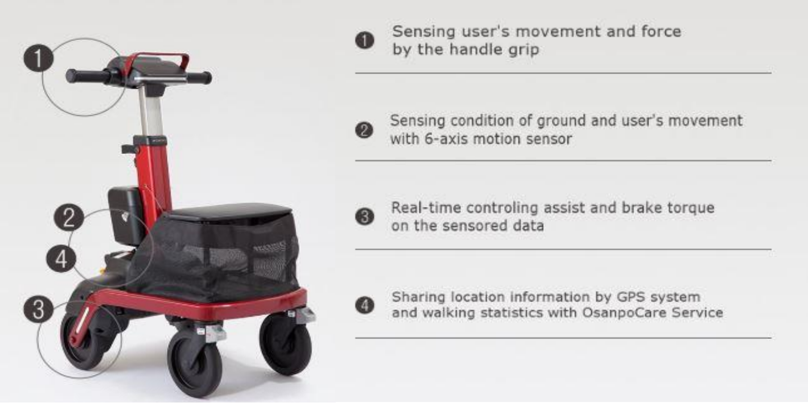

1
2
4
3
❶ Sensing user's movement and force by the handle grip
❷ Sensing condition of ground and user's movement with 6-axis motion sensor
❸ Real-time controling assist and brake torque on the sensored data
❹ Sharing location information by GPS system and walking statistics with OsanpoCare Service

Figure 7 - Commercial smart walker RT.1 [83].

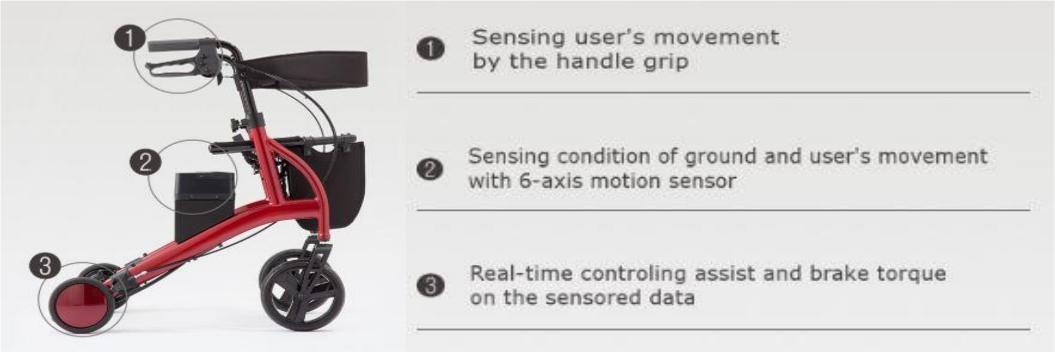

1
2
3
❶ Sensing user's movement by the handle grip
❷ Sensing condition of ground and user's movement with 6-axis motion sensor
❸ Real-time controling assist and brake torque on the sensored data

Figure 8 - Commercial smart walker RT.2 (Image adapted from [84]).

The RT.2 and the RT.1 smart walkers provide a few particularities that became the use of the walker more attractive. When the user is on uphill the torque is automatically controlled, and on downhills the brake torque is also automatically controlled, which facilitates its use. In the case of lateral inclination, it is possible to walk straight despite the gravity. If the user releases the walker on the slope unintentionally (detect by the sensor in the grip), the walker will stop. Which would not happen in standard walkers. Furthermore, if there is an abrupt increase of speed for some reason, e.g. a fall, the speed will decrease due to automatic braking. The RT.1 provides other services related to internet of things (IoT) [83], [84].

3.5. Smart Walker Patents

3.5.1. Search Strategy

Between October 31st, 2018, to November 4th ,2018, an advanced patent search was performed on United States Patent and Trademark Office (http://patft.uspto.gov). On November 10th, 2018, another patent search was performed on Espacenet (https://worldwide.espacenet.com). The select keyword used from search were [("walker") AND ("near-fall" OR "falling" OR "fall prediction" OR "fall detection" OR "fall prevention")]. The selection of patents was performed in three steps, first based on the title, second based on abstracts and schemes and ultimately based on the full text.

3.5.2. Search results

At the end of the process, a total of 10 patents were selected related to walkers. On United States Patent Trademark Office, a total of 17550 patents was found, while on Espacenet, it was found 126 patents. After eliminating the duplicate patents and based on title, 201 patents were selected. Based on abstracts and drawing 185 patents were excluded. Finally, based on full text, 10 patents were selected, and from these 8 were from States Patent Trademark Office and 2 from Espacenet, respectively. Figure 9 shows a flow diagram of the whole process for selecting patents.

The patents study performed on two databases included patents related to biological systems, soybean, child walkers, wagering games, walker improvements, another type of assistive device (for instance, canes), surgical devices among others not related with fall detection and prevention. All of these devices and systems were excluded. It is noteworthy that this search only included electronic systems implemented in walkers that can prevent or detect falls. The three patents presented in Table 8 are the closest or more important related to the concepts developed in the previous section.

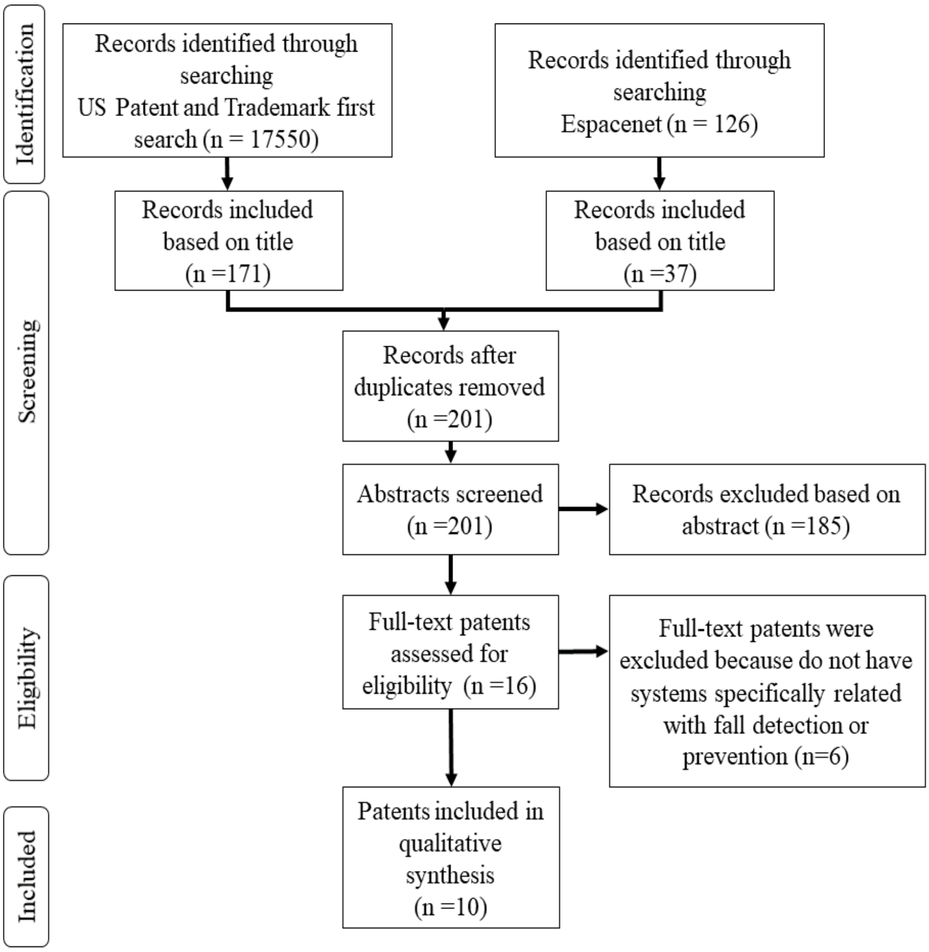

Identification
Records identified through searching US Patent and Trademark first search (n = 17550)
Records identified through searching Espacenet (n = 126)
Screening
Records included based on title (n =171)
Records included based on title (n =37)
Records after duplicates removed (n =201)
Abstracts screened (n =201)
Records excluded based on abstract (n =185)
Eligibility
Full-text patents assessed for eligibility (n =16)
Full-text patents were excluded because do not have systems specifically related with fall detection or prevention (n=6)
Included
Patents included in qualitative synthesis (n =10)

Figure 9 - Flow Diagram PRISMA for Patent Review.

Table 8 – The three most significative patents to the fall prediction/detection system developed

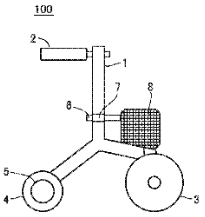
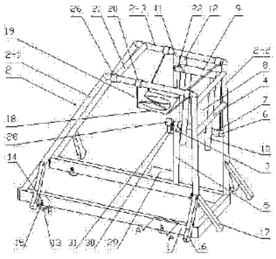
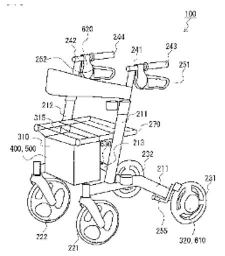
Name	**Patent Number**	**Scheme**
Walking assistive device	9,687,410	

Table 8 - Continued

Name	**Patent Number**	**Scheme**
Anti-falling walker	CN20181022168720180317	 26 21 20 2-3 11 32 12 9 19 2-1 2 18 26 14 2-2 8 4 7 6 10 3 5 17 15 13 31 30 29 A A 1 16
Electric walking assistance device	WO2014JP0270620140522	 100 620 242 244 241 243 252 251 212 211 315 270 310 400, 500 630 213 232 211 231 222 221 255 320, 610

The “Walking assistive device” approach is based on the distance to prevent the user from falling. This walker has a distance detection sensor, which enables the measurement of the distance between the walker and the user, and a control unit to control the driving unit that moves the walker. If the distance measured between the walker and the user is out of the safety distance range, the control unit will actuate in the motor and determine the rotation of this in order to prevent the user from falling. In this patent, other embodiments are addressed based on distance as well. One of these embodiments also uses a sensor that calculates the angle of inclination and sends it to safety distance range setting unit. Other embodiments are used, such as a sensor of pressures, speed sensor, among other always to improve the method to prevent falls. The patent number as well as its scheme are presented in Table 8.

The second patent in Table 8, the “Anti falling-walker” has mechanisms to avoid falls. This device has a hip strap that, in case of fall, can hold the user of the walker and prevent injuries associated with the possible fall. The walker is equipped with an alarm device, a displacement sensor, and a motor. All of these components are connected to the battery. When the user is falling and reaches a set position the displacement sensor will send a signal to the alarm device. The motor turns on and seat cushion will rotate downward, in order to prevent a fall.

The third patent in Table 8 has motors and a speed limiting unit, which allow the limitation of the rotation of the wheel when the speed is greater than a predetermined value. This device also has a leg motion detection unit that allows measuring the distance between the user and the walker. When the distance increases, the motor can stop or reverse the rotation. An attitude detection sensor is used to detect the attitude of the device and control the rotation of the wheel. With a grounding sensor is possible to know if the device is in contact with the ground. It is possible to control the rotation of the wheel according to the situation. In one situation which the walker changes the inclined state (for example, downward inclined state to the horizontal state) through these sensors is possible to detect the inclination, and the motors can be controlled in order to provide stable conduction. It is possible to conclude that all the sensors in the walker allow safe conduction that prevents the user from falling. This device also has a brake unit on handle controlled by the user.

3.6. Discussion

The walkers are essential to improve the quality of life and also the subject's mobility problems [66]. According to the WHO, millions of falls occur each year, and a large portion happens in older people. Medical assistance is needed in many cases [6]. Thus, it is possible to note that there is a clear necessity to decrease the number of falls and the injury associated with them.

Concerning smart walkers, in all devices found the only fall prevention strategy implemented was to stop the walker [75]–[82]. This prevention measure provides an increase in user safety, mainly in forward falls due to the support act of the walker. Regarding sideward falls and backward falls, this mechanism cannot be so efficient. Thus, more studies should be performed, and other strategies should be implemented in order to mitigate this problem improving the user's stability and balance. Regarding commercial smart walkers, only two devices were found, and the fall prevention strategy is similar to the literature. In relation to the patents selected, the approaches presented to prevent fall events are identical to those found in both literature and commercial devices.

CHAPTER 4 – SYSTEM OVERVIEW

Falls can culminate in various injuries and, in the worst-case scenario, death. In this respect, the main goal of this work is to develop a strategy that can be implemented in ASBGo SW to improve patient safety. In this chapter, an initial overview of the system is presented, which encompasses the selected sensors as well as their positions on the body. Then, all the components that compose the device are described in more detail, focusing mainly on the sensor characteristics. Once the device that was adapted for this work has been introduced, the experimental procedure for data acquisition is explained. Finally, the whole implemented methodology is presented, which can be divided into four parts: i) data processing; ii) data labeling; iii) feature computation; and iv) machine learning methodology.

4.1. System Overview

In order to collect data, a rollator, an IMU on lower trunk, and FSR underneath the shoes were used (Figure 10 a, b and c). The data acquired with these sensors will be used to implement the NF, INF and gait events detection approaches (Chapter 5, 6, and 7).

The IMU and FSR sensors and their location were selected for two main reasons. First, ASBGo SW already has one IMU implemented to be used on lower trunk. Second, the FSR can provide part of the information that can be acquired with the depth camera which is being developed. Furthermore, through literature analysis, it was possible to notice that the IMU as well as the force sensors have been used to develop fall-related strategies. In addition to the criteria aforementioned, the following aspects were also taken into account for the choice of sensors and their location: fewer sensors possible; not limiting the user's movements; lightweight; and easy to place on the subject.

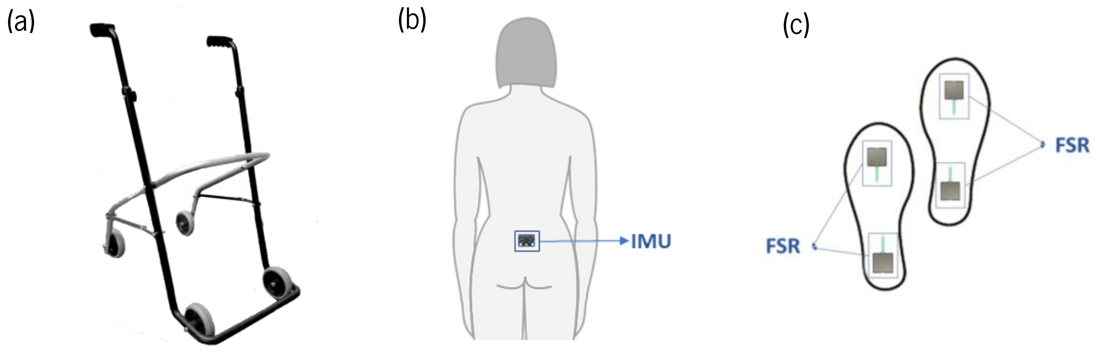


Figure 10 – (a) Rollator; (b) IMU position on-body; (c) FSR position on insoles.

The IMU was placed, more specifically, close to the COM, i.e., near to L5 lumbar vertebra and S1 sacral vertebra in adults [85]. The body COM is important in the study of human gait and balance control [86]. The acceleration acquired in the lower back position provide information about gait events (e.g., HS and TO) and gait parameters, such as walking speed [87]. This information can be useful to detect a NF once the balance and gait modifications can increase the risk of falling.

In this study FSR were used to detect NF and INF as the IMU but also as ground truth for the detection of gait events from inertial data (Chapter 7). The shoes used were equipped with two FSR in each insole, one placed on the front of the insole (toe zone) and the other on the back of the insole (heel zone) as shown in Figure 10 c).

The system is therefore formed by an IMU, placed in a waistband, and shoes with FSR [88]. A few system characteristics, as power supply were adapted for this study and will be presented in the following section.

4.2. Global Architecture

The system comprises a processing unit, an IMU to achieve inertial data from user' lower trunk, gait shoes equipped with FSR to detect when the foot contact on the ground, and lastly a computer. The computer supplies the system and saves all acquired data that will be analyzed later. A schematic of system overview is shown in Figure 11. The application initiates or stops the data collection. It is important to note that the defined sampling frequency was 100 Hz for all data sensors.

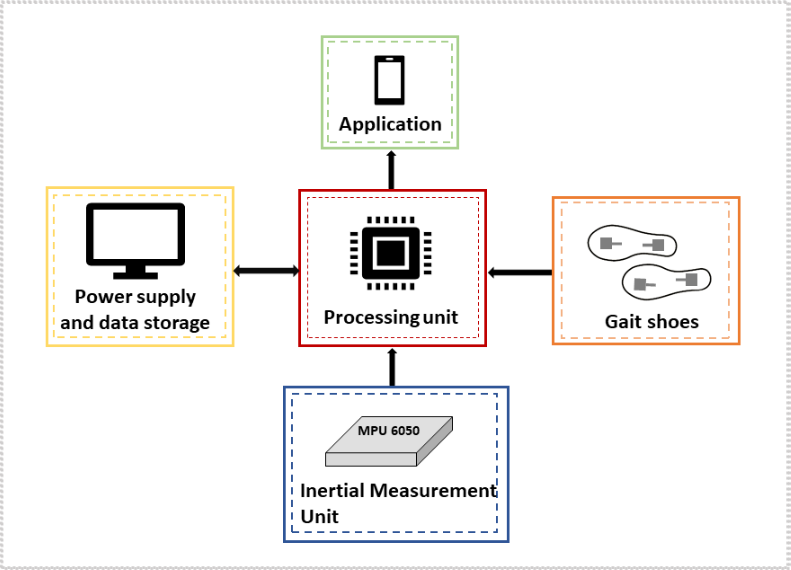


Figure 11 – The system architecture overview.

4.2.1 Sensors Overview

This section will present the characteristics of the sensors, namely the IMU and the FSR (Figure 12 and 13, respectively), and why some characteristics are important for the work developed.

The MPU-6050 (Motion Processing Unit) has a 3-axis gyroscope and 3-axis accelerometer. The small sensor dimension (4x4x0.9mm) is a positive aspect at one time it will be positioned on the user. This sensor enables to program the full-scale range of accelerometer and gyroscope sensors. Selecting the limit of which the accelerometer and gyroscope can read is a central aspect to be considered since the wrong choice can influence the data measured. In the case of the accelerometer, the full-scale range selected was ±8g and the gyroscope full-range scale chosen was 2000°/s which is the closest use in the study [89].

Figure 12- MPU 6050.

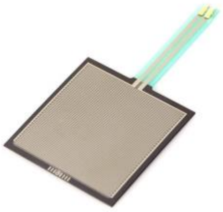


Figure 13 – Force sensor resistor.

It is necessary to apply a calibration procedure in order to minimize the measurement errors. In this regard, the technique used consists of placing the sensor in a horizontal table and recording the acceleration axis parallel to gravitational force for six different positions, as shown in Figure 14 [89]. In this study, the acceleration in each position was stored during 10s and the mean values of each axis was calculated to achieve the maximum and the minimum value to adjust the range.

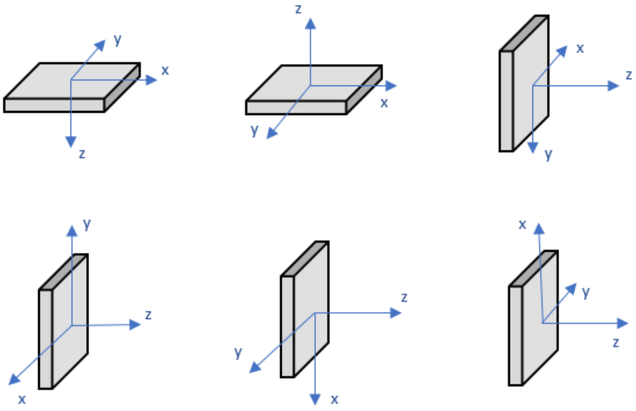


Figure 14 – The six position for accelerometer calibration.

Gyroscope calibration consisted of placing the sensor in one of the six positions and record samples during 10s. Later the mean of samples for each axis was calculated, and the offset values was subtracted to further collected samples.

From the signals acquired with the four square FSR, it is possible determine when heel or toe or both are in contact with the ground through the variation observed in the signal. These sensors have low weight and they are thickness, which means they do not influence the user's gait.

4.3 Data Acquisition Methods

This study was conducted with ten healthy young subjects (five females and five males; 66.5±11.32kg; 1.69±0.11m; 25±1.61 years). Before the tests none of the participants reported any disorder that could influence the gait pattern or compromise the tests. All participants agreed to perform the tests after the explanation of the whole procedure.

The first step performed before start record data from the participants consisted in collecting the data to calibrate the sensors as described in the previous section (once a day). After the conclusion of the first step, the participant was instrumented. Thus, the shoes and waistband were placed on the subject and also a strap on the sensors to fix better. The MTw Awinda (Xsens system) was a complementary system used during the tests for human motion tracking. This system is based on IMU, and in this particular case, it was applied the lower body configuration as represented in Figure 15. The Xsens system IMU that should be placed in the pelvis area was placed as close as possible to the waistband IMU. In order to guarantee that the information collected by the two systems is as reliable as possible and, consequently, it can be compared. Afterward prepared the participant the Xsens system was calibrated. It is important to mentioned that the waistband and the gait shoes described previously was synchronized with the Xsens system with a sampling frequency of 100 Hz.

For the experimental tests, the participants use a rollator to walk as shown in Figure 15. Four different activities were executed three times for two different velocities. It was asked the participant to walk at a comfortable velocity for him and another one very slow. These two velocities were selected to have more variability of data. Furthermore, users of ASBGo SW usually walk at a slow velocity, hence this is such a critical velocity.

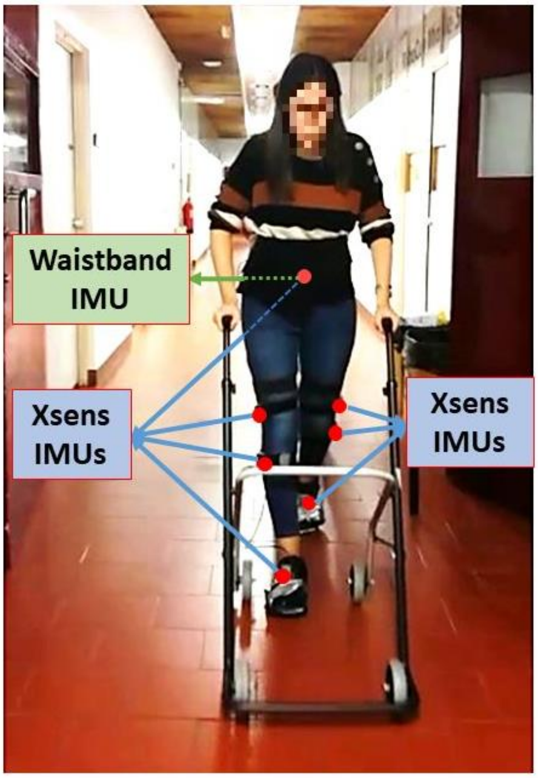


Figure 15 - IMU placement on the user's body and the rollator during the test.

The different activities performed by the participants are described in Table 9. For activities 2, 3 and 4 (Table 9), the subject walked during a certain time, undefined, and then simulated a NF for forward, right and left, respectively. The moment that the subject simulates the NF was determined either by an audible signal or the volunteer himself determined the instant. The selected approach was determined based on the volunteer's best performance, i.e. the one in which the volunteer was able to get closer to a real NF. A total of sixty trials were recorded for each activity. However, a few trials were removed because sensors failed, i.e., the data was not recorded correctly.

Table 9 – Activities description simulated with the rollator for a comfortable and slow velocity

Number	Activity Description
1	Walking straight for ten and a half meters with a rollator
2	Walking straight and simulate a forward NF with a rollator
3	Walking straight and simulate a right NF with a rollator
4	Walking straight and simulate a left NF with a rollator

4.4 Software Methodology Overview

Once the data acquisition is completed, it is necessary to process data in order to be further used for NF, INF and gait event detection. The data pre-processing involves data calibration, and filtering the data gathered with a high-pass and an exponential filters. Then, these data were labeled according to the intended purpose, i.e., to detect a NF or detect an INF or detect the gait events. Following, several characteristics were computed from the inertial data. After the dataset with all features was obtained, three different types of feature selection methods were applied. Finally, machine learning algorithms were implemented in order to determine whether a NF or an INF occurs or to detect gait events. A general overview of the methodology implemented in chapters, 5, 6 and 7 is shown in Figure 16.

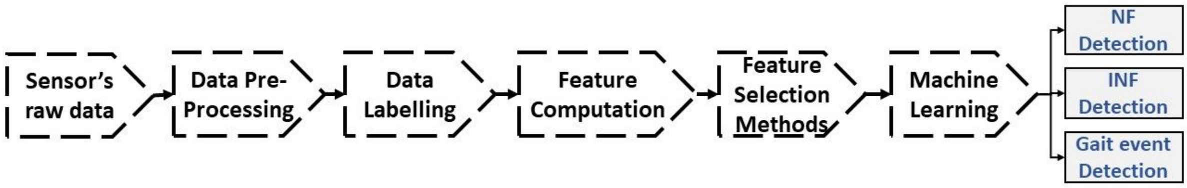


Figure 16 - General overview of the implemented methodology.

4.4.1 Data Pre-Processing and Feature Computation

The pre-processing data is a crucial step, since it allows the elimination of irrelevant information, retaining the essential one. In this respect, the accelerometer and gyroscope data were calibrate as mentioned previously. Thereafter, different filters were applied for each axis. In this study, one of the main goals is to develop a strategy that can be applied in real-time, thus, it is crucial not introduced delayed in data recorded. In this respect, it is desirable that the filters applied only depend on the current sample and the previous samples.

When a rotation is performed in the coronal plane, roll, and sagittal plane, pitch, the value of gravity acceleration measured change, so it is necessary compensated roll/pitch motion. Windau et al. [89], to obviate this problem, estimate the roll and pitch angle and then applied a rotation matrix to obtain the acceleration data measured in the normalized coordinate system. Therefore, in this dissertation, the roll and pitch angles were obtained based on a Kalman filter because the gyroscope is susceptible to drift, and this makes the angle obtained by integration unreliable. This filter is based on fusion sensor, it uses the gyroscope and accelerometer data to achieve a more reliable angle value [90]. After calculating the roll and pitch angle, a rotation matrix as described in [91] was implemented. After the aforementioned steps, a high-pass filter was applied to remove the gravity component ($9.8m/s^2$). In sum, for vertical acceleration axis, roll and pitch compensation were applied and then a high pass filter [92] represented by Equation (1):

$$y(n) = [x(n) - x(n-1)] + 0.995y(n-1)] \qquad (1)$$

Where y(n) is current output sample and x(n) current input sample. For anteroposterior and mediolateral axis, first was applied a high pass filter in order to eliminate the DC component and then an exponential filter represented by Equation (2) was used to smooth the signal.

$$y(n) = \alpha . x(n) + (1 - \alpha) . y(n-1) \qquad (2)$$

Where, α is the smoothing factor (0< α<1), y(n) current sample filtered, x(n) current sample and y(n-1) the previous sample filtered. In order to choose the best factor, the acceleration signal obtained with the waistband IMU was compared with Xsens lower back IMU based on Root Mean Square Error (RMSE) for both directions, anteroposterior and mediolateral. Only the exponential filter was used for each axis of the gyroscope (vertical, anteroposterior and mediolateral directions). The best factor was also chosen through RMSE between the recorded waistband and Xsens data. The results are presented in Table 10. It is important to highlight that just the data collected from the IMU placed on the waistband, with the participants walking forward, were used to compare with the Xsens acceleration and gyroscope data. The inertial signals obtained for three directions (vertical, mediolateral and anteroposterior) with Xsens and the IMU of waistband after filter process are shown in Figure 17 and Figure 18. The last step accomplished of pre-processing data was the data normalization based on the height of the participants.

Table 10 – RMSE results achieved with the Xsens signal and the signal recorded after the filtering process

Axis	**RMSE mean**	**Exponential factor**
Vertical acceleration	0.1935	-
Mediolateral acceleration	0.4409	0.3
Anteroposterior acceleration	0.5488	0.3
Vertical angular velocity	2.977	0.8
Mediolateral angular velocity	3.004	0.8
Anteroposterior angular velocity	3.8756	0.7

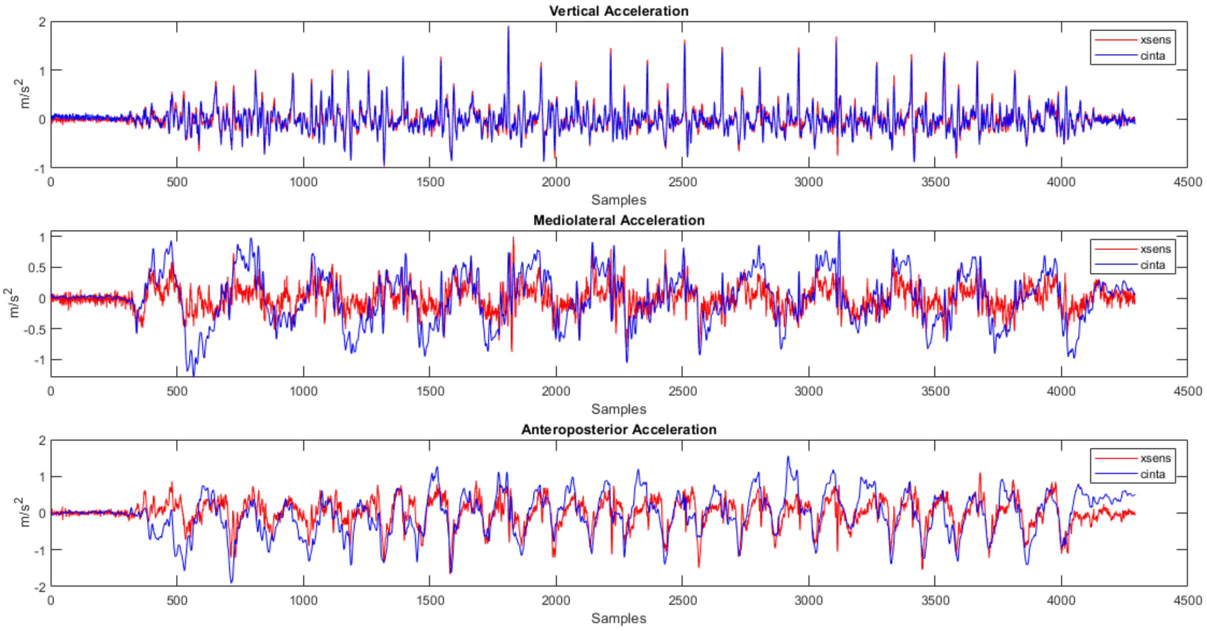


Figure 17 - Comparation between the Xsens signal with the waistband filtered signal from the three axis accelerometer data.

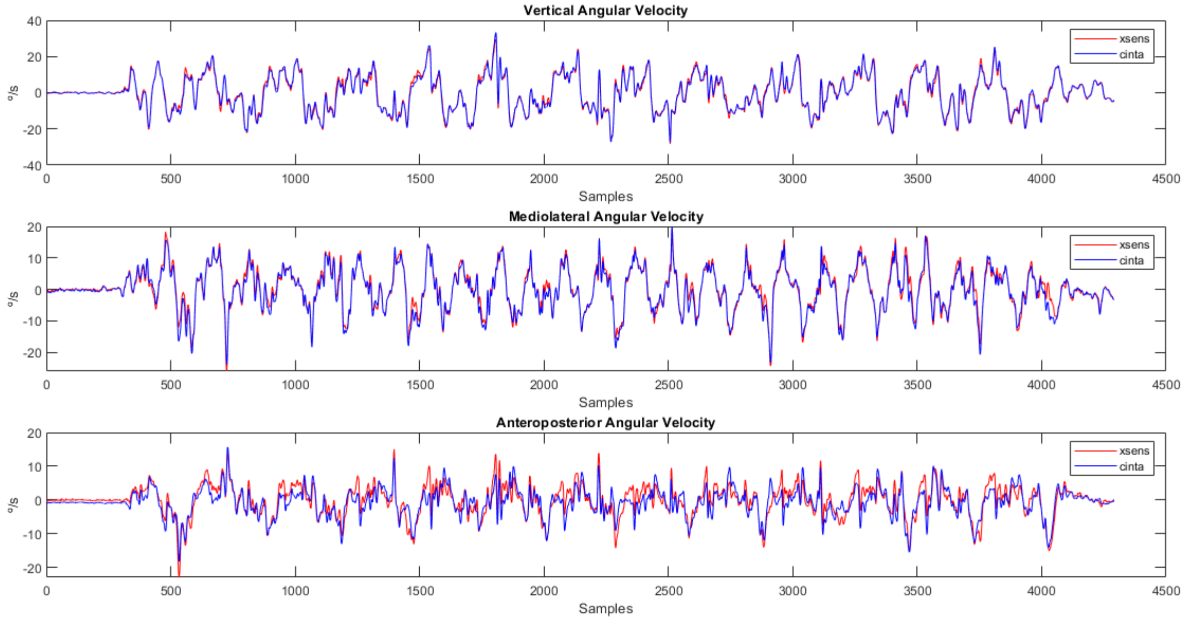


Figure 18 - Comparation between the the Xsens signal with the waistband filtered signal from the three axis of the gyroscope data.

Once the data have been filtered, they were labelled in 4 different ways as described in Table 11. For data labelling, the acceleration and angular velocity signals in the three axis, the FSR signals and acceleration sum vector magnitude signal ($\sqrt{x^2+y^2+z^2}$) were used, except for the gait event labelling. For this particular case, a thresholds-based algorithm was developed using only FSR signals. For each situation described in Table 11, more details will be given in the next chapters.

Concerning data labelling the first gait step was excluded because it was observed that FSR signals in some trials were not in accordance with what was expected. This may have happened because the subject while still in stance position is not entirely motionless and exerts more/less pressure in one of the FSR.

Table 11 - Description of different labelled signals

Number	Description
1	Normal walking and NF
2	NF direction (Forward, Right and Left)
3	Normal walking and INF
4	Gait events (TO and HS for each foot)

After the data pre-processing and data labelling were completed, several features were computed through acceleration and angular velocity data. These features were selected based on several studies presented in Appendix 1 Table 35. In addition to the features computed, other features were used, such as the FSR signals also presented in Appendix 1 Table 35. In this table only one extra feature is missing which is computed through the FSR data, described in Chapter 6.

One of the goals of this work is to develop an algorithm that can be implemented in real time. In this sense, all computed features only use the current or previous samples. The window size selected to compute certain features was 50ms with the exception of velocity and displacement that will be addressed later. It is important to note that some features mentioned in Appendix 1 were adapted for this work.

In order to calculate the three-axis velocity and displacement, it is necessary integrate one and two times the acceleration data, respectively. The steps followed to calculate the displacement were:

1. Subtract acceleration means obtained with a specific window length
2. Integrate acceleration to velocity by the trapezoidal method
3. Subtract velocity mean obtained through a specific window length
4. Integrate velocity to displacement by the trapezoidal method
5. Subtract displacement mean obtained through a specific window length

The window length for each axis was defined based on similarity between Xsens displacement recorded and waistband IMU displacement computed. The window sizes were 130, 80 and 60 samples for vertical, mediolateral and anteroposterior axis, respectively. It should be noted that it was necessary to apply a high pass Butterworth filter to Xsens data to compare the signals. The comparison between the Xsens displacement signals and the computed displacement signals for the three axis is illustrated in Figure 19.

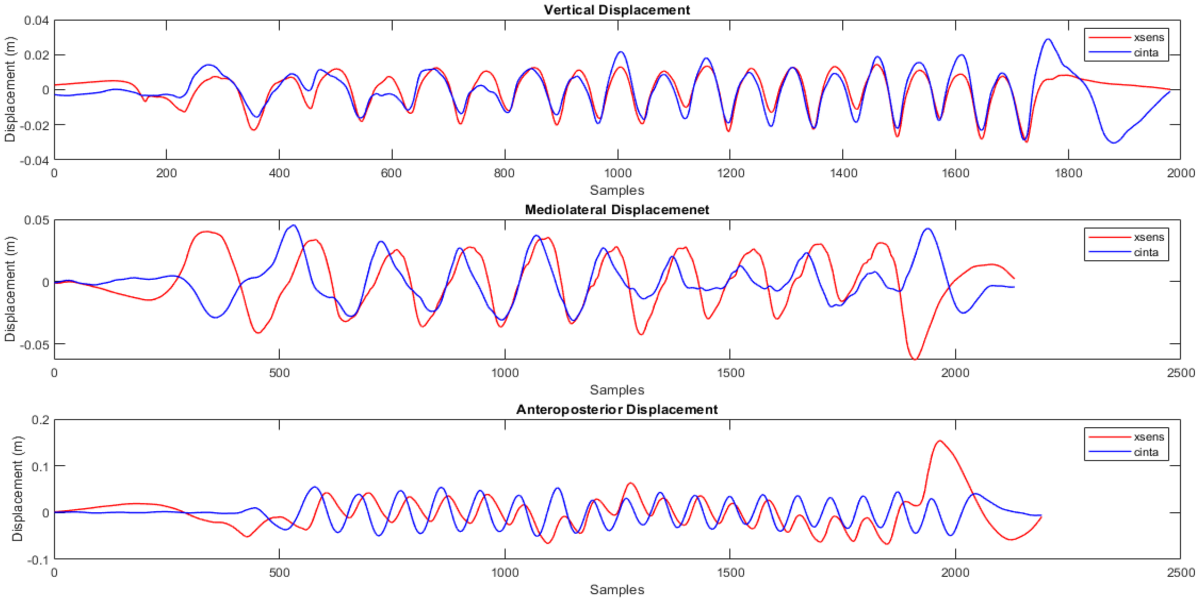


Figure 19 - Comparation between Xsens signals and waistband signals from the three axis of pelvis displacement.

In the subsequent section the following steps will be explained in more detail. More specifically, the feature selection methods and the machine learning algorithms used in this dissertation.

4.4.2 Machine Learning Approach

Concerning approaches to learning algorithms, there are three distinct modes: supervised learning, unsupervised learning, and reinforcement learning. In a supervised learning approach, the input data and the wanted outputs are specified (labelled data), and the objective is to build a model that learns from this data and is able to predict the correct output for unseen data. The indicated method can be subdivided into two main categories: classification and regression. In the case of unsupervised learning, it is provided only the input data without the required response [93]. Reinforcement learning is a type of machine learning that learns by interacting with the environment. This learning is based on the rewards and punishment of the actions performed [94]. In this work, it will be implemented supervised learning algorithms for classification problems. The main goal is to predict a class label from the input features provided to different classifiers and study which is the best classifier for different specific targets (NF detection, INF detection, and gait event detection).

The first two steps prior to study the classifiers algorithms are the normalization data and feature selection methods. Normalization data allows putting the data within a range, for instance, 0 and 1. This point is important for neural networks and algorithms concern to distance, such as KNN algorithm [95], [96]. Since the KNN is one of the algorithms studied in this work, the normalization data was the first step. In this case, the min-max normalization in the range [0 1] was implemented (Equation 3).

$$x' = \frac{x - min_A}{max_A - min_A} \quad (3)$$

Where x is the original value, min_A and max_A are the maximum and minimum value of A, respectively, and x' is the normalized value.

The feature selection methods were implemented in order to rank the features and improve the machine learning algorithm's performance. The quality of the training data is essential to achieve a model that is able to classify and generalize well. Having a large number of features can impair the model's training process if they have irrelevant or redundant information as well as noise [97]–[99]. That is why it is so significant to implement methods that allow ranking the features according to their relevance in order to reduce the number of features needed to train the model efficiently. The feature selection methods can be divided into three main methods: filter, wrapper and embedded [100].

Regarding the three methods mentioned, the filter is the least computationally expensive, in contrast to other methods, it does not use any classifier. This technique is based on the information of each feature individually. The wrapper methods use a classifier algorithm that assigns a score to the dataset according to the performance of the classifier. Of the three methods, this is the most expensive computationally (Table 12). Finally, in the embedded method, feature selection is part of the learning process of the classifier, i.e., feature selection is built during the training of the model [98], [100]. As in the classifiers, feature selection methods can also be supervised or unsupervised.

Table 12 - Use or not of classification algorithms and computational cost of feature selection method

Method	Classifiers	Computational cost
Filter	x	+
Wrapper	✓	+++
Embedded	✓	++

For this work, it was selected the Relieff, the MRMR, and the PCA methods for ranking feature dataset. The Relieff method is prone to redundant features but can deal with noise datasets. The MRMR identifies relevant and redundant features [99]. The first methods are supervised and the PCA is unsupervised, and all of them are filter methods. This type of method was chosen since they are less computationally expensive.

Different machine learning algorithms can be implemented for classification. Support Vector Machine (SVM), DT and KNN are some examples of supervised machine learning algorithms widely employed.

The KNN classifier is a simple learning algorithm. The construction of the model consists of saving the training data. The class assignment to an unknown point is based on the distances from that point to closest data training (k-neighbours). Thus, it is necessary to define the number of k-neighbours that will contribute to the decision. For example, if k = 1, the assigned class will be the same as the closest neighbour. In case the k is greater than 1, the assigned class will be the majority observed in the k-neighbours. In this model, the appropriate value of k is crucial since a k with a small value is very susceptible to data noise and the model is highly complex. However, with high k value the model is not so complex but can comprise several points of different classes [101], [102]. When exists different classes in the k-neighbours, it can be useful to assign a weight accordingly to the distances between the k-neighbour and the unknown point. In this way, closer neighbours would have a higher weight in the attribution of the class [103].

The DT algorithm is based on questions. The DT is formed by nodes that connect to each other through branches. The process starts at a node called *root* that represents all the data. Each node will divide the instances according to the values of the input attributes. The intermediate nodes are called *test*. The last node that does not connect to any additional node is called *leaf* and represents the final decision (class label). If the division of the nodes occurs successively until it reaches all leaves pure, we will have a more complex model and prone to overfitting the training data. There are two approaches to controlling complexity, namely, pre-pruning and pruning. The pre-pruning consists of stopping the formation of the tree previous (for instance, reduce the number of leaves). Pruning consists of removing nodes that are not crucial [101], [104].

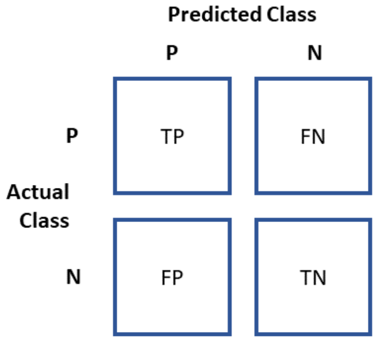
The SVM classifier tries to find a hyperplane in the n-dimensional space that best divides the classes. This hyperplane has an associated margin that is formed by only a few points of the training data (support vectors). Depending on the dataset a linear model may not be able to separate the classes in the best way. In this regard, nonlinear features can be added to represent the data, i.e., transform the input data into a higher dimension features space. Therefore, classes that were not linearly separable in initial space in this new space are already possible. For this purpose different kernels can be used, for example, the polynomial or the Gaussian kernel [101].

Other algorithms can be implemented for classification problems as well as Ensemble learning algorithms and LDA. Ensemble learning model consists of combining several classifiers to achieve better performance than would be achieved with any single model [105]. LDA algorithms are based on statistical data properties performed for each class. This model assumes that each class has Gaussian distribution and has same variance/covariance. The classification of new data is performed based on the probability of the data belong to a certain class. There is some extensions of LDA, such as Quadratic Discriminant Analysis (QDA) and Flexible discriminant analysis [106].

After building the model, it is necessary to evaluate its performance with unseen data, because during the model training it may occur overfitting or underfitting. One of the techniques used to evaluate the model performance is k-fold cross-validation. This method divides the data randomly in k folds. The classifier is trained and tested k times with different subsets. This means that k-1 folds are used to training the model and one fold for test and this is repeated until all folds have been used for testing [93], [106]. Throughout this dissertation a k=5 was always used.

Generally, the evaluation data are represented in a confusion matrix as shown in Figure 20. The row lines correspond to the actual classes and the columns represent the predicted class achieve by the classifier. The True Positives (TP) and True Negatives (TN) correspond to the correct classification and the False Negative (FN) and False Positive (FP) the misclassification. Different metrics can be calculated from the confusion matrix.

Some metrics usually employed are the Accuracy (ACC), Precision (PREC), Sensitivity (SENS) Specificity (SPEC), F-1 Score and Mathews Correlation Coefficient (MCC). These metrics are described thereafter.



		Predicted Class	
		P	N
Actual Class	P	TP	FN
	N	FP	TN

Figure 20 - Confusion matrix example.

ACC is the fraction of correct prediction dived by the number of total samples (Equation (4)). This metric is not the best metric to evaluate imbalanced data that is to say when has a significant difference between the number of instances of each class. In this case, the classifier can predict a high number of the majority class and achieve a good ACC. In case of imbalance data, other metrics can be used such as PREC and SENS [107].

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \tag{4}$$

PREC is the fraction of TP divided by all positive (TP and FP). This metric measures the proportion of predicted positives that were correctly classified (Equation 5).

$$\mathrm{PREC} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{5}$$

SENS is the number of TP divided by the sum of TP and FN (Equation 6).

$$\mathrm{SENS} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{6}$$

SPEC is the number of TN dived by the sum of TN and FP. This metric measures the proportion of actual negatives that were correctly classified (Equation 7).

$$\mathrm{SPEC} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} \tag{7}$$

F1 -score is the harmonic mean of precision and sensitivity. This metric can be better for imbalanced data than ACC [93] (Equation 8).

$$\mathrm{F1} - \mathrm{score} = \frac{2 * (\mathrm{SENS} + \mathrm{PREC})}{\mathrm{SENS} + \mathrm{PREC}} \tag{8}$$

The last metric presented is MCC that takes into account all confusion matrix (Equation 9).

$$\mathrm{MCC} = \frac{\mathrm{TP} * \mathrm{TN} - \mathrm{FP} * \mathrm{FN}}{\sqrt{(\mathrm{TP} + \mathrm{FP})(\mathrm{TP} + \mathrm{FN})(\mathrm{TN} + \mathrm{FP})(\mathrm{TN} + \mathrm{FN})}} \tag{9}$$

Chapter 5 – Near Fall Detection

The main goal of this chapter is to detect NF events and the direction of the NF when using a rollator. Through the analysis of the literature, it is possible to conclude that there is an evident concern to prevent a fall situation. Many approaches were developed based on wearable and non-wearable systems [12], [14]. Regarding smart walkers different methodologies were found in literature in order to prevent a fall by stopping the walker when a dangerous situation is detected [75]–[82]. However, none of the articles found in the state of the art presents a strategy based only on IMU (Lower trunk) and FSR (shoes) signals to detect a NF and the NF direction with a smart walker. Nonetheless, these sensors have been studied by other researchers with the intention to detect a fall or even to distinguish the fall direction without a smart walker. It must be highlighted that the first advances in the identification of a NF as well as the direction, in this work, aim an implementation of a strategy to avoid the dangerous situation in the future.

In this chapter, the whole process developed to detect a NF and its direction will be presented. For this purpose, the device described in Chapter 4 was used to collect the data to be able to implement the strategy described throughout the next section. The methodology followed, the results achieved, and their critical discussion are presented below.

5.1. Methods and Materials

In this chapter, it was performed two different studies one that consists of differentiating a NF from normal walking (Case 1), and another to distinguish the NF direction (Case 2). For this purpose, in the first place was necessary to collect the experimental data and perform the pre-processing as described in Chapter 4. Then, it was necessary distinguish between normal walking and NF, without considered the direction of the NF (data labelling, Case 1), and to distinguish the NF direction (data labelling, Case 2). Following this, several features were computed (Appendix 1 Table 35) for each case aforementioned. Subsequently, different feature selection methods were applied. Finally, distinct classifiers were implemented in order to detect a NF (case 1) and its direction (case 2). Case 1 is a binary classification problem and case 2 is a multiclass classification problem. In the second case, a distinction was made between NF to forward, right and left direction. In this study, the simulated backward NF were not performed since they presented a high risk of injury to the participants. Figures

21 and 22 depict an overview of the methodology implemented for case 1 and 2, respectively. Each case will be described in more detail below.

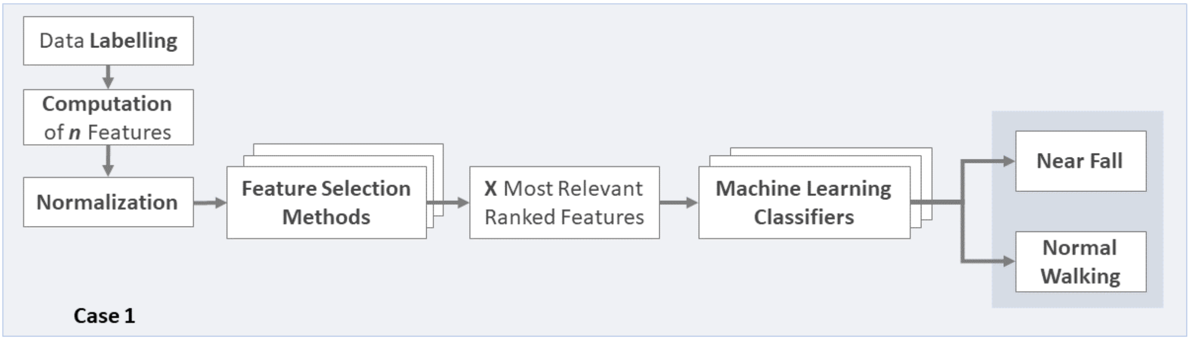

Data Labelling
Computation of *n* Features
Normalization
Feature Selection Methods
X Most Relevant Ranked Features
Machine Learning Classifiers
Near Fall
Normal Walking
Case 1

Figure 22 - Schematic overview of the different steps performed to discriminate the NF from normal walking.

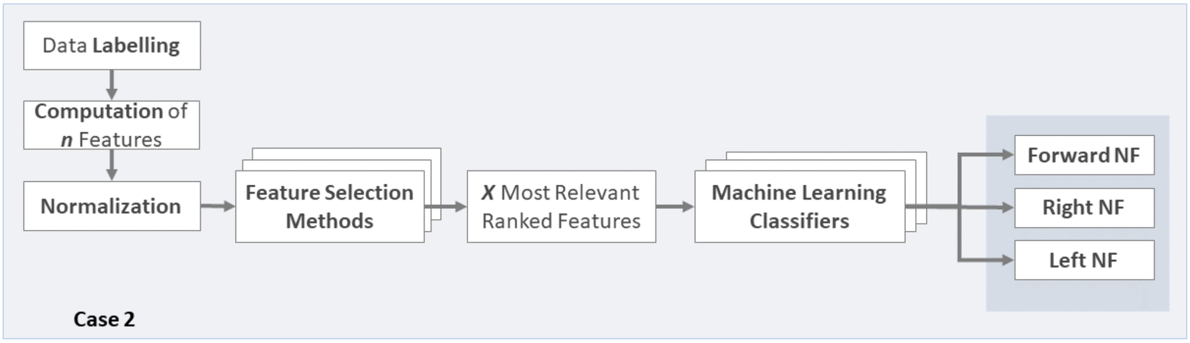

Data Labelling
Computation of *n* Features
Normalization
Feature Selection Methods
X Most Relevant Ranked Features
Machine Learning Classifiers
Forward NF
Right NF
Left NF
Case 2

Figure 21 -Schematic overview of the different steps performed to discriminate the NF direction.

5.1.1 Case 1

The distinction between NF and normal walking for each trial was determined based on the following signals: the acceleration signals in the three axes; the angular velocity signals in the three axis; the acceleration sum vector magnitude signal; and the four FSR signals. After labelling the data, features were estimated and normalized, and the dataset was divided between training and test data by, approximately, 70% for training and 30% for testing.

Three distinct feature selection methods were applied in data training to rank the features, namely, MRMR, Relieff and PCA (Figure 23). It was required to label the data before implementing the MRMR and Relieff algorithms, because these are supervised methods and, therefore, need labelling data. Only a specific number of features (***X***) were used for the next steps to reduce computational time. This number was defined based on PCA. First the number of Principal Components (PC) explaining at least 70% of the variance was estimated, afterward the values of these PC were summed and normalized. Then, the features where the sum of PC is greater than 1/ (number of features) were selected.

After concluding which features should be considered and, thus, obtaining the training dataset to be used, seven different machine learning algorithms were applied to build the models. In particular, LDA, QDA, DT, KNN with equal, inverse and squared inverse distance weight function and Ensemble algorithms. In order to know how many features were necessary to reach the best result, a progressive analysis was executed. This means that was built a model with the first most relevant feature, then with the first two most relevant features, this iteratively until the number of features be equal to ***X***.

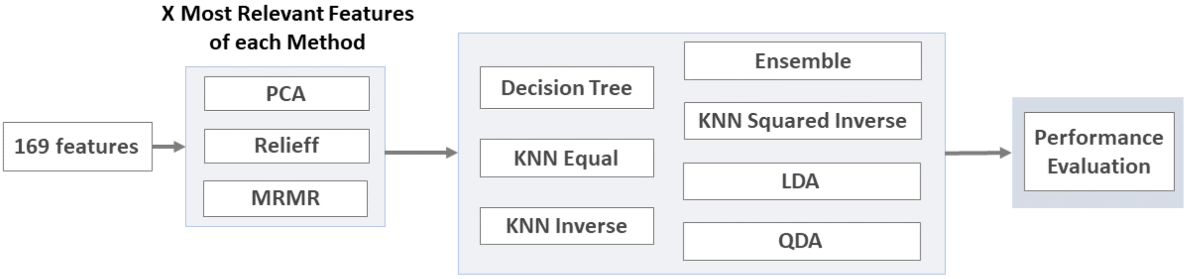


Figure 23 – Methodology implemented for building the machine learning models to detect a NF.

Thus, 21 different models with the best feature set were built. Subsequently, the models were tested with unseen data to understand how reliable the models are (Figure 24). The performance of the models was evaluated based on the metrics mentioned in Chapter 4.

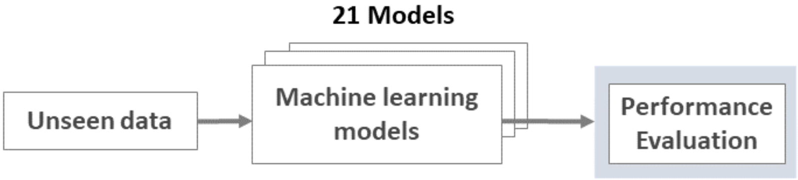


Figure 24 - Methodology implemented for evaluating the model performance with unseen data to detect a NF.

In order to improve the outcomes, the hyperparameter optimization was performed for the three models with the best performance. In case of KNN, different ks were tested for Euclidean distance. For DT, the 'auto' hyperparameter optimization option provided by MATLAB was implemented. This option allowed to optimize the minimum number of leaf node observations. The Ensemble model hyperparameters was also optimized with 'auto' hyperparameter optimization option provided by MATLAB. In order to optimize the number of ensembles learning cycles, learning rate for shrinkage and the minimum number of leaf node observations.

Since the dataset is a case of unbalanced data, approximately 85% of one class and 15% of the other class, two other approaches were implemented. In the first approach the minority class data was duplicated. In the second approach different misclassification costs were tested in order to decrease the number of samples correspond to the NF that are classified as normal walking.

The last step consisted in applying a post-processing algorithm to the data predicted by the best model built throughout the process described above. This post-processing aims to reduce the number of FP. A sample is only classified as NF if a given consecutive number of previous samples has also been classified as NF. A window size from 1 up to 25 samples was tested in order to achieve the best result.

5.1.2 Case 2

The case 2 is a multiclass problem, since it is intended to classify three different directions (forward, right and left). The first step consisted of labelling the data with three different classes based on the signals mentioned for case 1. Afterwards, the features were calculated and normalized between 0 and 1. The training dataset represented, approximately, 70% of entire dataset and the test dataset 30%. The feature selection methods (PCA, MRMR and Relieff) were applied in this new dataset to get a new ranking. Before implementing the classification algorithms, the PCA was applied in order to define a maximum number of features to use in the next steps, as described in the previous case.

The SVM with three different kernels, linear, Gaussian and polynomial were used in addition to the models applied for Case 1. In this case, particularly, each class represent, approximately, 32%, 35% and 33% of the entire dataset. At the end of this process, 3 models were obtained for each classifier (Figure 25).

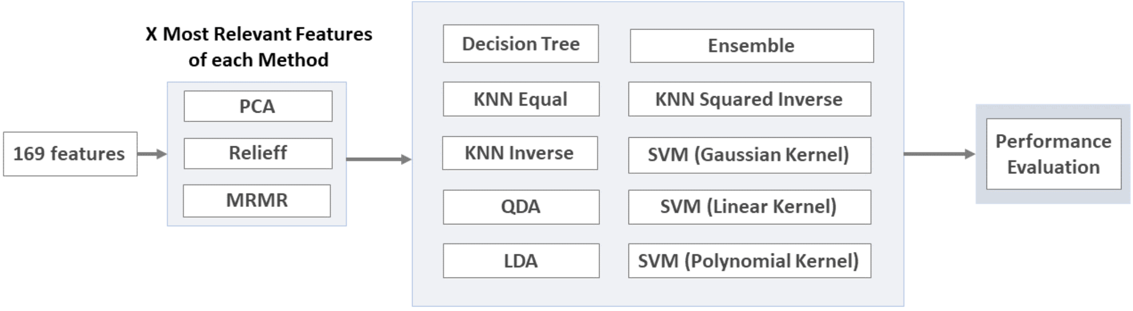


Figure 25 – Methodology implemented for building the machine learning models to discriminate the NF direction.

The 30 built models were tested with unseen data as despicted in Figure 26. In order to understand how it actually behaves. After the analysis of all the models performance, the hyperparameters of the model that best distinguished the NF directions were optimized. For this, it was used the option of hyperparameters optimization, 'auto', provided by MATLAB.

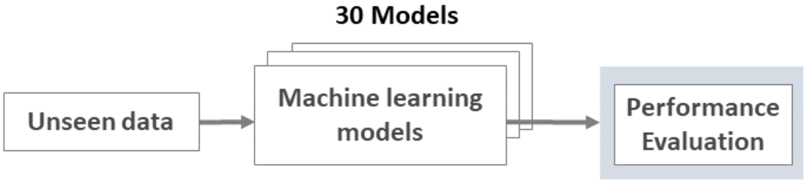


Figure 26 - Methodology implemented for evaluating the model performance with unseen data to discriminate the NF direction.

5.2. Results

5.2.1 Case 1

After feature computation and the application of feature selection methods, the number of PC that explain at least 70% of the variance was estimated, which corresponds to 13 PC as depicted in Figure 27. Based on the information from the 13 PC, 22 features were selected. However, sixty features were used, which means that they are almost three times more features. The sixty most relevant features ranked by the MRMR, Relieff and PCA are presented in Appendix 2 (Table 36).

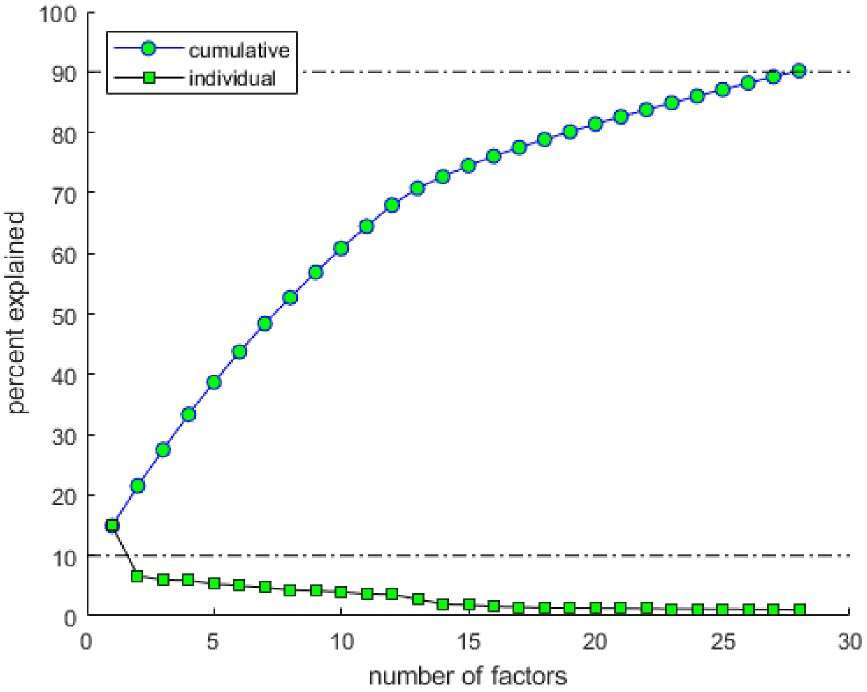


Figure 27 – Screen plot of PCA when using the dataset to detect a NF.

Subsequently, several models were built (LDA, QDA, DT, Ensemble, KNN Equal, KNN inverse, and KNN squared inverse) with different feature sets. The best cross-validation results for each combination of feature selection methods and classifiers algorithms are shown in Appendix 2 (Tables 37, 38 and 39) for best feature sets found. For MRMR and Relieff selection methods, the best performances were achieved with DT, Ensemble, and KNN with different weight distance functions. The LDA and QDA presented the worst performances, being the best MCC = 51.41% for LDA classifier with MRMR method. After selected the best number of features for each model, the models were tested with unseen data. The results achieved are presented in Table 13. The Ensemble algorithm was the classifier that achieved the best performance ACC= 95.18%, SENS=71.63%, SPEC=99.33%, PREC= 94.96%, F1-score= 81.66% and MCC= 79.99%.

Table 13 – Evaluation performance results achieved for all combinations of feature selection methods, number of features and machine learning models when tested with unseen data to detect a NF

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1-score	MCC	Number of Features
Relieff	LDA	87.19	57.48	92.42	57.19	57.33	49.79	23
	QDA	89.17	66.95	93.08	63.03	64.93	58.57	23
	KNN Equal	92.25	62.27	97.53	81.64	70.65	67.08	59
	DT	87.67	68.25	91.09	57.44	62.39	55.36	20
	Ensemble	93.72	71.40	97.65	84.25	77.29	74.01	41
MRMR	LDA	88.29	49.39	95.15	64.21	55.83	49.78	10
	QDA	90.58	61.42	95.72	71.65	66.14	60.95	3
	KNN Equal	93.91	68.66	98.36	88.05	77.15	74.46	60
	DT	91.05	72.67	94.29	69.14	70.86	65.61	45
	Ensemble	**95.18**	**71.63**	**99.33**	**94.96**	**81.66**	**79.99**	**51**
PCA	LDA	87.73	12.27	44.75	95.30	52.21	46.24	23
	QDA	87.63	12.38	47.25	94.74	53.35	46.88	23
	KNN Equal	89.04	49.31	96.04	68.69	57.41	52.25	60
	DT	84.67	64.19	88.28	49.10	55.64	47.19	37
	Ensemble	87.36	59.12	92.33	57.59	58.34	50.90	30

Since the best performance with unseen data were obtained by the DT, KNN Equal, and the Ensemble classifiers when using the MRMR feature section method, the hyperparameters of these models were optimized. Concerning the DT classifier, the minimum leaf size observation was the hyperparameter optimized, and the best value achieved for this hyperparameter was 2 (Appendix 2 Table 40). It should be mentioned that the value used without optimization was 1 (default value). The DT model performance with hyperparameter optimization is shown in Table 14.

Table 14 – Evaluation performance result of the DT model with the hyperparameter optimized when tested with unseen data to detect a NF

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1-score	MCC	Number of Features
MRMR	DT	91.38	72.57	94.69	70.62	71.60	66.53	45

Respecting KNN Equal, different values of k were tested, namely, 1,4,8,9,11,13,15 and 20. Through cross-validation data the best result was achieve with k=4 (Appendix 2 Table 41). The model performance with unseen data for k=4 is presented in Table 15. It should be noted that the default value of k was 1 (model performance presented in Table 13).

Table 15 – Evaluation performance result of the KNN Equal model with the hyperparameter optimized when tested with unseen data to detect a NF

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1-score	MCC	k	Number of Features
MRMR	KNN Equal	94.04	66.69	98.86	91.15	77.03	74.89	4	60

Concerning Ensemble classifier, the result of hyperparameters optimized is shown in Table 16. The overall metrics obtained for Ensemble classifier with 51 features and hyperparameters optimization are presented in Table 17.

Table 16 – Ensemble hyperparameters values optimized to detect a NF

Hyperparameters	Value
Ensemble Aggregation Method	Bag
Number of Learning Cycles	498
Learn Rate	NaN

Table 16 - Continued

Hyperparameters	Value
Minimum Leaf Size	1

Table 17 – Evaluation performance result of the Ensemble model with the hyperparameters optimized when tested with unseen data to detect a NF

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1- score	MCC	Number of Features
MRMR	Ensemble	95.17	71.76	99.29	94.68	81.65	79.93	51

In this study, the proportion of one class in the dataset is much higher than the other (unbalance data). In this regard, the first approach tested was the oversampling of training data. In other words, the data of minor class were duplicate. The metrics results are shown in Table 18. In this particular case, the MCC (79.13%) and F1-score (81.13%) decreased slightly concerning initial performance obtained. However, the SENS increased, approximately 0.4%. The cross-validation result is shown in Appendix 2 (Table 43).

Table 18 – Evaluation performance result of the Ensemble model with data oversampling when tested with unseen data to detect a NF

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1-score	MCC	Number of Features
MRMR	Ensemble	94.98	72.02	99.03	92.89	81.13	79.13	51

Different costs (5,3 and 10) were tested for FN. The results are shown in Table 19 for unseen data and the cross-validation results are shown in Appendix 2 (Table 44).

Table 19 – Evaluation performances results of the Ensemble model with the 51 most relevant features ranked by the MRMR method for different misclassification costs when tested with unseen data to detect a NF

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1-score	MCC	Cost
MRMR	Ensemble	95.00	71.95	99.05	93.02	81.14	79.16	3
		94.99	72.03	98.93	92.19	80.87	78.77	5
		94.53	71.76	98.54	89.64	79.71	77.23	10

Through Figure 28, it is possible to observe that from 15 features, the MCC, F1-score, SENS, and ACC do not increase significantly with cross-validation data. It was analysed the model performance when built with fifteen features and tested with unseen data. The overall metrics achieved were ACC= 94.90%, SENS= 70.88%, SPEC=99.14%, PREC= 93.53%, F1-score = 80.64% and MCC= 78.77% (Table 20). The cross-validation results are shown in Appendix 2 (Table 45).

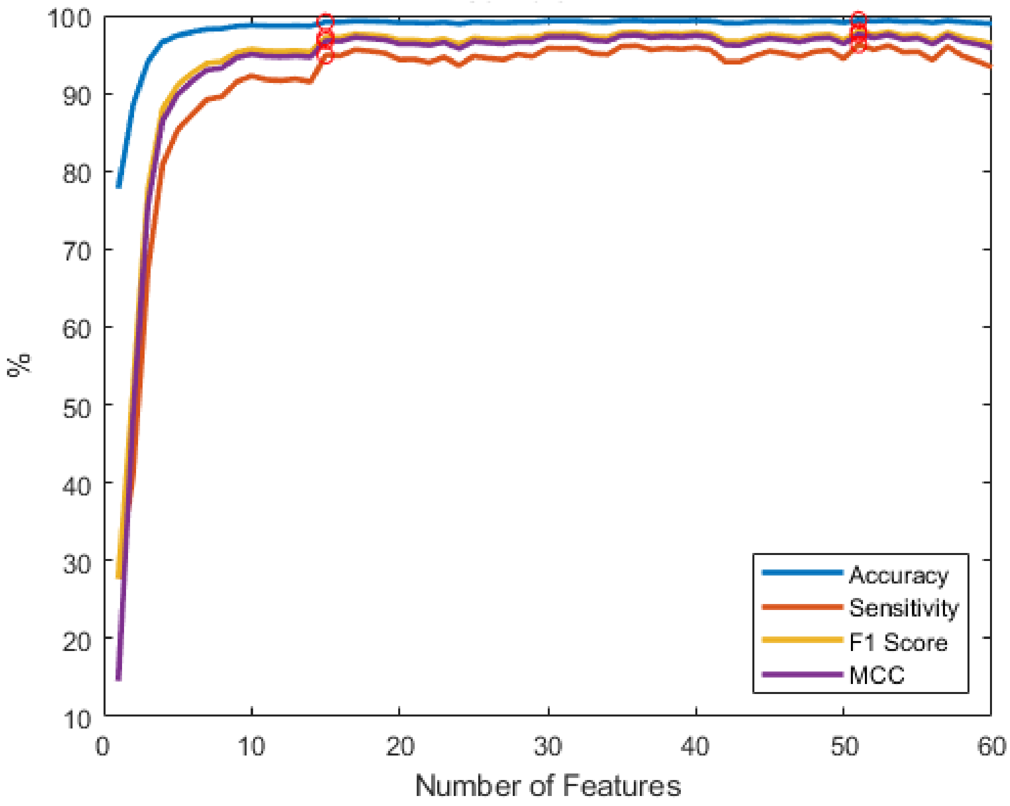


Figure 28 – Evaluation performance results reach with the Ensemble model trained from 1 up 60 first features ranked by MRMR method with cross-validation data to detect a NF.

Table 20 – Evaluation performance result of the Ensemble model with the 15 most relevant features ranked by the MRMR method when tested with unseen data to detect a NF

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1-score	MCC
MRMR	Ensemble	94.90	70.88	99.14	93.53	80.64	78.77

A post-processing algorithm was applied to the data predicted by the Ensemble classifier with the 51 most relevant features ranked by MRMR. The best result was reached with a window size of 22 samples. The NF is detected on average 710ms after the start and 1.48s before the end of the NF, being able to detect 100% of the NF that occurred. The number of misclassified normal walking samples decreased, approximately 98%. The number of NF detected, time to detect a NF, time until NF over are presented in Table 21 with and without post-processing. Through Figure 29, it is possible to compare the results obtained with and without post-processing procedure for all unseen data, which represent 56 NF in total.

Table 21 - Comparison of the results using post-processing algorithm and non-using post-processing algorithm

Post Processing	Number of NF Detected	Time to Detect a NF (s) (mean ± std)	Time until NF over (s) (mean ± std)	Window Size	NF Duration (s) (mean)
x	56/56	0.42 ± 0.38 s	1.76 ± 0.76 s	-	2.19 ± 0.88 s
✓	56/56	0.71 ± 0.48 s	1.48 ± 0.68 s	22	

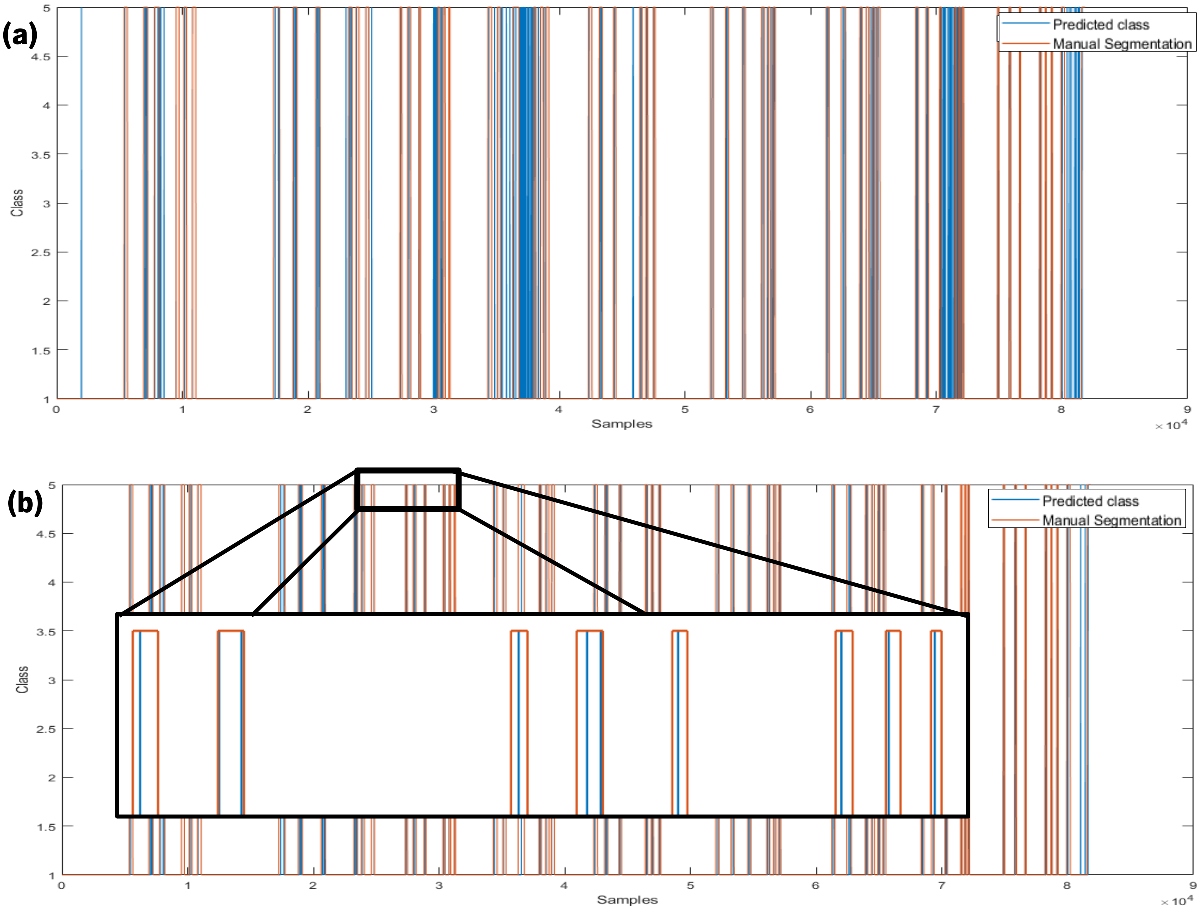


Figure 29 - Comparation between NF detection events non-use (a) and use (b) of the post-processing with a window size of 22 samples.

5.2.2 Case 2

According to the PCA it was possible to conclude that at least 70% of the variance is explained by 17 PC as depicted in Figure 30. Based on the information from the 17 PC, 31 features were estimated. As in the previous case, 60 features were used to perform the progressive analysis of feature set size in order to get the best model. The sixty most relevant features ranked by the MRMR, Relieff and PCA are presented in Appendix 2 (Table 46).

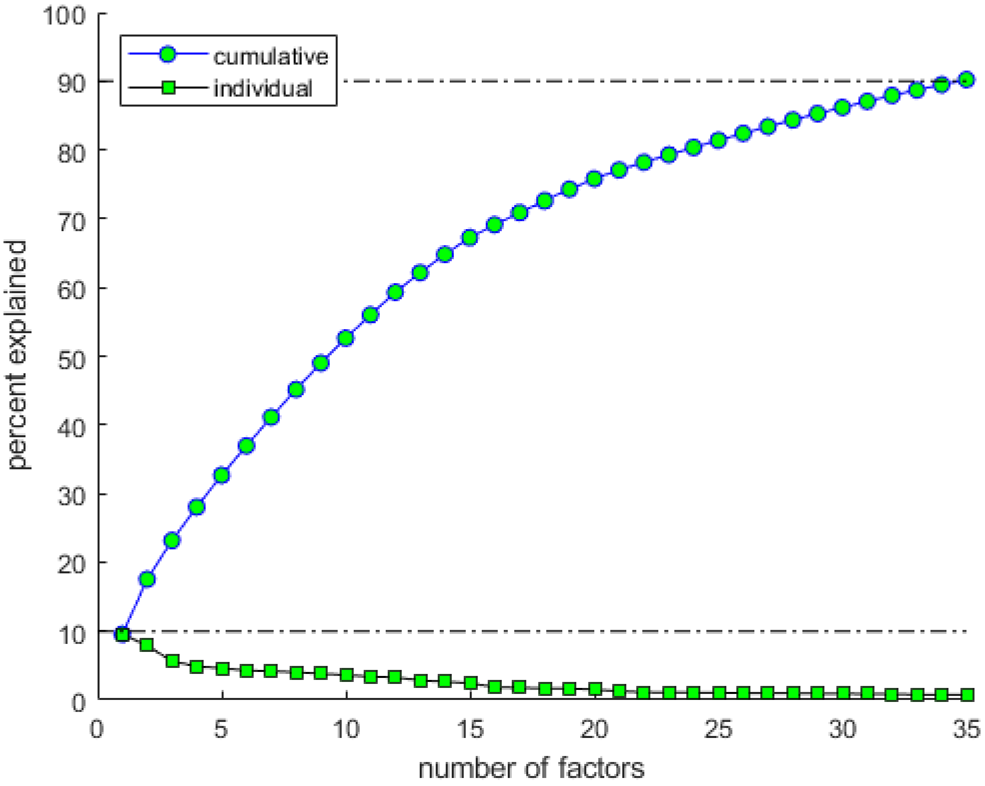


Figure 30 - Screen plot of PCA when using the dataset to detect the NF direction.

The results obtained for Case 2 are presented below, where the purpose is to detect the NF direction. Table 22 shows the performance achieve by several models when tested with unseen data. The best result was achieved by SVM classifier (Gaussian kernel) with the 60 most relevant features ranked by Relieff (ACC= 58.97%, SENS= 58.95%, SPEC=79.51%, PREC=58.96%, F1-score=58.92% and MCC = 38.45%). The worst performance was obtained by DT classifier with the 36 most relevant features ranked by PCA (ACC= 38.61%, SENS= 39.00%, SPEC=69.30%, PREC=38.31%, F1-score=38.47% and MCC = 8.04%). The cross-validation results are shown in Appendix 2 (Tables 47, 48 and 49).

Table 22 – Evaluation performance results achieved for all combinations of feature selection methods, number of features and machine learning models when tested with unseen data to detect the NF direction

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1-score	MCC	Number of Features
Relieff	KNN Equal	50.83	51.02	75.43	50.74	50.71	26.35	52
	DT	40.19	40.37	70.12	40.21	40.01	10.44	21
	Ensemble	48.26	48.54	74.26	49.12	48.27	23.07	24
	LDA	43.63	44.10	71.93	43.72	43.30	15.95	27
	QDA	45.98	46.32	73.16	46.96	45.87	19.79	29
	SVM Linear	49.54	49.54	74.83	49.71	49.53	24.43	56
	SVM Gaussian	**58.97**	**58.95**	**79.51**	**58.96**	**58.92**	**38.45**	**60**
	SVM Polynomial	54.88	54.76	77.42	54.42	54.75	32.19	36
MRMR	KNN Equal	44.62	44.90	72.38	44.78	44.52	17.25	7
	DT	41.74	42.05	70.92	41.73	41.72	12.85	36
	Ensemble	46.77	46.92	73.43	47.25	46.85	20.47	57
	LDA	45.54	46.13	72.92	45.12	45.03	19.20	4
	QDA	49.69	49.72	74.85	52.25	49.58	25.68	9
	SVM Linear	47.22	73.62	47.38	26.38	46.94	20.97	39
	SVM Gaussian	55.79	55.87	77.86	56.04	55.77	33.82	53
	SVM Polynomial	56.65	56.89	78.32	56.89	56.62	35.14	47
PCA	KNN Equal	44.05	44.09	72.02	43.98	44.01	16.06	59
	DT	38.61	39.00	69.30	38.31	38.47	8.04	27
	Ensemble	44.92	45.16	72.55	45.12	44.83	17.70	39
	LDA	45.78	45.74	72.88	46.00	45.59	18.75	58
	QDA	44.37	44.43	72.24	44.57	44.38	16.72	58
	SVM Linear	50.20	50.34	75.20	50.62	50.16	25.66	59

Table 22 – Continued

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1-score	MCC	Number of Features
PCA	SVM Gaussian	53.35	53.44	76.68	53.49	53.38	30.14	59
	SVM Polynomial	55.49	55.48	77.80	55.69	55.47	33.35	58

Since the best performance was achieved by SVM classifier (Gaussian Kernel), the hyperparameters were optimized, focusing on box constraint, and kernel scale. The results for hyperparameters optimization are shown in Table 23.

Table 23 –SVM Gaussian kernel hyperparameters values optimized to detect the NF direction

Hyperparameters	Value
Box Constraint	865.73
Kernel Scale	0.4531

All the evaluation metrics results obtained by SVM Gaussian kernel with 60 features ranked by the Relieff method are presented in Table 24. The cross-validation result is shown in Appendix 2 (Table 50).

Table 24 – Evaluation performance result of the SVM Gaussian kernel with the hyperparameters optimized when tested with unseen data to detect the NF direction

Feature Section Method	Classifier	ACC	SENS	SPEC	PREC	F1-score	MCC	Number of Features
Relieff	SVM Gaussian	54.24	54.61	77.22	54.50	53.97	31.83	60

5.3. Discussion

By analysing the performance of different models when tested with unseen data without the optimization of hyperparameters, it can be concluded that the best result was reached with Ensemble classifier using the MRMR method. The ACC, SPEC, PREC values are closer to 95%, and F1-score values and MCC are, approximately, 82% and 80%, respectively. The lowest value of all metrics was the SENS with 71.63%, which indicates a considerable number of FN. This means, the model is classifying samples as normal walking when they should be classified as NF. In this case, the subject is in a danger situation, and the model is unable to detect. The KNN Equal and DT classifiers present lower MCC and F1-score than Ensemble with MRMR. The performances obtained for each model during the validation were much higher than those obtained with the unseen data. This may indicate that the models may be adapting too much to the training data.

In order to improve the model performance, the hyperparameters of KNN Equal, DT and Ensemble classifiers were optimized when using the MRMR. The hyperparameter optimization in the case of Ensemble classifier did not present an improvement. When comparing with the initial model built, the difference was 0.01% for F1-score and 0.06% for MCC. For KNN Equal and DT classifiers the hyperparameters optimization also did not conferred a significant difference in the model performance.

In Case 1 the dataset is unbalanced, so in the sense of increasing the number of samples from minority class, the NF samples were duplicated. The Ensemble classifier with the 51 most relevant features ranked by MRMR was built with the new dataset. Although SENS has increased 0.4% and the SPEC 0.3%, on the other hand, PREC, F1-score and MCC decreased, 2.1%, 0.5%, 0.9%, respectively. It is possible to conclude that the oversampling of minority class does not improve the model performance despite the SENS increase. With misclassification costs, 3, 5 and 10, the SENS improved as pretended, however the SPEC, PREC, F1-Score and MCC decreased. Reduce the number FN is crucial but increase the FP is not beneficial. If a NF prevention strategy is applied, such as stop the walker, and the user is not prepared for this, a dangerous situation can be created.

From Figure 28, which represents the model performance for Ensemble classifier using the MRMR method for several feature sets, it is possible to conclude that using 15 or more features the model performance is almost constant. It is intended to detect the NF as quickly as possible in order to act when a dangerous situation is detected. The model performance reach with 15 features did not decrease very significantly, considering that the dataset has been reduced 75%. Across this analysis, it is possible to deduce that the 45 extra features do not add much more information to the model.

The Ensemble was the model that presented the best results in relation to the MCC and F1-score with only 51 features. So, in order to reduce the number of FP, a post-processing algorithm was applied. From the results obtained, the number of FP decreased considerably, 98%. However, the time required to detect the NF from the moment it starts increased by an average of 290ms for a window of 22 samples. However, it is still possible to detect a NF 1.48s before it ends. Although the model is not ideal, all NF were detected with and without post-processing. The window size defined to detect the NF may have to be adjusted depending on the prevention strategy that will be applied.

The diversity of data, since it was collected with two very different velocities which change the signal pattern as well as the small amount of NF data, can make the model not be able to learn as well as the intended.

In Case 2, to detect the NF direction, the models performance were very weak. The best performance was achieved with SVM Gaussian kernel using the Relieff method. However, the performance was very low with a F1-score of 58.92%, which means the model cannot reliably differentiate the NF direction. The performance obtained by SVM Gaussian kernel model with hyperparameter optimization was worse than registered without optimization.

The fact that the model built cannot distinguish the direction of the NF may be related to the small dataset. Another aspect, it is the fact that NF events can happen in several ways. Since NF are involuntary and aleatory situations, the same person and different people do not always perform a NF the same way.

CHAPTER 6 – INCIPIENT NEAR FALL DETECTION

This chapter presents the methodology followed to discriminate between an INF and normal walking based on machine learning classifiers. In this dissertation an "Incipient Near Fall" means the moment prior of a sudden change in the subject's balance. In particular, it is intended to investigate whether the INF presents any evidence which might indicate that the subject is in a dangerous situation. With the purpose to increase the time available for action to help the user to restore the balance as quickly as possible and, thus, avoid any accompanying injury or even a fall.

The approach developed involves six main sequential parts, namely, data recording, data pre-processing, data labeling, feature computation, feature selection methods and machine learning classifiers. The first two parts are described in detail in Chapter 4. Further details of the last four parts as well as the results obtained and their discussion are described below.

6.1. Methods and Materials

The Figure 31 depicts an overview of all steps accomplished subsequent to data acquisition and pre-processing. A total of ***n*** features was computed and normalized, after data labelling. Then, ***X*** most relevant features ranked by several feature selection methods were combined with different classifiers in order to achieve which feature combination produces the best classification performance. All these steps are described in more detail below.

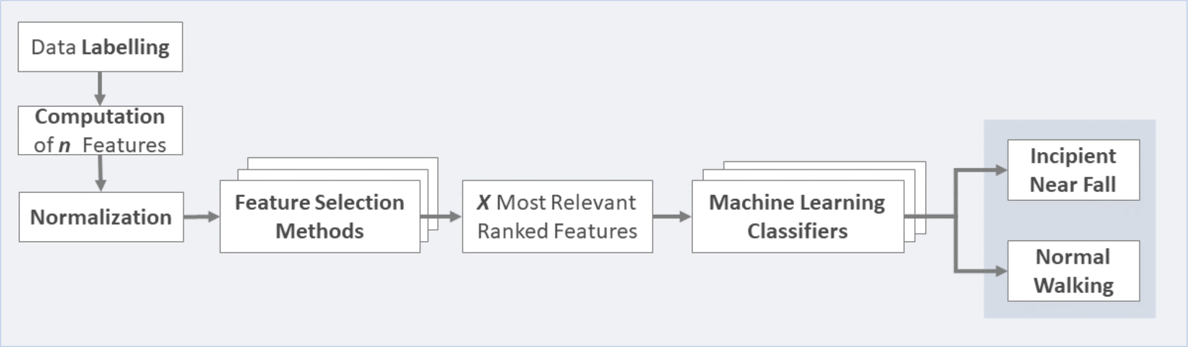

Data Labelling
Computation of *n* Features
Normalization
Feature Selection Methods
X Most Relevant Ranked Features
Machine Learning Classifiers
Incipient Near Fall
Normal Walking

Figure 31 - Schematic overview of the different steps performed to discriminate an INF from a normal walking.

Before explaining the data labelling process, it is important to introduce two gait events: the heel contact in the ground, HS, and the foot raising from the ground, TO. These two events can be detected using the FSR signals. The FSR placed in the heel zone allows the detection of the HS, when a quick rise in the signal value is evident. On the other hand, the FSR located in the front zone of the foot helps to detect the TO, when a quick decline in the signal value is observed. Both TO and HS events happen consecutively during walking since is a repetitive process [2][108].

The primary step to label the data was to determine the last gait cycle that happened before the beginning of the NF. Subsequently, in each test, the INF was labelled from the gait event before the NF occurred until the same event previously detected. For instance, if the last event before NF occurred was HS_{Right_NF}, the duration of the INF is defined from the HS_{Right} that occurred before the detection of the HS_{Right_NF} to the HS_{Right_NF} (complete gait cycle).

Several features were calculated after the data labelling, 170 in total. The first 169 features are the same as those computed in Chapter 5 and are presented in Appendix 1 (Table35). The new feature added is related to the HS and TO gait events. With these events it is possible to split the gait into four main states that are repeated over time, i.e., HS_{Right} up TO_{Left}, TO_{Left} up to HS_{Left}, HS_{Left} up TO_{Right}, and lastly TO_{Right} up HS_{Right}. Therefore, the new feature represents the duration of each state. The purpose of this is to investigate if the duration of each state is relevant to distinguish normal walking from INF situation. Data normalization between 0 and 1 was the last step conducted before the implementation of feature selection methods.

Three feature selection methods and five machine learning classifiers were implemented. The different classifiers used were, namely, DT, Ensemble and KNN, with an equal, inverse and squared inverse distance weighting functions. All combinations of feature selection methods, classifiers and number of features were tested to reach the best model performance. For this purpose, each model performance analysis was performed with a successive cross-validation with bigger feature sets. In each iteration, the next most significant feature given by the feature selection method is added to the feature set. Since executing an incremental analysis with 170 features is computationally expensive for five models, only ***X*** features were used. The number of features was determined based on PCA as performed in the previous chapter. The best feature set was selected according to cross-validation performance. It should be noted that the training dataset corresponds to approximately 70% of all data collected. At the end, 15 models were built (Figure 32).

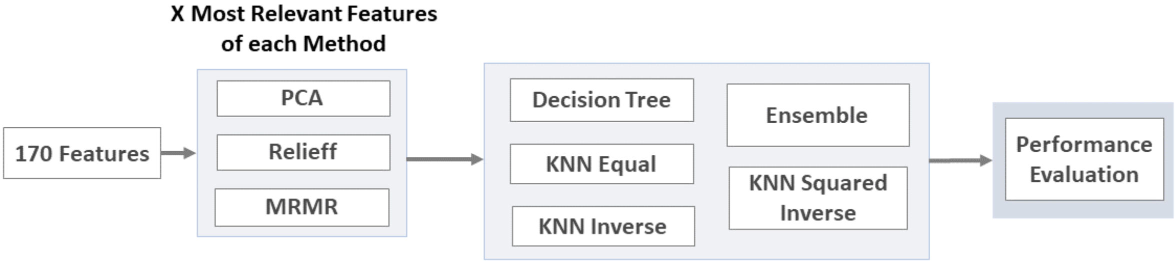


Figure 32 - Methodology implemented for building machine learning models to detect an INF.

The 15 models built were tested with unseen data which correspond to approximately 30% of all data acquired (Figure 33). Several metrics were calculated to understand how the models really generalize. The best model was selected based on the performance achieve with unseen data.

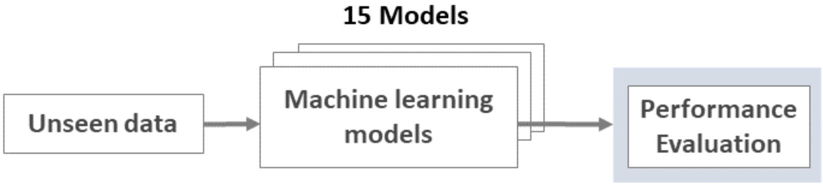


Figure 33 – Methodology implemented for evaluating the model performance with unseen data to detect an INF.

In this chapter, the training data is unbalanced, i.e. there is a high disproportionately instance ratio between classes. Roughly, 84% of the data correspond to normal walking class, while only 16% are INF data. In this sense, the INF data was duplicated, and a new training dataset was generated. However, only the most successful combination of the feature selection method and machine learning classifier achieved with the previous normalization was applied. The methodology followed for this situation was similar to the previous described, with the exception that only one feature selection method and one classifier was implemented.

Ultimately, the z-score normalization was implemented. For this specific case as for the preceding situation, just the best combination of the features selection method and classifier was employed.

6.2. Results

As depicted in Figure 34, 16 PC explain at least 70% of the variance. Then with the information contained in the 16 PC, 31 features were selected as the most relevant. However, it was decided to allocate almost the double of features (60 features) to provide a more complete analysis. The sixty most relevant features ranked by MRMR, Relieff and PCA are presented in Appendix 3 (Table 51).

It was performed an analysis with 60 different feature sets for the 15 possible combinations (5 classifiers with 3 feature selection methods). The number of features selected was based on the models performance achieved with cross-validation data. The best cross-validation results are presented in the Appendix 3 (Tables 52, 53 and 54). The Ensemble classifier was the one that yielded the best output, in general, independently of the feature selection methods. The Ensemble classifier produced MCC and F1-score results over 99% when using the PCA (MCC=99.65% and F1-score=99.71%) and Relieff (MCC=99.67% and F1-score=99.73%) methods , and relatively poorer results when using the MRMR method (MCC= 85.74% and F1-score 87.11%).

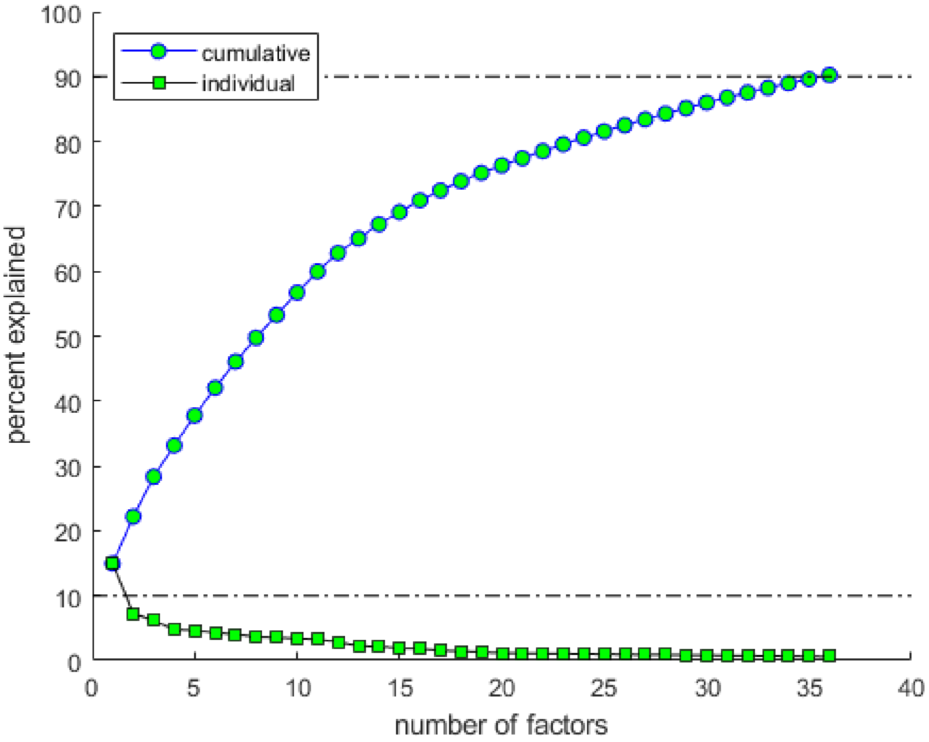


Figure 34 - Screen plot of PCA when using the dataset to detect an INF.

The models built were further tested with unseen data to determine the models behaviour. The overall metrics obtained are presented in Table 25. Evaluation performance metrics revealed that the Ensemble classifier presented the best results when using the 40 most relevant features ranked by the Relieff method.

Table 25 – Evaluation performance results achieved for all combinations of feature selection methods, number of features and machine learning models when tested with unseen data to detect an INF

Feature Selection Method	**Classifier**	**ACC**	**SENS**	**SPEC**	**PREC**	**F1 - score**	**MCC**	**Number of Features**
MRMR	DT	70.14	21.94	79.05	16.22	18.65	0.88	57
	Ensemble	80.48	4.38	94.55	12.92	6.54	1.75	10
	KNN Equal	71.47	18.29	81.30	15.31	16.67	0.38	3

Table 25 - Continued

Feature Selection Method	**Classifier**	**ACC**	**SENS**	**SPEC**	**PREC**	**F1 - score**	**MCC**	**Number of Features**
Relieff	DT	73.10	32.21	80.66	23.54	27.20	11.39	11
	Ensemble	**81.86**	**24.36**	**92.49**	**37.50**	**29.54**	**20.27**	**40**
	KNN Equal	69.82	29.91	77.20	19.52	23.62	6.05	9
PCA	DT	70.76	22.95	79.60	17.22	19.68	2.29	44
	Ensemble	82.93	20.56	94.46	40.68	27.31	20.22	38
	KNN Equal	73.96	23.72	83.24	20.74	22.13	6.60	60

Figure 35 represents the evolution of ACC, SENS, MCC and F1-score performance obtained with the Ensemble classifier trained from 1 to 60 most relevant features ranked by the Relieff method. The analysis of Figure 35 shows that with only 8 features the overall model training performance is quite high (ACC=99.89%, SENS=99.95%, MCC= 99.62%, and F1-Score= 99.68%). It is noticeable that from the eight features the model performance during training remains almost constant.

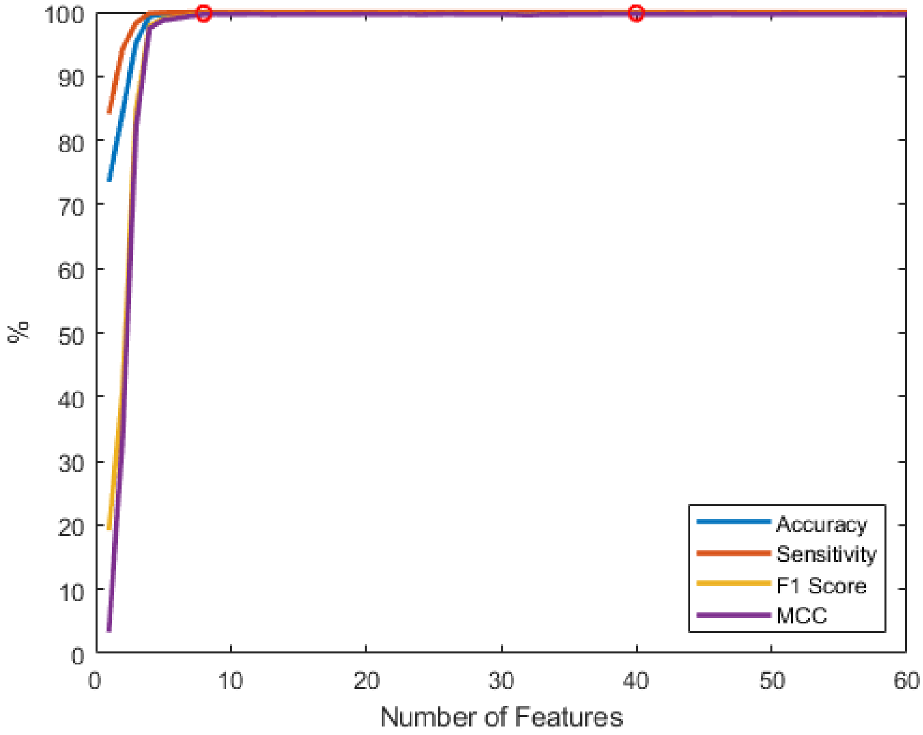


Figure 35 - Evaluation performance results reached with the Ensemble model trained from 1 up to 60 first features ranked by Relieff method with cross-validation data.

Since the dataset used to train the model presents a significantly larger number of instances of one class than the other class, the data of the minority class (INF) was duplicated, and the Ensemble classifier with the features ranked by Relieff method was applied. In Appendix 3 (Table 55) is show the cross-validation performance achieve for the best feature set. Table 26 presents the result produced by the model when tested with unseen data.

Table 26 – Evaluation performance result of the Ensemble model with data oversampling when tested with unseen data to detect an INF

Feature Selection Method	**Classifier**	**ACC**	**SENS**	**SPEC**	**PREC**	**F1 - score**	**MCC**	**Number of Features**
Relieff	Ensemble	84.40	15.94	97.05	49.99	24.17	21.68	47

Another type of normalization was applied, namely, the z-score normalization. For this study only the Relieff feature selection method, and the Ensemble classifier was implemented. The best performance result achieved with the cross-validation data for this combination was with the 20 most relevant features (Appendix 3 Table 56). Therefore, the model built with the most relevant features was subsequently tested with unseen data and the overall metrics reached are presented in Table 27.

Table 27- Evaluation performance result of the Ensemble model with z-score normalization when tested with unseen data to detect an INF

Feature Selection Method	**Classifier**	**ACC**	**SENS**	**SPEC**	**PREC**	**F1 - score**	**MCC**	**Number of Features**
Relieff	**Ensemble**	**83.24**	**24.57**	**94.09**	**43.46**	**31.39**	**23.87**	**20**

6.3. Discussion

The first step was to determine the best feature set for each combination of feature selection method and machine learning algorithm through cross-validation. From all possible combinations, it was concluded that the Ensemble classifier with PCA and Relieff methods reached the highest MCC and F1-score outcomes, around 100%. However, the model performance when tested with unseen data decreased, drastically, around 80% and 70% for MCC and F1-score, respectively. Although high results were obtained with the cross-validation data, the same did not happen with the unseen data which indicates overfitting. This means the model is adjusting very well to the training data, not being able to distinguish between normal walking and INF events.

By analysing Figure 35, it was possible to establish that from eight features the overall performance of the Ensemble classifier when using the Relieff method is practically constant. The difference in terms of MCC and F1-score was, roughly, 0.05% for both metrics. This indicates that the other features added do not provide a considerable benefit to the learning model process. Thus, increasing the number of features (more than 60 features) in this case will not improve the model performance.

As this is an unbalanced dataset, minority class data were oversampled in order to improve model performance. Nevertheless, it was observed that this failed, i.e., the model remained unable to distinguish the two classes.

The z-score normalization was implemented and a new model with the combination of the Relieff method and the Ensemble classifier was built. Even though there was a very insignificant improvement, the model still fails to discriminate between a normal walking and an INF event. It should be noted that as in the previous case, the model, although performing very well with the cross-validation data with unseen data this was not observed.

The poor performance of the models in distinguish between normal walking and an INF may indicate that there is no relevant information before the NF to detect that the user is in a dangerous situation. On the other hand, the machine learning algorithms applied may not have been the most indicated to identify the differences between normal walking and an INF event. It may be more suitable for this situation, for instance, deep learning algorithms. Another hypothesis that may have resulted in the poor performance of the models may have been the small amount of data available related to the INF situations.

CHAPTER 7 – GAIT EVENT DETECTION USING A WALKER

The main objective of this chapter is to detect two gait events, TO and HS of each foot, when using a walker. It is intended to identify the HS and TO events using machine learning models with inertial signals acquired on the lower trunk as inputs. FSR signals gathered from each insole will be used as ground truth.

To do so, four major steps were followed after gathered data and pre-processing, namely: i) data labelling; ii) computation of several features; iii) implementation of different feature selection methods; and iv) training and testing distinct machine learning models. Once again, the best feature set was analyzed for each machine learning algorithm determined by feature selection methods. The results obtained and their discussion are presented below, included a more detailed description of the approach applied.

7.1. Gait analysis and Fall Risk

The assistive devices for walking are generally used by subjects who have mobility problems and need aid for locomotion. Subjects who have experienced a fracture or who suffer from neurological disorders, e.g. stroke, may require a walking aid [109]. The neurological disorders can affect the gait pattern and increase the risk of falling, such as patients with PD who have a higher risk to fall forward due to the festinate gait [4]. In these cases, of gait abnormality, the gait analysis (GA) can provide an extra information related to the user and can be also useful for clinicians [110].

The human gait is characterized by events that are repeated over time and can be described as a set of gait cycles [2], [108]. The gait cycle can be divided in two major phases: stance and swing. The first one comprises 60% of the gait cycle and the other 40%. The stance phase corresponds to the time while the foot is on the ground (starts with HS event), while swing phase correspond to period where the foot is in the air (starts with TO event). A complete gait cycle, generally, includes the period between ground contacts of the same heel, i.e., a gait cycle, is usually defined by the start and the end of a HS. The term stride has the same meaning as gait cycle and comprises two steps. One step starts with the heel contact and ends when the opposite heel contacts with the ground as exemplified in Figure 36 [108], [111].

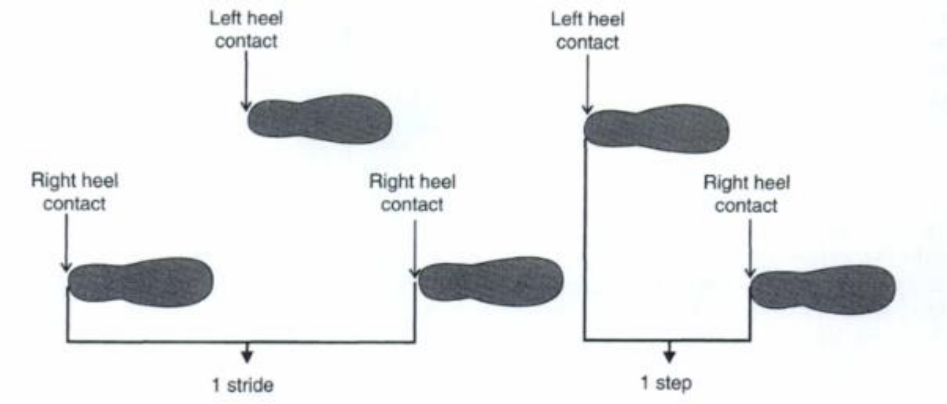


Figure 36 - Representation of stride and step [111].

The stance and swing phases can also be subdivided in eight periods as shown in Figure 37, such as [108], [112], [113]:

- Initial contact: corresponds to the instant the foot contacts the ground and represents the beginning of stance phase (0-2% of GC).
- Loading response: corresponds to the period where both feet are in contact with the ground, starts with the initial floor contact and ends with opposite foot lifting on the ground (0-10% of GC).
- Mid stance: starts when the opposite foot rise from the ground and continues till the bodyweight is aligned over the forefoot (10-30% GC).
- Terminal stance: starts when the heel lifts from the ground and proceeds until the opposite foot is in contact with the ground (30-50% of GC).
- Pre-swing: starts with the other foot contact with the ground and culminates with ipsilateral TO (corresponds to the transition of stance phase to the swing phase) (50-60% of GC)
- Initial swing: starts with foot rises from the ground and continuous until the other foot is opposite the stance foot (60-73% of GC).
- Mid swing: corresponds to the moment which swinging foot passes the opposite stance foot (73-87%).
- Terminal swing: starts when the tibia is vertical and continue until the heel contact with the ground.

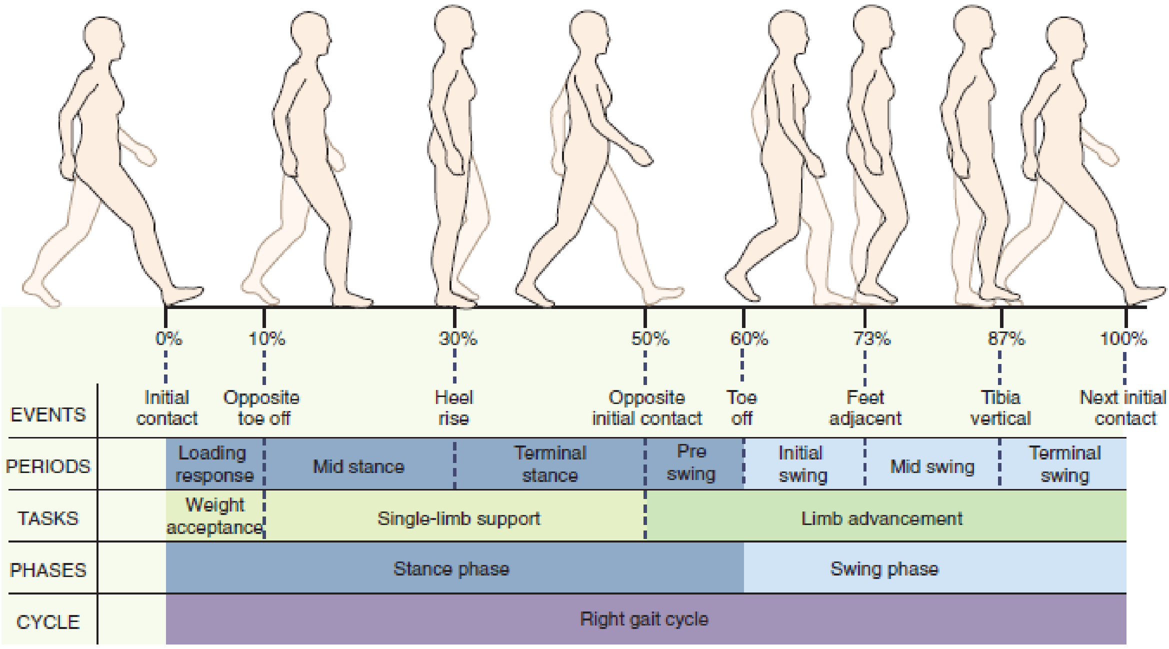


Figure 37- Human gait phases and corresponding events during a right gait cycle [113].

Several parameters associated with gait movements can be considered for GA, such as: kinetic parameters related to torque and moment; kinematic parameters concerning to joint and lower extremities motions; electromyography parameters related to muscle activity; and the spatiotemporal parameters associated with the distance and time measurements (e.g. step length, cadence, stride length and swing time) [112], [114]. Several approaches have been developed in order to measure the different gait parameters. For this purpose, both non-wearable (e.g. cameras and force platform) and wearable sensors (e.g. IMU and pressure insoles) have been tackled [2], [110], [112], [115].

The GA methods have an important role in medical field [110]. In the rehabilitation area it can be important to access the recovery process and planning the treatment, and can be also useful to diagnose and estimate the fall risk [112]. For example, patients with multiple sclerosis present modification in spatiotemporal parameter, translated by a decrease in steps length and cadence [110]. The use of assistive devices also alters the spatiotemporal parameters, such as, cadence, step time, step length, stance and swing time [109].

7.2. Material and Methods

The methodology followed to detect TO_{Right}, TO_{Left}, HS_{Right} and HS_{Left} events is similar to methodologies adopted for Chapter 5 and 6. The experimental procedure for the IMU and FSR data gathering is common to that carried out for the other chapters and is described in Chapter 4. Nevertheless, for this specific study, only data of normal walking were included. This means that from the data collected during the tests which pretend to simulate a NF were excluded the information corresponding to INF and NF being used only the normal walking data. With the purpose of obtaining the maximum amount of data possible. Figure 38 shows a schematic of the methodology applied to detect the gait events after data collection and pre-processing. It is important emphasize that the focus is to detect the gait events using exclusively the inertial data collected on lower trunk.

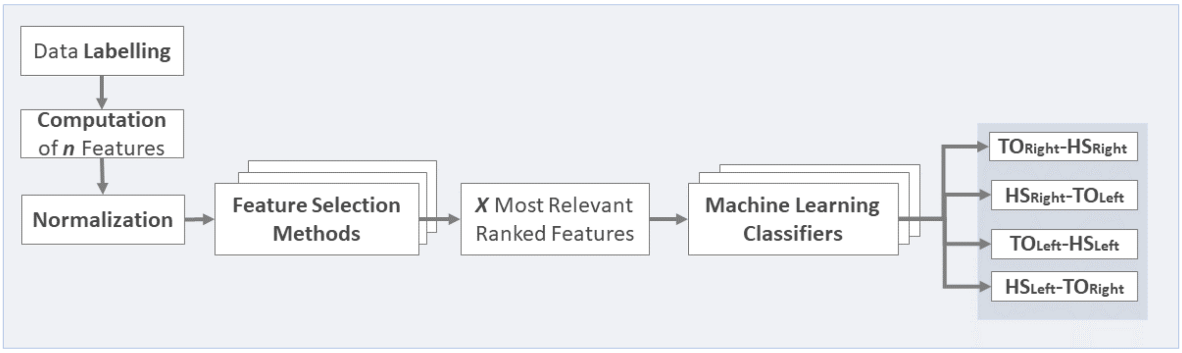

Data Labelling
Computation of *n* Features
Normalization
Feature Selection Methods
X Most Relevant Ranked Features
Machine Learning Classifiers
TORight-HSRight
HSRight-TOLeft
TOLeft-HSLeft
HSLeft-TORight

Figure 38 – Schematic overview of the different steps performed to discriminate human gait events.

In order to detect the two gait events through the FSR signals, an algorithm was developed which detects the sudden alterations in force signals (TO_{Right}, TO_{Left}, HS_{Right} and HS_{Left}). Therefore, through the FSR signals, the data was labelled in four states: TO_{Right}-HS_{Right} (state1), HS_{Right}-TO_{Left} (state 2), TO_{Left}-HS_{Left} (state 3) and lastly HS_{Left}-TO_{Right} (state 4) (Figure 39). It should be noted that each state transition corresponds either to HS or TO of the right or left foot.

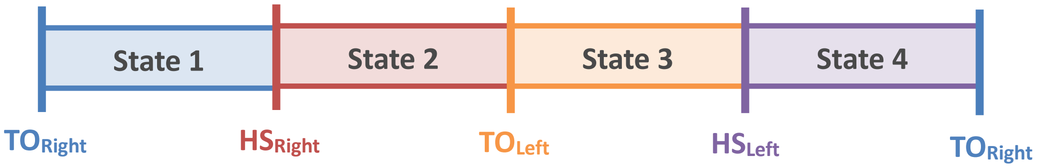


Figure 39 - Scheme of human gait division used for gait events detection.

Subsequently, all the features presented in the Appendix 1 (Table 35) were used with the exception of the FSR data which were excluded for this analysis. Which means, that in total 166 different features constitute the final dataset. The dataset was further separated in approximately 70% for training and 30% for testing.

After obtaining the final feature set, the data were normalized between 0 and 1. With the intention of obtaining the most relevant features, three feature selection methods were applied, namely, MRMR, Relieff and PCA. This whole process has to be performed again, since the dataset is different from that used in Chapter 5 and 6. Out of 166 features, only ***X*** were considered to achieve the best model as previously realized due to computational cost issues. The number of selected features was determined based on the PCA method as performed in the chapter 5.

Concerning machine learning algorithms, 7 different classifiers were used: DT, LDA, QDA, Ensemble and the KNN with an equal, inverse and squared inverse distance weighting functions. A progressive analysis of the number of features for all possible combination of classifiers and feature selection methods was performed with the intention of getting the best feature set. Among the different feature sets used to build the models, the most suitable set was determined according to the cross-validation performance. Thus, at the end, 21 different models were obtained (Figure 40).

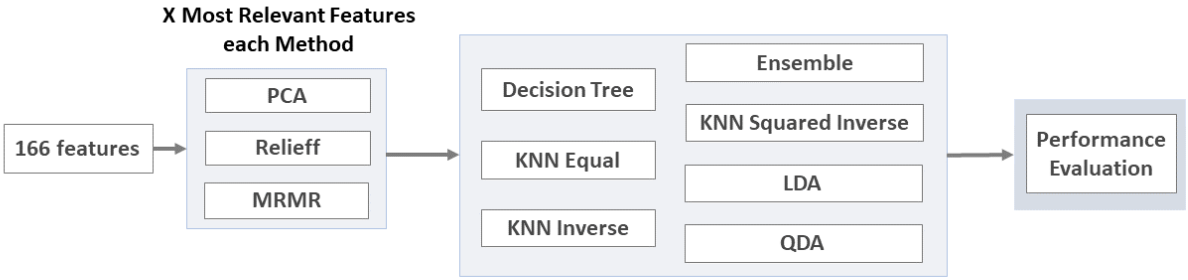


Figure 40 - Methodology implemented for building the machine learning models to detect human gait events.

After building all models, each model was tested with unseen data (Figure 41). For evaluate the model performance all the metrics mentioned in Chapter 4 were calculated (ACC, SENS, SPEC, PREC, MCC and F1-score). Through the analysis of the metrics mentioned, the model that best generalizes was the model selected.

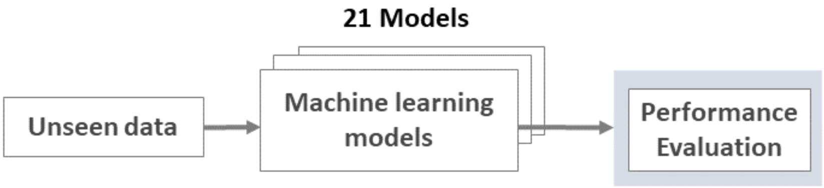


Figure 41 - Methodology implemented for evaluating the model performance with unseen data to detect human gait events.

Since it is intended to implement a real-time gait event detector, the time required to feature computation is crucial. Thus, for the best combination of the classifier with the feature set ranked by a specific method achieved previously, an analysis of the cross-validation metrics for each feature set was performed (F1-score, MCC, ACC and SENS). Then, a new model was built and tested with the unseen data.

All hyperparameter values were optimized for the best combination of classifier and feature selection for the best number of features encountered, with the intention of building a model with better performance. For this purpose, an option provided by MATLAB was used, which allows the optimization of all hyperparameters of the classifier. After building the model this one was tested with unseen data to test its predictive potential.

Lastly, a post-processing algorithm was applied to the predicted data by the best model and they were compared against the classifier results. In order to determine if there was a more reliable event detection.

The post-processing algorithm implemented is depicted in the flowchart of Figure 42. This algorithm aims to decrease the misclassified samples in a short period of time. Initially, it is necessary to pre-define the window size (number of samples) that contains only previous samples. Then, it is checked which class is the most present and that class is assigned to the current sample. If there are classes that contain the same number of samples inside the window, the class assigned to the current sample is the same as that assigned to the previous sample. To enhance post-processing algorithm, two restrictions were imposed:

- If the dominant class inside of window is the class 2 (state 2), the class assigned to the previous sample cannot be class 3 or 4. Therefore, if this happens, the class assigned to the previous sample is assigned to the current sample.

- If the dominant class is class 4 (state 4) the class assigned to the previous sample cannot be class 1 or 2. If this is the case, the same procedure shall be followed as described above.

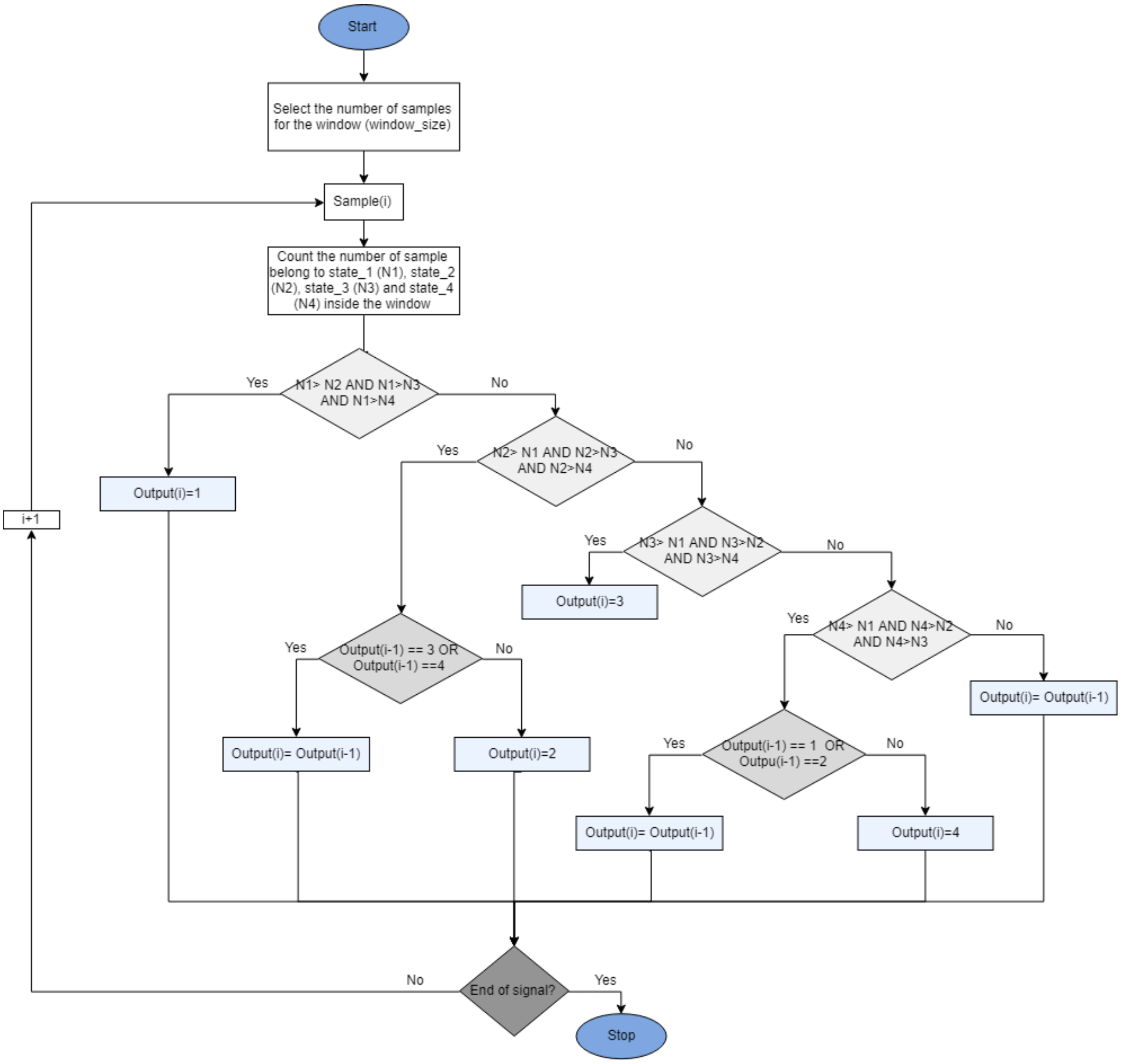

Start
Select the number of samples for the window (window_size)
Sample(i)
Count the number of sample belong to state_1 (N1), state_2 (N2), state_3 (N3) and state_4 (N4) inside the window
Yes
N1> N2 AND N1>N3 AND N1>N4
No
Output(i)=1
Yes
N2> N1 AND N2>N3 AND N2>N4
No
i+1
Yes
N3> N1 AND N3>N2 AND N3>N4
No
Output(i)=3
Yes
N4> N1 AND N4>N2 AND N4>N3
No
Yes
Output(i-1) == 3 OR Output(i-1) ==4
No
Output(i)= Output(i-1)
Output(i)= Output(i-1)
Output(i)=2
Yes
Output(i-1) == 1 OR Outpu(i-1) ==2
No
Output(i)= Output(i-1)
Output(i)=4
No
End of signal?
Yes
Stop

Figure 42 - Flowchart of the post-processing algorithm implemented to detect human gait events.

7.3. Results

Through the PCA, it was determined that 18 PC explain at least 70% of the variance as illustrated in Figure 43. Based on the information from the 18 PC, 36 features were selected. However, for this study 60 features were considered which allows a more complete analysis of the minimum number of features required to build a reliable model. The sixty most relevant features ranked by MRMR, Relieff and PCA are shown in Appendix 4 (Table 57).

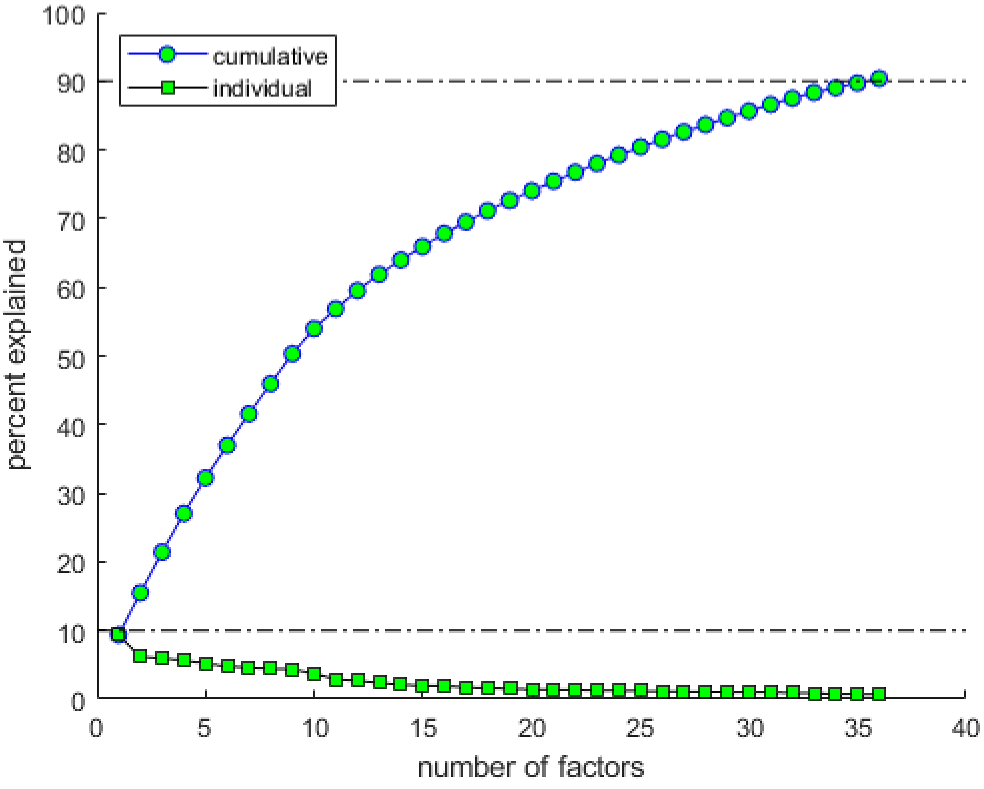


Figure 43 - Screen plot of PCA when using the dataset to detect human gait events.

The next step was to obtain the best number of features from the 60*number of feature selection methods possible sets for each machine learning classifier. The model that presented the best performance with cross-validation data was selected. The metrics obtained for each model during the validation phase for the best feature set are presented in the Appendix 4 (Table 58, 59 and 60). Subsequently, the models were tested with unseen data in order to find the best model. The evaluation performance (ACC, SENS, SPEC, PREC, F1-score and MCC) of each model with unseen data is shown in Table 28. The model that presented the best performance was the Ensemble with the first 52 most relevant features ranked by the Relieff method (ACC =91.11%, SENS=87.62%, SPEC=96.78%, PREC= 90.70%, F1-score=88.98% and MCC= 86.04%).

Table 28 - Evaluation performance results of all machine learning models when tested with unseen data to detect human gait events

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1-score	MCC	Number of Features
MRMR	LDA	74.73	67.05	91.17	68.79	67.74	57.20	32
	QDA	73.27	62.58	90.41	66.50	63.59	55.11	16
	KNN Equal	89.26	86.44	96.31	86.93	86.68	83.02	18
	DT	81.51	76.90	93.59	77.87	77.33	71.01	36
	Ensemble	90.82	87.42	96.69	90.21	88.66	85.60	48
Relieff	LDA	73.82	66.56	91.02	67.21	66.78	57.94	54
	QDA	73.87	65.06	90.79	67.50	65.90	57.21	54
	KNN Equal	85.11	81.25	94.77	82.73	81.93	76.82	50
	DT	80.33	76.39	93.24	76.36	76.37	69.61	30
	Ensemble	**91.11**	**87.62**	**96.78**	**90.70**	**88.98**	**86.04**	**52**
PCA	LDA	66.58	58.18	88.76	58.08	58.06	46.83	60
	QDA	66.89	59.50	88.99	59.16	59.23	48.18	60
	KNN Equal	66.76	62.45	88.39	62.21	62.32	50.70	59
	DT	79.33	75.16	92.88	75.24	75.19	68.08	59
	Ensemble	89.19	85.22	69.10	88.62	86.66	83.10	59

With the purpose of achieve the best trade-off between model performance and number of features needed to decrease the computational time, the performance of the Ensemble classifier with different feature sets ranked by the Relieff method was evaluated. The sets analyzed included from 1 to 60 most relevant features (Figure 44).

The analysis of Figure 44 was followed by the construction of 3 Ensemble models with 18, 25 and 30 most relevant features ranked by Relieff method. The performance results of each model built when tested with unseen data are shown in Table 29. Considering the three models, the best performance was obtained with 30 most relevant features but there was a decrease of ACC, F1-score and MCC values of, approximately, 2.6%, 3.3% and 4.1%, respectively. The cross-validation results are shown in Appendix 4 (Table 61).

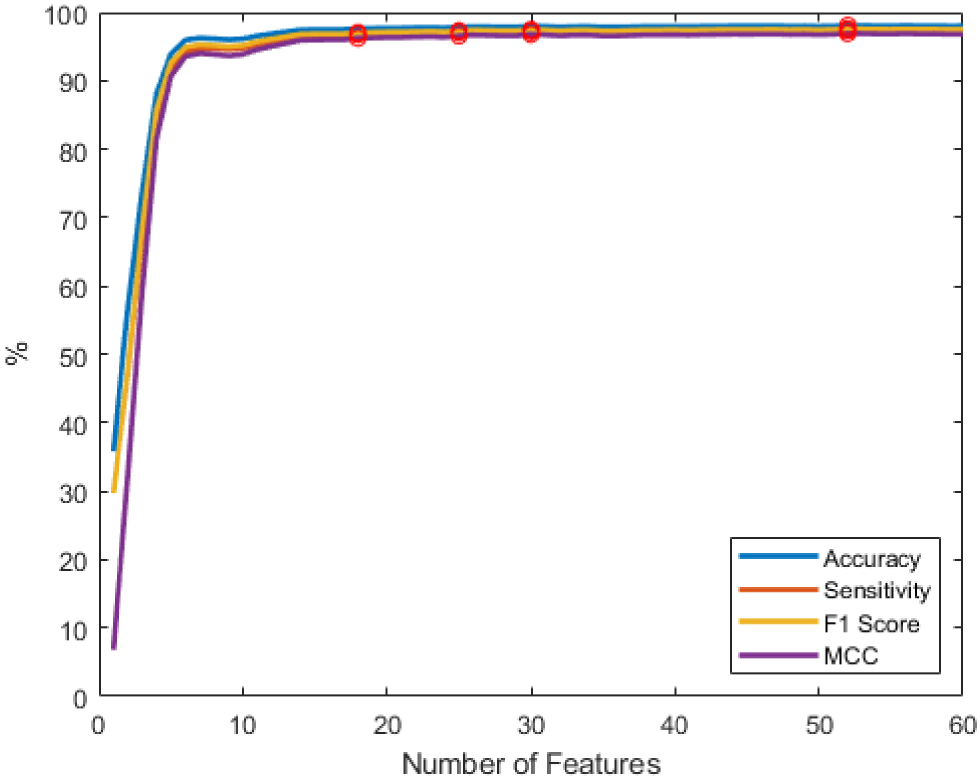


Figure 44 – Evaluation performance result reached with the Ensemble model trained from 1 up to 60 most relevant features ranked by the Relieff method to detect human gait events.

Table 29 – Evaluation performance results of the Ensemble model trained with 18, 25 and 30 features when tested with unseen data to detect human gait events

Feature Selection Method	**Classifier**	**ACC**	**SENS**	**SPEC**	**PREC**	**F1-score**	**MCC**	**Number of Features**
Relieff	Ensemble	87.58	83.35	95.57	86.14	84.56	80.40	18
		87.97	83.64	95.67	86.93	85.06	81.06	25
		88.56	84.36	95.89	87.44	85.70	81.90	30

Once selected the model that most successfully identifies the human gait events, the optimization of its hyperparameters was performed. The six hyperparameters optimized were: Ensemble Aggregation Method, Number of Learning Cycles, Learning Rate, Minimum Leaf size, Maximum Number of Splits, Number of Variables to Sample and Split Criterion. The values get for each hyperparameter are shown in Table 30. The performance results achieved by the Ensemble classifier with 52 features ranked by the Relieff method and hyperparameters optimized when tested with unseen data is presented in Table 31. The cross-validation results are shown in Appendix 4 (Table 62).

Table 30 - Ensemble hyperparameters values optimized to detect human gait events

Hyperparameters	Value
Ensemble Aggregation Method	RusBoost
Number of Learning Cycles	453
Learn Rate	0.0938
Minimum Leaf Size	1
Maximum Number of Splits	81017
Number of Variable to Sample	NaN
Split Criterion	Deviance

Table 31 - Evaluation performance result of the Ensemble model with the hyperparameters optimized when tested with unseen data to detect human gait events

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1-score	MCC	Number of Features
Relieff	**Ensemble**	**91.66**	**89.48**	**97.11**	**90.12**	**89.79**	**86.94**	**52**

After building the model, it was analysed the data gathered through the Ensemble classifier with the hyperparameters optimization. In order to accomplish a further analysis of the output generated by the model when tested with unseen data. Figure 45 shows a comparison of the ground truth data with the data predicted for a part of the complete test dataset. The blue rectangles in Figure 45 represent the misclassified samples in a short period of time, and which occur throughout all the test dataset.

Finally, the post-processing was applied to the output from the model in order to decrease the misclassified samples. The results obtained directly after post-processing for a window of 16 and 20 samples are shown in Table 32. A significant overall reduction was observed in the metrics calculated when compared to metrics obtained without any post-processing. There was a decrease of 8.5% and 11.5% in ACC, 10.7% and 14.2% in F1-score, lastly 13.6% and 18.1% in MCC for windows of 16 and 20 samples, respectively.

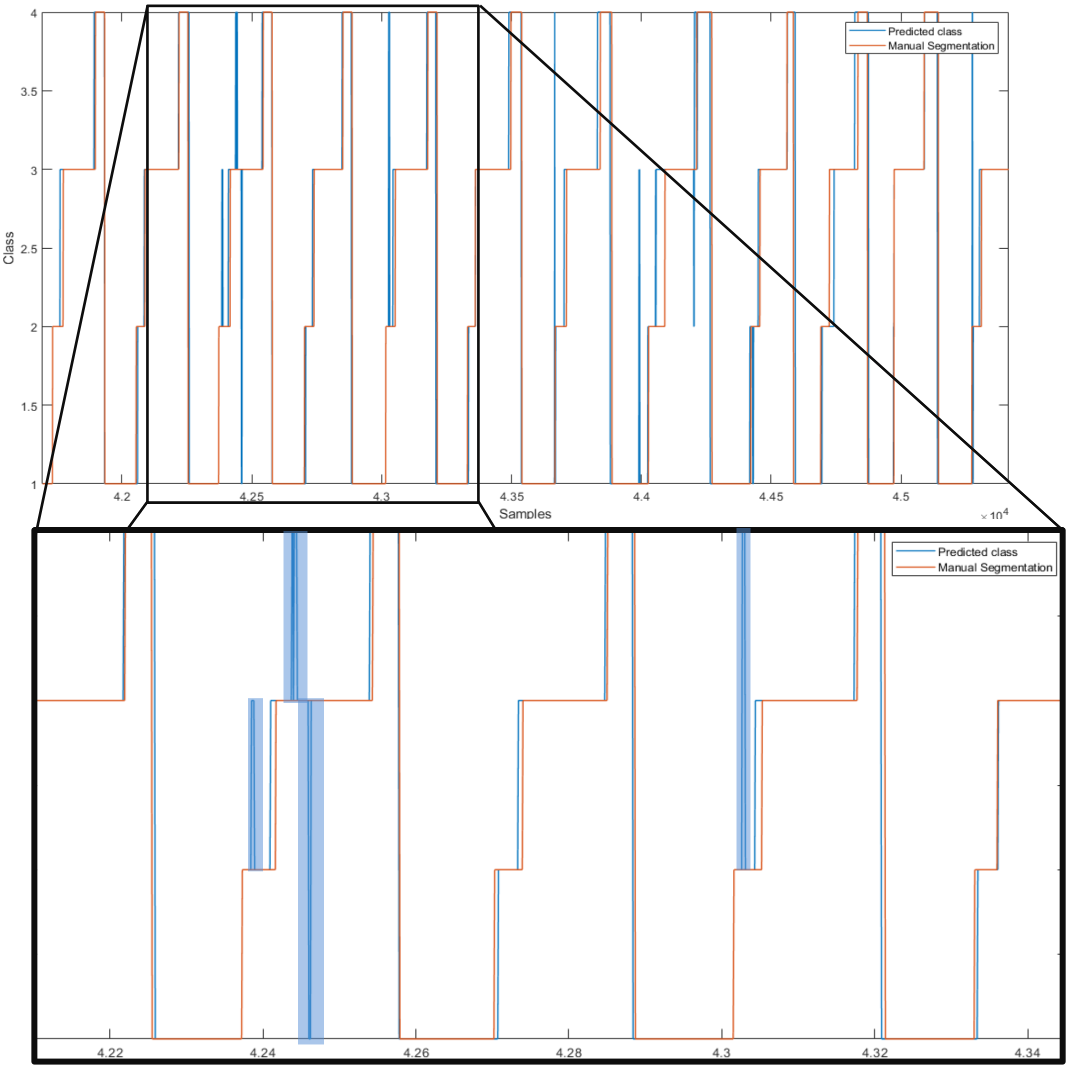


Figure 45 - Comparison between ground truth and the data predicted by the Ensemble classifier with hyperparameters optimized for part of the test dataset.

Table 32 – Evaluation performance results achieved using the post-processing algorithm with a window of 16 and 20 samples in the data predicted by the Ensemble model with hyperparameters optimized

Window (Samples)	ACC	SENS	SPEC	PREC	F1- score	MCC
16	83.13	78.82	94.18	79.46	79.13	73.36
20	80.21	75.31	93.17	75.91	75.60	68.82

The post-processing algorithm applied produced a delay of 8 and 10 samples for a window with 16 and 20 samples, respectively. This is due to the fact that the post-processing algorithm only classifies the current sample as TO_{right}, TO_{left}, HS_{Right} and HS_{left} if half of the samples in the window have also been classified with the respective event. Thereby the metrics were recalculated with the delay rectified. The results of the models performance are shown in Table 33 without the imposed delay in the transitions by the post-processing. In general, there was an improvement in overall metrics when compared with those obtained without post-processing. In particular, an increase of, approximately, 1.2% and 1.3% in ACC, 1.3% and 1.4% in F1-score, and 1.7% and 1.9% in MCC with windows of 16 and 20 samples, respectively.

Table 33 - Evaluation performance results achieved using the post-processing algorithm with a window of 16 and 20 samples in the data predicted by the Ensemble model with hyperparameters optimized (without considering the delay)

Window (Samples)	**ACC**	**SENS**	**SPEC**	**PREC**	**F1-score**	**MCC**
16	92.82	90.64	97.50	91.51	91.07	88.62
20	**92.92**	**90.78**	**97.53**	**91.70**	**91.23**	**88.82**

The Figure 46 represents the comparison between the signal obtained after post-processing application against the ground truth for a part of the test dataset. The post-processing was applied to the data provided by the Ensemble classifier with the hyperparameters optimized and trained with the 52 most relevant features ranked by the Relieff method. The blue rectangles represent the areas where the misclassified samples in a short period of time were removed. It is important to note that the signal presented in Figure 46 does not included the delay that is produced in the transition detection when a post-processing algorithm is employed.

The performance achieved for each class is shown in Table 34 for the three cases with the most outstanding results. In other words, the results predicted by the Ensemble model with the optimization of the hyperparameters without the use of the post-processing algorithm and with the use of the post-processing algorithm with a window of 18 and 20 samples, without the influence of delay. In general, the best results were obtained with the application of post-processing algorithm with a window of 20 samples. There was an increase of 1.1% in F1-score and 1.7% in MCC for class 1, 1.8% in F1-score and 2.1% in MCC for class 2, 1% in F1-score and 1.5% in MCC for class 3, and, 1.9% in F1-score and 2.2% in MCC for class 4 in relation to that obtained without the application of post-processing algorithm.

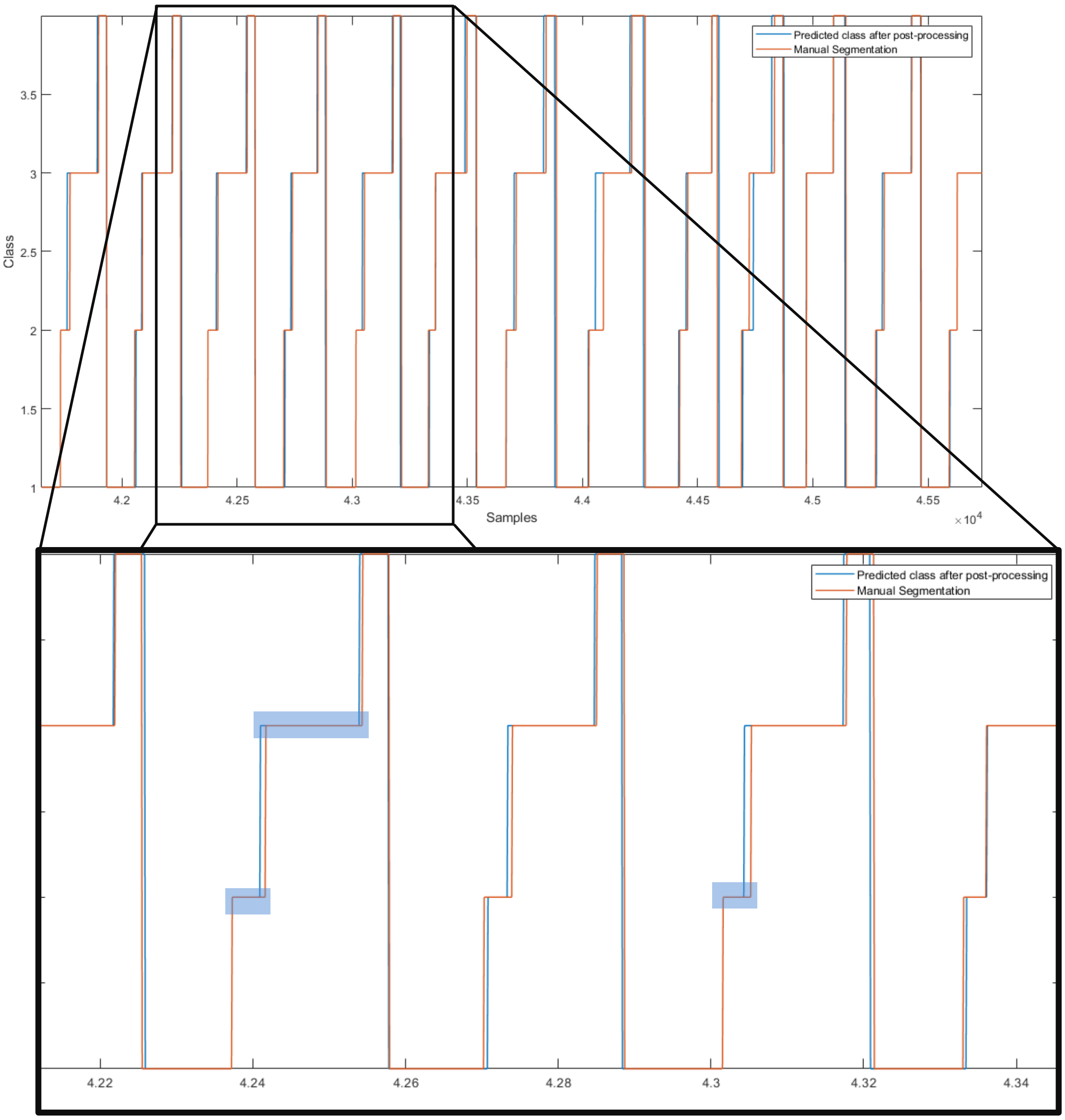


Figure 46 - Comparison between the ground truth and the data predicted by the Ensemble classifier with hyperparameters optimized when using the post-processing algorithm with a window of 20 samples.

Table 34- Evaluation performance results achieved for each class predicted by the Ensemble model with the hyperparameters optimized with used and non-used of post processing algorithm to detect human gait events

Class	Post-processing	Window (Samples)	SENS	SPEC	PREC	F1-score	MCC
TO_{Right}-HS_{Right} (1)	x	-	94.44	96.45	94.12	94.28	90.83
	✓	16	95.90	96.80	94.74	95.32	92.48
	✓	**20**	95.88	**96.82**	**94.76**	**95.33**	**92.50**
HS_{Right}-TO_{Left} (2)	x	-	83.63	97.57	85.35	84.48	81.89
	✓	16	84.79	97.96	87.54	86.14	83.86
	✓	**20**	**85.21**	97.92	87.37	**86.28**	**84.00**
TO_{Left}-HS_{Left} (3)	x	-	94.48	96.22	92.91	93.69	90.34
	✓	16	95.29	96.80	93.98	94.63	91.79
	✓	**20**	**95.42**	96.79	**93.98**	**94.69**	**91.88**
HS_{Left}-TO_{Right} (4)	x	-	85.38	98.19	88.10	86.72	84.69
	✓	16	86.59	98.46	89.82	88.17	86.38
	✓	**20**	**86.61**	**98.60**	**90.69**	**88.60**	**86.89**

7.4. Discussion

With respect to the first step accomplished, which aims to obtain the different models with the best feature set, it is possible to conclude through the validation performance that the LDA and QDA classifiers were the models with worst results. This was also verified when the model was tested with unseen data. Through the performances of the models with unseen data, it was noted that all models built with PCA method yielded worst performances than those obtained with MRMR and Relieff methods. Nonetheless, the model with best result performance was the Ensemble classifier with the 52 most relevant features ranked by the Relieff method (ACC =91.11%, F1-score=88.98% and MCC= 86.04%).

Since it is intended to implement the classifier in real time, it is important to explore if it is feasible to minimize computing time. In this regard, Figure 44 was analysed verifying that the model performance with validation data from 18 features remained almost constant. In this sense, three different models with 18, 25 and 30 features were built. However, it was concluded that the model performance worsened considerably. A decrease of 2.6% in ACC, 3.3% in F1-score and 4.1% in MCC was registered for the best of the three models.

Considering all the results mentioned above, the model that was most capable of predict the 2 gait events (HS_{Right}, HS_{Left}, TO_{Right} and TO_{Left}) was the Ensemble with the 52 most relevant features ranked by the Relieff method. With the objective of obtaining the best performance, the hyperparameters of the Ensemble classifier with the 52 most relevant features were optimized. Although there has been an improvement in the model performance, it was slightly significant with an increase of 0.6% in ACC, 0.8% in F1-score and 0.9% in MCC. Despite the small improvement in model performance, the Ensemble classifier with the hyperparameters optimized proved to be the best classification model built for the gait event detection (ACC =91.66%, F1-score= 89.79% and MCC =86.94%).

The application of the post-processing algorithm aimed removing the misclassified samples in a short period of time which were observed in the predicted data. However, this post-processing implies a delay of 10 samples (with a window of 20 samples) in the transition detections. The metrics for assessing post-processing performance were calculated without the delay implied by post-processing. This is because it was noted that the model performance decreased considerably since 10 samples are misclassified at each transition. Otherwise, it was not possible to determine whether the post-processing algorithm effectively removes misclassified samples in a short period of time and its impact on the final results. Based on the results presented in Figures 45 and 46 it is proven that the misclassified samples in a short period of time were in fact removed. The ACC, F1-score and MCC increased by 1.3%, 1.4% and 1.9%, respectively with the post-processing algorithm. The delay caused by the algorithm may or may not be tolerable in real time depending on how accurate the detection of the events is required. For example, in the case of the calculation of spaciotemporal parameters, the presence of misclassified samples in a short period of time will have a strongly negative impact.

An analysis of the metrics obtained for each class shows that, regardless of the use or non-use of post-processing the classes that are better identified are 1 (TO_{Right}-HS_{Right}) and 3 (TO_{Left}-HS_{Left}) with F1-score and MCC above 93% and 90%, respectively. The F1-Score and MCC metrics reached for each class were roughly 94% and 90% for TO_{Right}-HS_{Right} class, 84% and 81% for HS_{Right}-TO_{Left} class, 94% and 90% for TO_{Left}-HS_{Left} class, and, 87%, 85% for HS_{Left}-TO_{Right} class, respectively, without post-processing. With implementation of the post-processing algorithm with a window of 20 samples there was an improvement in overall metrics. Concerning F1-score and MCC, there was an increase of approximately 1.1% and 1.7% for TO_{Right}-HS_{Right}, 1.8% and 2.1% for HS_{Right}-TO_{Left}, 1% and 1.5% for TO_{Left}-HS_{Left}, and 1.9% and

2.2% HS_{Left}-TO_{Right}. The greatest improvements were achieved in the classes that achieved the worst performance without the use of post-processing.

The performance of the models may not have been better since the signals were collected at two very different velocities. However, in the future, the implementation of other algorithms, such as deep learning algorithms, and more robust feature selection methods should be investigated to increase the effectiveness in gait event detection.

CHAPTER 8 – CONCLUSION

Throughout this dissertation, it was evident that the detection of falls and NF are a real concern to society, especially for the elderly and patients with neuromotor diseases. Several factors, intrinsic and extrinsic, contribute to an increased risk of falling. A large number of deaths by accident around the world are related to falls and have an economic implication on the population. In the scientific community, there is a notable focus on the development of fall detection systems as well as warning mechanisms. However, studies to detect the pre-impact phase and prevent the fall have also been addressed [14] [64].

The walkers are frequently used by subjects who present a high risk of falling, hence the need for fall and NF detection methods when using walkers. In scientific literature, as well as in commercial systems, when a high-risk situation is detected, smart walkers generally stop automatically to support their user. New strategies must be built on this basis. Focusing on smart walkers, it was recognized that in the commercial area as research groups the fall detection and prevention is an important topic, and patents have been found in this regard. In this dissertation, the objective is to move forward to these strategies by taking the first steps towards the detection of a NF and an INF, and in addition the detection of two gait events, the TO and the HS.

Inertial data close to the COM were gathered from the subject as well as FSR signals of the heel and toe of each foot. To this purpose, an existing system had to be adapted. The Xsens MTw Awinda was used to collect simultaneously inertial data from pelvis, near of COM, and the data were used to decide the pre-processing approach. This system was also important to develop an algorithm to estimate velocity and displacement from the adapted system data. NF were simulated in three directions (forward, right and left) at two different velocities, comfortable and slow. The slow velocity was crucial for this study, as it was noted that patients who use the ASBGo SW walk at very lower velocities. The comfortable velocity was defined in order to increase the data variability and, thus, allow to conduct a broader study.

Several features were computed mainly based on the inertial data collected in the three axis, in order to provide a better discrimination of the information and distinguish between a normal walking and a NF situation. The methodology followed throughout this dissertation was based on machine learning algorithms. In this sense, initially, all features were normalized between 0 and 1 followed by the application of three feature selection methods (MRMR, Relieff, and PCA). At the computational level, it was only possible, in this dissertation, to implement feature selection methods of the filter type. The classifiers used were, in particular, Ensemble, DT, LDA, QDA, SVM with Gaussian, Linear, Polynomial kernels and the KNN with an equal, inverse and squared inverse distance weighting functions. It is important to mention that algorithms used for the NF, INF and gait events detection changed by time constraints. It was tested several combinations of classifiers and feature selection methods in order to reach the best model. An incremental analysis from 1 to 60 most significant features ranked by the feature selection methods was performed to find the best feature set.

Regarding the NF detection, it was investigated whether it was possible to detect it and identify its direction based on the features computed. With respect to the detection of a NF, the hyperparameters of the three models that obtained the best results with default hyperparameters were optimized. However, it was concluded that the best built model was the Ensemble with the 51 most relevant features ranked by the Relieff method (F1-score= 81.66% and MCC =79.99%) with default hyperparameters. Other strategies to improve the model performance were implemented, such as the oversampling of the minority class and the attribution of a cost to misclassification. Nevertheless, no improvement was produced. The last step was to implement a post-processing algorithm to eliminate situations that were wrongly detected as a NF. Despite the post-processing algorithm detected a NF with a few milliseconds later, it was verified that about 98% of FP were eliminated. The NF were 100% detected with and without the application of post-processing, with 1.48±0.76s and 1.76±0.68s before the subject restores the balance, respectively. The post-processing algorithm can be crucial if a fall prevention strategy is implemented in the future, such as stopping the walker. This alteration can lead to a dangerous situation if the user is not expecting such reaction from the walker. However, this will have to be investigated thoroughly. Concerning the detection of the NF direction, it can be established that no model was able to distinguish it efficiently. The best performance was obtained by the SVM classifier (Gaussian kernel) with the first 60 features ranked by Relieff method (F1-score=58.92% and MCC=38.45%).

The purpose of the INF detection was to investigate if moments before the NF occurred there was any information indicating the danger situation. The methodology implemented was ineffective for INF detection. The best model with data normalized between 0 and 1 obtained F1-score=29.54%. and MCC=20.27%, being clear that it cannot distinguish the normal gait from an INF. Nonetheless, the z-score normalization as well as the oversampling of the minority class were implemented, however, there were no relevant improvements. These results suggest that there is no relevant information prior to the occurrence of the NF, which would allow detect the NF in advance. However, further studies should be addressed.

The purpose of the gait events detection is to identify the TO and HS events that enable the division of the gait cycle into two phases, stance and swing. The GA allows the extraction of other user information that can be important for the medical professionals. In this case, the best combination reached was the Ensemble classifier with the 52 most significant features ranked by the Relieff method with the optimized hyperparameters (F1-score=89.79% and MCC=86.94%). A post-processing algorithm was also implemented to eliminate the misclassified samples in a short period of time verified in the model outcome. This post-processing implies a delay in the detection of transitions. However, its performance was analyzed without considering this delay. Thus, there was an improvement in the model performance but with the consequence of the delay that is imposed (F1-score= 91.23% and MCC = 88.82%).

The work accomplished throughout the dissertation enables to answer the RQ addressed in Chapter 1:

- **RQ1:** Which sensors have the highest potential for detecting a NF and what is the most viable approach for the problem under study?

  The most suitable sensors are the IMU and the FSR because the inertial signals are widely used for the fall detection and the FSR are good option for gait event detection. Furthermore, the ASBGo SW is already equipped with an IMU used for the pelvis area. The approach selected as the most viable for the detection of the gait events, INF and NF was based on machine learning algorithms. Since the dataset presents variability and some patterns and trends are not evident for threshold-based algorithms.

- **RQ2:** What is the minimum number of features necessary, and what is the best combination of feature selection methods and classifier algorithms to detect a NF and its direction while using a walker?

  Regarding the NF detection, the best model built was the Ensemble classifier with the 51 most significant features ranked by the Relieff method (ACC=95.18%, SENS=71.63%, SPEC=99.33%, PREC=94.96%, F1-score=81.66% and MCC=79.99%). This model was able to detect all NF 1.76 ±0.76s before the subject restores the balance. With post-processing, a large percentage of FP were eliminated, and all NF were detected 1.48±0.68s before their end.

Concerning the detection of the NF direction, the best performance result was achieved with the combination of SVM classifier model with Gaussian kernel and the 60 most relevant features ranked by the Relieff method (ACC=58.97%, SENS=58.95%, SPEC=79.51%, PREC=58.96%, F1-score=58.92%, and MCC=38.45%).

- **RQ3:** Is it possible to detect an INF while using a walker?

  Based on the methodology followed, it was not possible to distinguish an INF from normal walking. The best model built was the Ensemble with the 20 most relevant features ranked by the Relieff method (ACC=83.24%, SENS= 24.57%, SPEC= 94.09%, PREC= 43.46%, F1-score= 31.39% and MCC=23.87%). In this case, the normalization applied was the z-score contrary to the other cases.

- **RQ4:** What is the best combination of classifier, feature selection method, and the number of features that presented the best performance in detecting gait events while using a walker?

  The best model performance achieved for the detection of TO and HS events was the Ensemble classifier with the 52 most relevant features ranked by the Relieff method involving the optimization of hyperparameters. The results achieved for each class were:

  - TO_{Right}-HS_{Right}: F1-Score=94.28% and MCC=90.83%
  - HS_{Right}-TO_{Left}: F1-Score=84.48% and MCC=81.89%
  - TO_{Left}-HS_{Left}: F1-Score=93.69% and MCC=90.34%
  - HS_{Left}-TO_{Right}: F1-Score=86.72% and MCC=84.69%

  With the application of the post-processing algorithm with a window of 20 samples it was possible to eliminate most of the misclassified samples in a short period of time with the disadvantage of introducing a delay in the detection of gait events. The results obtained without considering the delay were:

  - TO_{Right}-HS_{Right}: F1-Score=95.33% and MCC=92.50%
  - HS_{Right}-TO_{Left}: F1-Score=86.26% and MCC=84.00%
  - TO_{Left}-HS_{Left}: F1-Score=94.69% and MCC=91.88%
  - HS_{Left}-TO_{Right}: F1-Score=88.60% and MCC=86.89%

7.1. Future Work

For future work, it is imperative to implement the methodology in the ASBGo SW with more sensors and conduct more experimental tests that include other activities, such as walking in inclined surfaces and start/stop walking beyond those already performed. More discriminatory velocities should also be carried out in future experimental tests and, it is vital to conduct these tests with a broader age range to achieve a more robust approach.

Another important aspect that must be taken into account is the gathering of more information from the human body, such as data on the torso, head, legs, thighs, and information that permits determining if the hands are on the handle grip. This information can be an advantage to improve the model performance to detect a NF, but especially to build a model that is able to distinguish the direction of a NF and to reliably detect an INF.

Regarding the machine learning methodology accomplished, embedded and wrapper feature selection methods should be implemented since in this dissertation only filter feature selection methods were employed. The construction of models based on deep learning, such as a multilayer perceptron or a convolutional neural network should be conducted for all the cases under study, mainly, for the detection of the NF direction and INF once the models obtained in this dissertation were not feasible.

It is essential to detect more gait events beyond the HS and TO tackled in this work to augment patient-related information. The elaboration of an algorithm that computes the gait parameters is also one of the future steps that must be considered to improve the GA.

A camera-based approach should be introduced as wearable sensors are not the most comfortable since the subject has to perform the tasks with them and it is necessary to place and remove them after each use of the walker. This procedure becomes even more relevant when the target is people with low mobility and/or physical restrictions due to illness conditions.

Finally, a fall prevention strategy must be addressed to increase the subject's safety and avoid any injury due to the imbalance which could lead to a fall. This strategy can be, for instance, stop the walker and/or change the walker's trajectory to restore the subject's balance.

References

[1] M. Roberts, D. Mongeon, and F. Prince, "Biomechanical parameters for gait analysis: a systematic review of healthy human gait," *Phys. Ther. Rehabil.*, vol. 4, no. 1, p. 6, 2017.

[2] J. Rueterbories, E. G. Spaich, B. Larsen, and O. K. Andersen, "Methods for gait event detection and analysis in ambulatory systems," *Med. Eng. Phys.*, vol. 32, no. 6, pp. 545–552, 2010.

[3] M. Akhtaruzzaman, A. A. Shafie, and M. R. Khan, "Gait Analysis: Systems, Thechnologies, and Importance," *J. Mech. Med. Biol.*, vol. 16, no. 07, p. 1630003, Nov. 2016.

[4] W. Pirker and R. Katzenschlager, "Gait disorders in adults and the elderly: A clinical guide," *Wien. Klin. Wochenschr.*, vol. 129, no. 3–4, pp. 81–95, 2017.

[5] S. Yoshida, "A Global Report on Falls Prevention Epidemiology of Falls."

[6] W. H. Organization, "Falls," 2018. [Online]. Available: http://www.who.int/news-room/fact-sheets/detail/falls. [Accessed: 17-Nov-2018].

[7] E. R. Burns, J. A. Stevens, and R. Lee, "The direct costs of fatal and non-fatal falls among older adults — United States," *J. Safety Res.*, vol. 58, pp. 99–103, 2016.

[8] S. Berry and R. Miller, "Falls: Epidemiology, Pathophysiology, and Relationship to Fracture," *Curr Osteoporos Rep.*, vol. 6, no. 4, pp. 149–154, 2008.

[9] K. L. Perell, A. Nelson, R. L. Goldman, S. L. Luther, N. Prieto-Lewis, and L. Z. Rubenstein, "Fall Risk Assessment Measures: An Analytic Review," *Journals Gerontol. Ser. A Biol. Sci. Med. Sci.*, vol. 56, no. 12, pp. M761–M766, Dec. 2001.

[10] J. Silva and I. Sousa, "Instrumented timed up and go: Fall risk assessment based on inertial wearable sensors," in *2016 IEEE International Symposium on Medical Measurements and Applications (MeMeA)*, 2016, pp. 1–6.

[11] J. Howcroft, J. Kofman, and E. D. Lemaire, "Review of fall risk assessment in geriatric populations using inertial sensors.," *J. Neuroeng. Rehabil.*, vol. 10, no. 1, p. 91, Aug. 2013.

[12] T. Xu, Y. Zhou, and J. Zhu, "New advances and challenges of fall detection systems: A survey," *Appl. Sci.*, vol. 8, no. 3, 2018.

[13] Y. S. Delahoz and M. A. Labrador, "Survey on fall detection and fall prevention using wearable and external sensors," *Sensors (Switzerland)*, vol. 14, no. 10, pp. 19806–19842, 2014.

[14] X. Hu and X. Qu, "Pre-impact fall detection," *Biomed. Eng. Online*, vol. 15, no. 1, p. 61, Dec. 2016.

[15] T. Tamura, T. Yoshimura, M. Sekine, M. Uchida, and O. Tanaka, "A wearable airbag to prevent

fall injuries," *IEEE Trans. Inf. Technol. Biomed.*, vol. 13, no. 6, pp. 910–914, 2009.

[16] H. Bateni and B. Maki, "Assistive Devices for Balance and Mobility : Benefits , Demands , and Adverse Consequences," *Arch Phys Med Rehabil*, vol. 86, no. January, pp. 134–145, 2005.

[17] I. Pang, Y. Okubo, D. Sturnieks, S. R. Lord, and M. A. Brodie, "Detection of Near Falls Using Wearable Devices," *J. Geriatr. Phys. Ther.*, p. 1, 2018.

[18] I. Maidan, T. Freedman, R. Tzemah, N. Giladi, A. Mirelman, and J. M. Hausdorff, "Introducing a new definition of a near fall: Intra-rater and inter-rater reliability," *Gait Posture*, vol. 39, no. 1, pp. 645–647, Jan. 2014.

[19] Xinguo Yu, "Approaches and principles of fall detection for elderly and patient," in *HealthCom 2008 - 10th International Conference on e-health Networking, Applications and Services*, 2008, pp. 42–47.

[20] N. Noury *et al.*, "Fall detection - Principles and methods," in *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*, 2007, pp. 1663–1666.

[21] N. Noury, P. Rumeau, A. K. Bourke, G. ÓLaighin, and J. E. Lundy, "A proposal for the classification and evaluation of fall detectors," *IRBM*, vol. 29, no. 6, pp. 340–349, Dec. 2008.

[22] C.-Y. Hsieh, K.-C. Liu, C.-N. Huang, W.-C. Chu, and C.-T. Chan, "Novel Hierarchical Fall Detection Algorithm Using a Multiphase Fall Model," *Sensors (Basel).*, vol. 17, no. 2, Feb. 2017.

[23] M. Boltz, "Fall Prevention: Assessment, Diagnoses, and Intervention Strategies," in *Evidence-Based Geriatric Nursing Protocols for Best Practice*, 4th Editio., M. Boltz, E. Capezuti, T. Fulmer, and D. Zwicker, Eds. New York: Springer Publishing Company, 2012, pp. 273–276.

[24] T. Cimilli Ozturk, R. Ak, E. Unal Akoglu, O. Onur, S. Eroglu, and M. Saritemur, "Factors Associated With Multiple Falls Among Elderly Patients Admitted to Emergency Department," *Int. J. Gerontol.*, vol. 11, no. 2, pp. 85–89, 2017.

[25] S. I. Sharif, A. B. Al-Harbi, A. M. Al-Shihabi, D. S. Al-Daour, and R. S. Sharif, "Falls in the elderly: assessment of prevalence and risk factors," *Pharm. Pract. (Granada).*, vol. 16, no. 3, p. 1206, Sep. 2018.

[26] J. A. Stevens, J. E. Mahoney, and H. Ehrenreich, "Circumstances and outcomes of falls among high risk community-dwelling older adults," *Inj. Epidemiol.*, vol. 1, no. 1, pp. 1–9, 2014.

[27] D. Kim and S. Ahrentzen, "Environmental and Behavioral Circumstances and Consequences of Falls in a Senior Living Development," *J. Hous. Elderly*, vol. 31, no. 3, pp. 286–301, 2017.

[28] P. Tyrrell and A. K. P. Jones, "Post-stroke pain," in *Management of Post-Stroke Complications*, 2015, pp. 241–275.

[29] A. Tsur and Z. Segal, "Falls in stroke patients: Risk factors and risk management," *Isr. Med. Assoc. J.*, vol. 12, no. 4, pp. 216–219, 2010.

[30] B. Homann *et al.*, "The impact of neurological disorders on the risk for falls in the community dwelling elderly: A case-controlled study," *BMJ Open*, vol. 3, no. 11, pp. 1–9, 2013.

[31] Y. A. M. Grimbergen, M. Munneke, and B. R. Bloem, "Falls in Parkinson's disease," *Curr. Opin. Neurol.*, vol. 17, no. 4, pp. 405–415, 2004.

[32] J. G. Nutt, B. R. Bloem, N. Giladi, M. Hallett, F. B. Horak, and A. Nieuwboer, "Freezing of gait: moving forward on a mysterious clinical phenomenon," *Lancet Neurol.*, vol. 10, no. 8, pp. 734–744, Aug. 2011.

[33] J. Shah *et al.*, "Increased foot strike variability in Parkinson's disease patients with freezing of gait," *Park. Relat. Disord.*, vol. 53, no. April, pp. 58–63, 2018.

[34] M. Stokes and E. Stack, *Physical Management for Neurological Conditions*, 3rd ed. Elsevier Health Sciences, 2011.

[35] P. G. Levine, *Stronger After Stroke*, 2nd ed. Demos Medical Publishing, 2012.

[36] K. Obrant, "Nonpharmacological Prevention of Osteoporotic Fractures," in *Management of Fractures in Severely Osteoporotic Bone*, 1st Editio., K. Obrant, Ed. London: Springer London, 2000, pp. 333–336.

[37] M. Terroso, N. Rosa, A. Torres Marques, and R. Simoes, "Physical consequences of falls in the elderly: A literature review from 1995 to 2010," *European Review of Aging and Physical Activity*, vol. 11, no. 1. pp. 51–59, 13-Apr-2014.

[38] K. Chaccour, R. Darazi, A. H. El Hassani, and E. Andres, "From Fall Detection to Fall Prevention: A Generic Classification of Fall-Related Systems," *IEEE Sens. J.*, vol. 17, no. 3, pp. 812–822, 2017.

[39] X. Kong, L. Meng, and H. Tomiyama, "Fall detection for elderly persons using a depth camera," in *2017 International Conference on Advanced Mechatronic Systems (ICAMechS)*, 2017, pp. 269–273.

[40] Y. Angal and A. Jagtap, "Fall detection system for older adults," in *2016 IEEE International Conference on Advances in Electronics, Communication and Computer Technology (ICAECCT)*, 2016, pp. 262–266.

[41] S. Zhang, Z. Li, Z. Wei, and S. Wang, "An automatic human fall detection approach using RGBD cameras," *Proc. 2016 5th Int. Conf. Comput. Sci. Netw. Technol. ICCSNT 2016*, pp. 781–784, 2017.

[42] G. M. Basavaraj and A. Kusagur, "Vision based surveillance system for detection of human fall," *RTEICT 2017 - 2nd IEEE Int. Conf. Recent Trends Electron. Inf. Commun. Technol. Proc.*, vol. 2018-Janua, pp. 1516–1520, 2018.

[43] J. L. Chua, Y. C. Chang, and W. K. Lim, "Visual based fall detection through human shape variation and head detection," *IMPACT 2013 - Proc. Int. Conf. Multimed. Signal Process. Commun. Technol.*, pp. 61–65, 2013.

[44] C. Y. Lin, S. M. Wang, J. W. Hong, L. W. Kang, and C. L. Huang, "Vision-based fall detection through shape features," *Proc. - 2016 IEEE 2nd Int. Conf. Multimed. Big Data, BigMM 2016*, pp. 237–240, 2016.

[45] K. R. Bhavya, J. Park, H. Park, H. Kim, and J. Paik, "Fall detection using motion estimation and accumulated image map," in *2016 IEEE International Conference on Consumer Electronics-Asia (ICCE-Asia)*, 2016, no. 3, pp. 1–2.

[46] V. D. Nguyen, M. T. Le, A. D. Do, H. H. Duong, T. D. Thai, and D. H. Tran, "An efficient camera-based surveillance for fall detection of elderly people," *Proc. 2014 9th IEEE Conf. Ind. Electron. Appl. ICIEA 2014*, pp. 994–997, 2014.

[47] V. Vaidehi, K. Ganapathy, K. Mohan, A. Aldrin, and K. Nirmal, "Video based automatic fall detection in indoor environment," in *International Conference on Recent Trends in Information Technology, ICRTIT 2011*, 2011, pp. 1016–1020.

[48] C. Taramasco *et al.*, "A Novel Monitoring System for Fall Detection in Older People," *IEEE Access*, vol. 6, pp. 43563–43574, 2018.

[49] C. Nadee and K. Chamnongthai, "Ultrasonic array sensors for monitoring of human fall detection," in *2015 12th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*, 2015, pp. 1–4.

[50] G. Mokhtari, Q. Zhang, and A. Fazlollahi, "Non-wearable UWB sensor to detect falls in smart home environment," *2017 IEEE Int. Conf. Pervasive Comput. Commun. Work. PerCom Work. 2017*, pp. 274–278, 2017.

[51] A. Lopez, D. Perez, F. J. Ferrero, and O. Postolache, "A Real-Time Algorithm to Detect Falls in the Elderly," in *2018 IEEE International Symposium on Medical Measurements and Applications (MeMeA)*, 2018, vol. 8266, pp. 1–5.

[52] J. He, M. Zhou, X. Wang, and Y. Han, "Application of Kalman filter and k-NN classifier in wearable fall detection device," in *2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing,*

Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), 2017, pp. 1–7.

[53] T. de Quadros, A. E. Lazzaretti, and F. K. Schneider, "A Movement Decomposition and Machine Learning-Based Fall Detection System Using Wrist Wearable Device," *IEEE Sens. J.*, vol. 18, no. 12, pp. 5082–5089, Jun. 2018.

[54] I. L. de Araujo, L. Dourado, L. Fernandes, R. M. D. C. Andrade, and P. A. C. Aguilar, "An Algorithm for Fall Detection using Data from SmartWatch," in *2018 13th Annual Conference on System of Systems Engineering (SoSE)*, 2018, pp. 124–131.

[55] Y. Tao, H. Qian, M. Chen, X. Shi, and Y. Xu, "A Real-time intelligent shoe system for fall detection," in *2011 IEEE International Conference on Robotics and Biomimetics*, 2011, pp. 2253–2258.

[56] L. Montanini, A. Del Campo, D. Perla, S. Spinsante, and E. Gambi, "A Footwear-Based Methodology for Fall Detection," *IEEE Sens. J.*, vol. 18, no. 3, pp. 1233–1242, Feb. 2018.

[57] T. Sivaranjani, L. DhiviyaLakshmi, R. Yogaaravinth, J. Srivishnu, M. M. S. Karthick, and A. Praveenkumar, "Fall assessment and its injury prevention using a wearable airbag technology," in *2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, 2017, pp. 2539–2541.

[58] G. Rescio, A. Leone, and P. Siciliano, "Supervised machine learning scheme for electromyography-based pre-fall detection system," *Expert Syst. Appl.*, vol. 100, pp. 95–105, Jun. 2018.

[59] Jian Liu and T. E. Lockhart, "Development and Evaluation of a Prior-to-Impact Fall Event Detection Algorithm," *IEEE Trans. Biomed. Eng.*, vol. 61, no. 7, pp. 2135–2140, Jul. 2014.

[60] O. Aziz, E. J. Park, G. Mori, and S. N. Robinovitch, "Distinguishing near-falls from daily activities with wearable accelerometers and gyroscopes using Support Vector Machines," in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2012, pp. 5837–5840.

[61] T. Iluz *et al.*, "Automated detection of missteps during community ambulation in patients with Parkinson's disease: a new approach for quantifying fall risk in the community setting," *J. Neuroeng. Rehabil.*, vol. 11, no. 1, p. 48, 2014.

[62] N. H. Chehade, P. Ozisik, J. Gomez, F. Ramos, and G. Pottie, "Detecting stumbles with a single accelerometer," in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2012, pp. 6681–6686.

[63] J. M. H. Karel *et al.*, "Towards unobtrusive in vivo monitoring of patients prone to falling," in *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, 2010, pp. 5018–5021.

[64] S. Chaudhuri, H. Thompson, and G. Demiris, "Fall Detection Devices and Their Use With Older Adults," *J. Geriatr. Phys. Ther.*, vol. 37, no. 4, pp. 178–196, 2014.

[65] J. Liu and T. E. Lockhart, "Age-related joint moment characteristics during normal gait and successful reactive-recovery from unexpected slip perturbations," *Gait Posture*, vol. 30, no. 3, pp. 276–281, Oct. 2009.

[66] J. D. Hsu, J. W. Michael, and J. R. Fisk, "Canes, crutches, and walkers," in *AAOS Atlas of Orthoses and Assistive Device*, 4th Editio., J. D. Hsu, J. W. Michael, and J. R. Fisk, Eds. Elsevier Health Sciences, 2008, pp. 533–535.

[67] C. A. Cifuentes and A. Frizera, "Assistive Device for Human Mobility and Gait Rehabilitation," in *Human-Robot Interaction Strategies for Walker-Assisted Locomotion*, vol. 115, Cham: Springer International Publishing, 2016, pp. 3–9.

[68] P. Paul and B. Williams, *Brunner & Suddarth's Textbook of Canadian Medical-surgical Nursing*. Lippincott Williams & Wilkins, 2009.

[69] T. F. Bastos, K. Dinesh, and S. P. Arjunan, *Devices for Mobility and Manipulation for People with Reduced Abilities*. CRC Press, 2014.

[70] M. Martins, C. Santos, and A. Frizera, "Online control of a mobility assistance Smart Walker," in *2012 IEEE 2nd Portuguese Meeting in Bioengineering (ENBENG)*, 2012, pp. 1–6.

[71] M. Martins, C. Santos, E. Seabra, L. Basilio, and A. Frizera, "A new integrated device to read user intentions when walking with a Smart Walker," in *2013 11th IEEE International Conference on Industrial Informatics (INDIN)*, 2013, pp. 299–304.

[72] A. Tereso, M. Martins, C. P. Santos, M. Vieira da Silva, L. Gonçalves, and L. Rocha, "Detection of Gait Events and Assessment of Fall Risk Using Accelerometers in Assisted Gait," in *2014 11th International Conference on Informatics in Control, Automation and Robotics (ICINCO)*, 2014, pp. 788–793.

[73] M. Martins, C. Santos, A. Frizera, and R. Ceres, "Real time control of the ASBGo walker through a physical human-robot interface," *Meas. J. Int. Meas. Confed.*, vol. 48, no. 1, pp. 77–86, 2014.

[74] V. Faria, J. Silva, M. Martins, and C. Santos, "Dynamical system approach for obstacle avoidance in a Smart Walker device," in *2014 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 2014, pp. 261–266.

[75] Y. Hirata, S. Komatsuda, and K. Kosuge, “Fall prevention control of passive intelligent walker based on human model,” *2008 IEEE/RSJ Int. Conf. Intell. Robot. Syst. IROS*, pp. 1222–1228, 2008.

[76] S. Taghvaei, Y. Hirata, and K. Kosuge, “Vision-based human state estimation to control an intelligent passive walker,” in *2010 IEEE/SICE International Symposium on System Integration*, 2010, pp. 146–151.

[77] S. Taghvaei and K. Kosuge, “Image-based fall detection and classification of a user with a walking support system,” *Front. Mech. Eng.*, vol. 13, no. 3, pp. 427–441, 2018.

[78] W. Xu, J. Huang, and L. Cheng, “A Novel Coordinated Motion Fusion-Based Walking-Aid Robot System,” *Sensors*, vol. 18, no. 9, p. 2761, Aug. 2018.

[79] S. Irgenfried and H. Wörn, “Motion Control and Fall Prevention for an Active Walker Mobility Aid,” in *Mechanisms and Machine Science*, V. Petuya, C. Pinto, and E.-C. Lovasz, Eds. Dordrecht: Springer Netherlands, 2014, pp. 157–164.

[80] J. Huang, W. Xu, S. Mohammed, and Z. Shu, “Posture estimation and human support using wearable sensors and walking-aid robot,” *Rob. Auton. Syst.*, vol. 73, pp. 24–43, Nov. 2015.

[81] W.-H. Mou, M.-F. Chang, C.-K. Liao, Y.-H. Hsu, S.-H. Tseng, and L.-C. Fu, “Context-aware assisted interactive robotic walker for Parkinson’s disease patients,” in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 329–334.

[82] M. Azqueta-Gavaldon, I. Azqueta-Gavaldon, M. Woiczinski, K. Bötzel, and E. Kraft, “Automatic Braking System and Fall Detection Mechanism for Rollators,” in *Proceedings of the 6th International Conference on Bioinformatics and Biomedical Science - ICBBS ’17*, 2017, pp. 158–161.

[83] RT.Works, “Robot assist device RT.1.” [Online]. Available: https://www.rtworks.co.jp/eng/product/rt1.html.

[84] RT.Works, “Robot Assist Walker RT.2.” [Online]. Available: https://www.rtworks.co.jp/eng/product/rt2.html.

[85] M. P. Boelen, *Health professionals’ guide to physical management of Parkinson’s disease*. Human Kinetics, 2009.

[86] T. J. Lulic and O. Muftic, “Trajectory of The Human Body Mass Centre During Walking at Different Sspeed,” in *International Design Conference*, 2002, pp. 797–802.

[87] B. Auvinet *et al.*, “Reference data for normal subjects obtained with an accelerometric device,” *Gait Posture*, vol. 16, no. 2, pp. 124–134, 2002.

[88] H. Gonçalves, "Functional Feedback Vibrotactile System for Patients with Parkinson ' s Disease : Freezing of Gait," University of Minho, 2017.

[89] E. Engineering and D. Teichmann, "An Analysis on Sensor Locations of the Human Body for Wearable Fall Detection Devices : Principles," *Sensor*, vol. 16, no. 8, pp. 1–25, 2016.

[90] A. Olivares, J. M. Górriz, J. Ramírez, and G. Olivares, "Accurate Human Limb Angle Measurement in Telerehabilitation: Sensor Fusion through Kalman, LMS and RLS Adaptive Filtering," in *Ambient Intelligence and Future Trends-International Symposium on Ambient Intelligence (ISAmI 2010)*, 2010, pp. 97–104.

[91] J. Windau and L. Itti, "Situation awareness via sensor-equipped eyeglasses," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 5674–5679.

[92] R. M. Rangayyan, *Biomedical Signal Analysis*. Wiley, 2015.

[93] A. C. Muller and S. Guido, *Introduction to Machine Learning with Python and Scikit-Learn*. 2015.

[94] S. J. Russell and P. Norvig, *Artificial Intelligence A Modern Approach*, Third Edit. Pearson Education, Inc., 2003.

[95] S. Kotsiantis, P. E. Pintelas, and D. Kanellopoulos, "Data Preprocessing for Supervised Learning," *Int. J. Comput. Sci.*, vol. 1, pp. 111–117, Jan. 2006.

[96] J. Han and M. Kamber, "Data Preprocessing," in *Data Mining, Southeast Asia Edition*, 2nd ed., Elsevier Science, 2006, pp. 70–72.

[97] M. A. Hall and G. Holmes, "Benchmarking attribute selection techniques for discrete class data mining," *IEEE Trans. Knowl. Data Eng.*, vol. 15, no. 6, pp. 1437–1447, Nov. 2003.

[98] S. B. Kotsiantis, "Feature selection for machine learning classification problems: a recent overview," *Artif. Intell. Rev.*, vol. 42, no. 1, Jun. 2011.

[99] A. Danasingh, S. Balarmurugan, and J. Epiphany, "Literature Review on Feature Selection Methods for High-Dimensional Data," *Int. J. Comput. Appl.*, vol. 136, no. 1, pp. 9–17, Feb. 2016.

[100] G. Roffo, "Feature Selection Library (MATLAB Toolbox)," 2016.

[101] A. Muller and S. Guido, "Supervised Learning," in *Inroduction to Machine Learning with Python*, First Edit., D. Schanafelt, Ed. "O'Reilly Media, Inc.", 2016, 2016, pp. 25–129.

[102] S. Bruney, "Supervised Classification," in *Process, Data and Classifier Models for Accessible Supervised Classification Problem Solving*, Vubpress, 2010, pp. 9–22.

[103] G. Rebala, A. Ravi, and S. Churiwala, "Clustering," in *An Introduction to Machine Learning*, Cham: Springer International Publishing, 2019, pp. 67–76.

[104] O. Maimon and L. Rokach, “Introduction to Decision Tree,” in *Data Mining With Decision Trees: Theory and Applications*, 2nd Editio., World Scientific, 2014, pp. 1–16.

[105] L. Rokach, “Ensemble-based classifiers,” *Artif. Intell. Rev.*, vol. 33, no. 1–2, pp. 1–39, Feb. 2009.

[106] J. Brownlee, *Master Machine Learning Algorithms: Discover How They Work and Implement Them From Scratch*. Machine Learning Mastery, 2017.

[107] L. Rokach and O. Maimon, “Evaluation of Classification Trees,” in *Data Mining witj Decision Trees*, 2nd editio., World Scientific, 2015, pp. 31–60.

[108] J. Perry, *Gait Analysis: Normal and Pathological Function*. SLACK, 1992.

[109] T. Roman de Mettelinge and D. Cambier, “Understanding the Relationship Between Walking Aids and Falls in Older Adults,” *J. Geriatr. Phys. Ther.*, vol. 38, no. 3, pp. 127–132, 2015.

[110] A. Muro-de-la-Herran, B. Garcia-Zapirain, and A. Mendez-Zorrilla, “Gait Analysis Methods: An Overview of Wearable and Non-Wearable Systems, Highlighting Clinical Applications,” *Sensors*, vol. 14, no. 2, pp. 3362–3394, Feb. 2014.

[111] J. Hamill and K. M. Knutzen, *Biomechanical Basis of Human Movement*. Lippincott Williams & Wilkins, 2006.

[112] W. Tao, T. Liu, R. Zheng, and H. Feng, “Gait analysis using wearable sensors,” *Sensors*, vol. 12, no. 2, pp. 2255–2283, 2012.

[113] D. D. Neumann, *Kinesiology of the Musculoskeletal System: Foundations for Rehabilitation*, 2nd Editio. Mosby/Elsevier, 2010.

[114] B. Mariani, “Assessment of Foot Signature Using Wearable Sensors for Clinical Gait Analysis and Real-Time Activity Recognition,” École Polytechinque Fédérale de Lausanne, 2012.

[115] J. Taborri, E. Palermo, S. Rossi, and P. Cappa, “Gait Partitioning Methods: A Systematic Review,” *Sensors*, vol. 16, no. 1, p. 66, Jan. 2016.

[116] J. Jacob *et al.*, “A fall detection study on the sensors placement location and a rule-based multi-thresholds algorithm using both accelerometer and gyroscopes,” in *2011 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE 2011)*, 2011, pp. 666–671.

[117] A. Özdemir and B. Barshan, “Detecting Falls with Wearable Sensors Using Machine Learning Techniques,” *Sensors*, vol. 14, no. 6, pp. 10691–10708, Jun. 2014.

[118] I. Putra, B. James, E. Gaura, and V. Rein, “An Event-Triggered Machine Learning Approach for Accelerometer-Based Fall Detection,” *Sensors*, vol. 18, pp. 1–18, 2017.

[119] I. Cleland *et al.*, “Optimal Placement of Accelerometers for the Detection of Everyday Activities,”

Sensors, vol. 13, pp. 9183–9200, 2013.

[120] M. Kangas, A. Konttila, I. Winblad, and T. Jämsä, "Determination of simple thresholds for accelerometry-based parameters for fall detection," *Conf. Proc. IEE EMBS*, pp. 1367–1370, 2007.

[121] C. Lai, S. Chang, H. Chao, and Y.-M. Huang, "Detection of Cognitive Injured Body Region Using Multiple Triaxial Accelerometers for Elderly Falling," *IEEE Sens. J.*, vol. 11, no. 3, pp. 763–770, Mar. 2011.

[122] F. Bianchi, S. J. Redmond, M. R. Narayanan, S. Cerutti, N. H. Lovell, and S. Member, "Barometric Pressure and Triaxial Accelerometry- Based Falls Event Detection," *IEEE Trans. NEURAL Syst. Rehabil. Eng.*, vol. 18, no. 6, pp. 619–627, 2010.

[123] M. Iosa, T. Marro, S. Paolucci, and D. Morelli, "Research in Developmental Disabilities Stability and harmony of gait in children with cerebral palsy," *Res. Dev. Disabil.*, vol. 33, no. 1, pp. 129–135, 2012.

[124] S. Hwang, M. Ryu, Y. Yang, and N. Lee, "Fall Detection with Three-Axis Accelerometer and Magnetometer in a Smartphone," pp. 65–70.

[125] J. Dai, X. Bai, Z. Yang, Z. Shen, and D. Xuan, "PerFallD : A Pervasive Fall Detection System Using Mobile Phones," in *2010 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*, 2010, pp. 292–297.

[126] S. Zhao, W. Li, W. Niu, R. Gravina, and G. Fortino, "Recognition of Human Fall Events Based on Single Tri-axial Gyroscope," in *2018 IEEE 15th International Conference on Networking, Sensing and Control (ICNSC)*, 2018.

[127] S. Ranakoti *et al.*, *Human Fall Detection System over IMU Sensors Using Triaxial Accelerometer*, vol. I. Springer Singapore, 2019.

[128] K. Chen, J. Yang, and F. Jaw, "Accelerometer-Based Fall Detection using Feature Extraction and Support Accelerometer-based fall detection using feature extraction and support vector machine algorithms," *Instrum. Sci. Technol.*, vol. 44, no. 4, pp. 333–342, 2016.

[129] S. Liu and W.-C. Cheng, "Fall Detection with the Support Vector Machine during Scripted and Continuous Unscripted Activities," *Sensors*, vol. 12, pp. 12301–12316, 2012.

[130] I. P. E. Suardiyana, J. Brusey, and E. Gaura, "A Cascade-Classifier Approach for Fall Detection," in *MOBIHEALTH*, 2015.

[131] E. P. Doheny *et al.*, "Displacement of Centre of Mass During Quiet Standing Assessed Using Accelerometry in Older Fallers and Non-Fallers," in *34th Annual International Conference of the*

IEEE EMBS, 2012, pp. 3300–3303.

APPENDICES

Appendix 1

In this appendix is presented the complete list of features computed for chapter 4,5 and 6.

Table 35 – List of features computed and its description

Feature Number	Feature Label	Description	Article
1	Acc_V	Vertical Acceleration	*
2	Acc_ML	Mediolateral acceleration	*
3	Acc_AP	Anteroposterior acceleration	*
4	Gyro_V	Vertical angular velocity	*
5	Gyro_ML	Mediolateral angular velocity	*
6	Gyro_AP	Anteroposterior angular velocity	*
7	FSR_TR	Force sensor resistor (toe right)	*
8	FSR_HR	Force sensor resistor (heel right)	*
9	FSR_TL	Force sensor resistor (toe left)	*
10	FSR_HL	Force sensor resistor (heel left)	*
11	SVMAcc	Sum vector magnitude of acceleration	[116]
12	SVMGyr	Sum vector magnitude of angular velocity	[116]
13	sk_V_Acc	Skewness of acceleration (Vertical axis)	[117]
14	sk_ML_Acc	Skewness of acceleration (Mediolateral axis)	[117]
15	sk_AP_Acc	Skewness of acceleration (Anteroposterior axis)	[117]
16	sk_SVMAcc	Skewness of SVM of acceleration	[117]
17	sk_V_Gyr	Skewness of angular velocity (Vertical axis)	[117]
18	sk_ML_Gyr	Skewness of angular velocity (Mediolateral axis)	[117]
19	sk_AP_Gyr	Skewness of angular velocity (Anteroposterior axis)	[117]

Table 35 -Continued

Feature Number	Feature Label	Description	Article
20	sk_SVMGyr	Skewness of SVM of angular velocity	[117]
21	kur_V_Acc	Kurtosis of acceleration (Vertical axis)	[117]
22	kur_ML_Acc	Kurtosis of acceleration (Medial Lateral axis)	[117]
23	kur_AP_Acc	Kurtosis of acceleration (Anteroposterior axis)	[117]
24	kur_SVMAcc	Kurtosis of SMV of Accelerarion	[117]
25	kur_V_Gyr	Kurtosis of angular velocity (Vertical axis)	[117]
26	kur_ML_Gyr	Kurtosis of angular velocity (Medial Lateral axis)	[117]
27	kur_AP_Gyr	Kurtosis of angular velocity (Anteroposterior axis)	[117]
28	kur_SVMGyr	Kurtosis of SMV of angular velocity	[117]
29	MinAcc_V	Minimum acceleration (Vertical axis)	[118]
30	MinAcc_ML	Minimum acceleration (Mediolateral axis)	[118]
31	MinAcc_AP	Minimum acceleration (Anteroposterior axis)	[118]
32	MinGyr_V	Minimum angular velocity (Vertical axis)	[118]
33	MinGyr_ML	Minimum angular velocity (Mediolateral axis)	[118]
34	MinGyr_AP	Minimum angular velocity (Anteroposterior axis)	[118]
35	MinSVMAcc	Minimum SVM of acceleration	[118]
36	MinSVMGyr	Minimum SVM of angular velocity	[118]
37	MaxAcc_V	Maximum acceleration (Vertical axis)	[118]
38	MaxAcc_ML	Maximum acceleration (Mediolateral axis)	[118]
39	MaxAcc_AP	Maximum acceleration (Anteroposterior axis)	[118]
40	MaxGyr_V	Maximum angular velocity (Vertical axis)	[118]
41	MaxGyr_ML	Maximum angular velocity (Mediolateral axis)	[118]
42	MaxGyr_AP	Maximum angular velocity (Anteroposterior axis)	[118]
43	MaxSVMAcc	Maximum SVM of acceleration	[118]

Table 35 - Continued

Feature Number	Feature Label	Description	Article
44	MaxSVMGyr	Maximum SVM of angular velocity	[118]
45	MeanAcc_V	Mean Acceleration (Vertical axis)	[117]
46	MeanAcc_ML	Mean Acceleration (Mediolateral axis)	[117]
47	MeanAcc_AP	Mean Acceleration (Vertical axis)	[117]
48	MeanGyr_V	Mean Angular Velocity (Vertical axis)	[117]
49	MeanGyr_ML	Mean Angular Velocity (Mediolateral axis)	[117]
50	MeanGyr_AP	Mean Angular Velocity (Anterior posterior axis)	[117]
51	MeanSVMAcc	Mean SVM of Acceleration	[117]
52	MeanSVMGyr	Mean SVM of Angular Velocity	[117]
53	VarAcc_V	Variance of Acceleration (Vertical axis)	[117]
54	VarAcc_ML	Variance of Acceleration (Mediolateral axis)	[117]
55	VarAcc_AP	Variance of Acceleration (Anteroposterior axis)	[117]
56	VarGyr_V	Variance of Angular Velocity (Vertical axis)	[117]
57	VarGyr_ML	Variance of Angular Velocity (Mediolateral axis)	[117]
58	VarGyr_AP	Variance of Angular Velocity (Anteroposterior axis)	[117]
59	VarSVMAcc	Variance of SVM of Acceleration	[117]
60	VarSVMGyr	Variance of SVM of Angular Velocity	[117]
61	StdAcc_V	Standard Deviation of Acceleration (Vertical axis)	*
62	StdAcc_ML	Standard Deviation of Acceleration (Mediolateral axis)	*
63	StdAcc_AP	Standard Deviation of Acceleration (Anteroposterior axis)	*
64	StdGyr_V	Standard Deviation of Angular Velocity (Vertical axis)	*

Table 35 - Continued

Feature Number	Feature Label	Description	Article
65	StdGyr_ML	Standard Deviation of Angular Velocity (Mediolateral axis)	*
66	StdGyr_AP	Standard Deviation of Angular Velocity (Anteroposterior axis)	*
67	StdSVMAcc	Standard Deviation of Acceleration SVM	*
68	StdSVMGyr	Standard Deviation of Angular Velocity SVM	*
69	CorrelationAcc_V_ML	Correlation Between Acceleration Vertical and Mediolateral axis	[119]
70	CorrelationAcc_V_AP	Correlation Between Acceleration Vertical and Anteroposterior axis	[119]
71	CorrelationAcc_ML_AP	Correlation Between Acceleration Medial Lateral and Anteroposterior axis	[119]
72	CorrelationGyr_V_ML	Correlation Between Angular Velocity Vertical and Mediolateral axis	[119]
73	CorrelationGyr_V_AP	Correlation Between Angular Velocity Vertical and Anteroposterior axis	[119]
74	CorrelationGyr_ML_AP	Correlation Between Angular Velocity Medial Lateral and Anteroposterior axis	[119]
75	HP_Filter_AP	Anteroposterior acceleration with high pass filter	*
76	HP_Filter_ML	Mediolateral acceleration with high pass filter	*
77	HP_Filter_V	Vertical acceleration with high pass filter	*
78	SVMRAW		*
79	Dynamic_Sum_Vector	Dynamic Sum Vector	[120]
80	Vertical_Acceleration	Vertical Acceleration	[120]
81	Total_angular_Change	Total angular change	[116]
82	Resultant_Angular_Acceleration	Resultant angular acceleration	[116]

Table 35 - Continued

Feature Number	Feature Label	Description	Article
83	asma	Activity Signal Magnitude Area	[121]
84	SMA	Signal Magnitude Area	[122]
85	PPV_Acc_V	Peak-to-peak values of Acceleration (Vertical axis)	[123]
86	PPV_Acc_ML	Peak-to-peak values of Acceleration (Mediolateral axis)	[123]
87	PPV_Acc_AP	Peak-to-peak values of Acceleration (Anteroposterior axis)	[123]
88	PPV_Gyr_V	Peak-to-peak values of Angular Velocity (Vertical axis)	[123]
89	PPV_Gyr_ML	Peak-to-peak values of Angular Velocity (Mediolateral axis)	[123]
90	PPV_Gyr_AP	Peak-to-peak values of Angular Velocity (Anteroposterior axis)	[123]
91	PPV_SVMAcc	Peak-to-peak values of SVM of Acceleration	[123]
92	PPV_SVMGyr	Peak-to-peak values of SVM of Angular Velocity	[123]
93	RMS_Acc_V	Root Mean Square of Acceleration (Vertical axis)	[123]
94	RMS_Acc_ML	Root Mean Square of Acceleration (Mediolateral axis)	[123]
95	RMS_Acc_AP	Root Mean Square of Acceleration (Anteroposterior axis)	[123]
96	RMS_SVMAcc	Root Mean Square of Angular Velocity (Vertical axis)	[123]
97	RMS_Gyr_V	Root Mean Square of Angular Velocity (Mediolateral axis)	[123]
98	RMS_Gyr_ML	Root Mean Square of Angular Velocity (Anteroposterior axis)	[123]

Table 35 - Continued

Feature Number	Feature Label	Description	Article
99	RMS_Gyr_AP	Root Mean Square of SVM of Acceleration	[123]
100	RMS_SVMGyr	Root Mean Square of SVM of Angular Velocity	[123]
101	RI_Acc_V	Ratio Index of Acceleration (Vertical axis)	[123]
102	RI_Acc_ML	Ratio Index of Acceleration (Mediolateral axis)	[123]
103	RI_Acc_AP	Ratio Index of Acceleration (Anteroposterior axis)	[123]
104	RI_AccSVM	Ratio Index of SVM of Acceleration	[123]
105	RI_Gyr_V	Ratio index of Angular Velocity (Vertical axis)	[123]
106	RI_Gyr_ML	Ratio index of Angular Velocity (Mediolateral axis)	[123]
107	RI_Gyr_AP	Ratio index of Angular Velocity (Anteroposterior axis)	[123]
108	RI_GyrSVM	Ratio Index of SVM of Angular Velocity	[123]
109	RI_PPV_Acc_V	Ratio Index of Peak-to-peak of Acceleration (Vertical axis)	[123]
110	RI_PPV_Acc_ML	Ratio Index of Peak-to-peak of Acceleration (Mediolateral axis)	[123]
111	RI_PPV_Acc_AP	Ratio Index of Peak-to-peak of Acceleration (Anteroposterior axis)	[123]
112	RI_PPV_Gyr_V	Ratio Index of Peak-to-peak of Angular Velocity (Vertical axis)	[123]
113	RI_PPV_Gyr_ML	Ratio Index of Peak-to-peak of Angular Velocity (Mediolateral axis)	[123]
114	RI_PPV_Gyr_AP	Ratio Index of Peak-to-peak of Angular Velocity (Vertical axis)	[123]
115	RI_PPV_SVM_Acc	Ratio Index of Peak-to-peak of SVM of Acceleration	*
116	RI_PPV_SVM_Gyr	Ratio Index of Peak-to-peak of SVM of Angular Velocity	*

Table 35 - Continued

Feature Number	Feature Label	Description	Article
117	Quaternion_1	First element of quaternion vector	[124]
118	Quaternion_2	Second element of quaternion vector	[124]
119	Quaternion_3	Third element of quaternion vector	[124]
120	Quaternion_4	Fourth element of quaternion vector	[124]
121	Roll	Roll (MahonyAHRS)	[124]
122	Pitch	Pitch (MahonyAHRS)	[124]
123	Yaw	Yaw (MahonyAHRS)	[124]
124	AbsVerticalAcc	Absolute vertical acceleration	[125]
125	RAC	SVM of Resultant angle change	[126]
126	RAC_AP	Resultant angle change (Anteroposterior axis)	[126]
127	RAC_ML	Resultant angle change (Mediolateral axis)	[126]
128	RAC_V	Resultant angle change (Vertical axis)	[126]
129	MRAA	Maximum resultant angular acceleration	[126]
130	FF_D	Sum of Fluctuation Frequency of all axis	[126]
131	FF_D_AP	Fluctuation Frequency (Anteroposterior axis)	[126]
132	FF_D_ML	Fluctuation Frequency (Mediolateral axis)	[126]
133	FF_D_V	Fluctuation Frequency (Vertical axis)	[126]
134	RAAcc	SVM of Resultant of Average Acceleration	[127]
135	RAAcc_AP	Resultant of Average Acceleration (Anteroposterior axis)	[127]
136	RAAcc_ML	Resultant of Average Acceleration (Mediolateral axis)	[127]
137	RAAcc_V	Resultant of Average Acceleration (Vertical axis)	[127]
138	RSTD	SVM of Resultant of Standard Deviation	[127]
139	RSTD_AP	Resultant of Standard Deviation (Anteroposterior axis)	[127]

Table 35 - Continued

Feature Number	Feature Label	Description	Article
140	RSTD_ML	Resultant of Standard Deviation (Mediolateral axis)	[127]
141	RSTD_V	Resultant of Standard Deviation (Vertical axis)	[127]
142	Slope	Slope	[128]
143	Fast_Change_Vector	Fast Change Vector	[129]
144	SVMAcc_Horizontal_Plane	SVM of Acceleration in the horizontal Plane	[128]
145	EMA	Exponential moving average	[130]
146	RA	Rotational Angle of Acceleration SVM	[128]
147	Z_score	Z-Score	[127]
148	MAD	Magnitude of Angular Displacement	[124]
149	RDCAcc	Acceleration Resultant of Delta Changes	[127]
150	RDCGyr	Angular Velocity Resultant of Delta Changes	[127]
151	GravityComponentAcc_V	Gravity component of Acceleration (Vertical Axis)	*
152	GravityComponentAcc_ML	Gravity component of Acceleration (Mediolateral Axis)	*
153	GravityComponentAcc_AP	Gravity component of Acceleration (Anteroposterior Axis)	*
154	Vel_V	Vertical Velocity	[131]
155	Vel_ML	Mediolateral Velocity	[131]
156	Vel_AP	Anteroposterior Velocity	[131]
157	d_V	Vertical Displacement	[131]
158	d_ML	Mediolateral Displacement	[131]
159	d_AP	Anteroposterior Displacement	[131]
160	DCOM	Cumulative horizontal displacement	[131]
161	SL_V	Vertical Cumulative horizontal sway length	[131]

Table 35 - Continued

Feature Number	Feature Label	Description	Article
162	SL_ML	Mediolateral Cumulative horizontal sway length	[131]
163	SL_AP	Anteroposterior Cumulative horizontal sway length	[131]
164	SV_V	Vertical Mean sway velocity	[131]
165	SV_ML	Mediolateral Mean sway velocity	[131]
166	SV_AP	Anteroposterior Mean sway velocity	[131]
167	DR_V	Vertical Displacement Range	[131]
168	DR_ML	Mediolateral Displacement Range	[131]
169	DR_AP	Anteroposterior Displacement Range	[131]

Appendix 2

This appendix presents the cross-validation results concerning chapter 5.

Case 1

Table 36 – The sixty most relevant features ranked by MRMR, Relieff and PCA methods to detect a NF. Each number corresponds to the respective feature of Table 35

Rank	MRMR	Relieff	PCA
1	47	7	72
2	29	9	71
3	157	8	69
4	139	10	74
5	32	141	73
6	156	120	70
7	30	118	132
8	164	164	133
9	145	167	131
10	7	161	7
11	3	138	10

Table 36 - Continued

Rank	MRMR	Relieff	PCA
12	154	145	9
13	2	157	8
14	5	122	119
15	135	119	122
16	159	139	130
17	39	154	17
18	161	140	19
19	46	123	18
20	31	121	117
21	167	36	141
22	126	117	148
23	136	153	145
24	41	97	138
25	49	81	139
26	33	52	121
27	45	99	140
28	127	136	14
29	124	134	161
30	140	137	164
31	99	158	167
32	34	44	134
33	36	32	35
34	128	148	51
35	134	135	123
36	165	42	118
37	35	48	135
38	97	12	96
39	162	156	100
40	52	40	84

Table 36 - Continued

Rank	MRMR	Relieff	PCA
41	168	159	125
42	77	50	99
43	81	4	52
44	95	34	81
45	138	6	44
46	75	155	95
47	163	98	36
48	169	29	93
49	147	41	43
50	51	165	160
51	44	49	144
52	166	147	12
53	151	168	97
54	76	162	15
55	153	35	83
56	80	169	94
57	38	163	11
58	84	166	90
59	152	93	98
60	98	39	31

Table 37 – Cross-validation evaluation performance results of different machine learning models trained with the features ranked by the MRMR method to detect a NF

Classifier	ACC	SENS	SPEC	PREC	F1 - Score	MCC	Number of Features
LDA	88.23	52.31	94.77	64.52	57.78	51.41	10
QDA	90.96	63.31	95.99	74.18	68.31	63.35	3
DT	98.70	95.36	99.31	96.17	95.76	95.00	45
Ensemble	99.37	96.25	99.94	99.67	97.93	97.58	51
KNN Equal	99.89	99.63	99.94	99.68	99.66	99.59	60

Table 37- Continued

Classifier	ACC	SENS	SPEC	PREC	F1 - Score	MCC	Number of Features
KNN Inverse	99.89	99.63	99.94	99.68	99.66	99.59	60
KNN Squared Inverse	99.89	99.63	99.94	99.68	99.66	99.59	60

Table 38 – Cross-validation evaluation performance results of different machine learning models trained with the features ranked by the Relieff method to detect a NF

Classifier	ACC	SENS	SPEC	PREC	F1 - Score	MCC	Number of Features
LDA	87.66	61.40	92.44	59.62	60.50	53.20	23
QDA	89.79	72.27	92.98	65.18	68.54	62.58	23
DT	99.21	97.15	99.59	97.71	97.43	96.96	20
Ensemble	99.87	99.39	99.96	99.79	99.59	99.52	41
KNN Equal	99.93	99.71	99.97	99.81	99.76	99.71	59
KNN Inverse	99.93	99.71	99.97	99.81	99.76	99.71	59
KNN Squared Inverse	99.93	99.71	99.97	99.81	99.76	99.71	59

Table 39 - Cross-validation evaluation performance results of different machine learning models trained with the features ranked by the PCA method to detect a NF

Classifier	ACC	SENS	SPEC	PREC	F1 - Score	MCC	Number of Features
LDA	87.78	49.86	94.68	63.02	55.68	49.15	23
QDA	88.18	52.28	94.71	64.23	57.64	51.22	23
DT	99.09	96.72	99.52	97.37	97.04	96.51	37
Ensemble	99.85	99.23	99.9	99.80	99.51	99.43	30
KNN Equal	94.30	76.11	97.61	85.25	80.42	77.27	60
KNN Inverse	94.30	76.11	97.61	85.25	80.42	77.27	60
KNN Squared Inverse	94.30	76.11	97.61	85.25	80.42	77.27	60

Table 40 – Cross-validation evaluation performance result of the DT model with the hyperparameters optimized to detect a NF

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1-score	MCC	Number of Features
MRMR	DT	98.70	95.28	99.32	96.21	95.74	94.97	45

Table 41 – Cross-validation evaluation performance results of the KNN Equal with the 60 most relevant features ranked by the MRMR method for different values of k to detect a NF

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1- score	MCC	k
MRMR	KNN Equal	99.65	99.88	99.61	97.82	98.84	98.64	4
		99.19	94.93	99.97	99.80	97.30	96.87	8
		99.23	95.36	99.94	99.64	97.45	97.03	9
		98.98	93.73	99.93	99.60	96.57	96.03	11
		98.72	92.09	99.93	99.58	95.69	95.04	13
		98.44	90.31	99.92	99.53	94.70	93.94	15
		97.65	85.02	99.95	99.68	91.76	90.79	20

Table 42 – Cross-validation evaluation performance result of the Ensemble model with the hyperparameters optimized to detect a NF

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1- score	MCC	Number of Features
MRMR	Ensemble	99.33	95.99	99.94	99.64	95.74	94.97	51

Table 43 – Cross-validation evaluation performance result of the Ensemble model with data oversampling to detect a NF

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1-score	MCC	Number of Features
MRMR	Ensemble	99.83	99.63	99.90	99.73	99.68	99.56	51

Table 44 – Cross-validation evaluation performance results of the Ensemble model with the 51 most relevant features ranked by the MRMR method for different misclassification cost to detect a NF

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1- score	MCC	Cost
MRMR	Ensemble	99.53	98.04	99.80	98.90	98.47	98.19	5
		99.55	97.73	99.88	99.30	98.51	98.25	3
		99.45	98.50	99.63	97.97	98.23	97.91	10

Table 45 – Cross-validation evaluation performance result of the Ensemble model with the 15 most relevant features ranked by the MRMR method to detect a NF

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1- score	MCC
MRMR	Ensemble	99.18	94.89	99.96	99.75	97.26	96.82

Case 2

Table 46 - The sixty most relevant features ranked by MRMR, Relieff and PCA methods to detect the NF direction. Each number corresponds to the respective feature of Table 35

Rank	MRMR	Relieff	PCA
1	120	118	72
2	119	122	74
3	134	119	71
4	165	120	69
5	10	117	73
6	123	8	70
7	145	7	8
8	74	9	7
9	136	148	10
10	71	121	9
11	155	10	132
12	73	141	131
13	98	123	133

Table 46 - Continued

Rank	MRMR	Relieff	PCA
14	137	139	121
15	32	140	118
16	162	138	119
17	144	136	122
18	147	157	148
19	135	137	117
20	49	135	130
21	67	134	139
22	2	145	17
23	114	158	19
24	72	153	123
25	166	99	18
26	107	159	138
27	140	154	140
28	113	42	141
29	96	155	164
30	57	156	167
31	111	164	161
32	25	167	134
33	168	161	135
34	59	32	145
35	104	34	52
36	122	50	81
37	153	97	44
38	23	6	51
39	157	48	35
40	99	36	84
41	106	40	147
42	105	165	96

Table 46 - Continued

Rank	MRMR	Relieff	PCA
43	103	166	100
44	7	169	136
45	97	163	95
46	65	4	36
47	102	168	125
48	101	162	99
49	26	147	97
50	156	41	93
51	91	98	12
52	110	49	168
53	112	95	162
54	108	44	165
55	22	81	154
56	60	52	14
57	33	39	137
58	13	31	120
59	115	30	43
60	160	47	24

Table 47 - Cross-validation evaluation performance results of different machine learning models trained with the features ranked by the MRMR method to detect the NF direction

Classifier	ACC	SENS	SPEC	PREC	F1-score	MCC	Number of Features
LDA	53.47	53.26	76.73	53.17	53.14	29.98	4
QDA	57.69	57.92	78.98	58.97	57.51	37.46	9
DT	97.83	97.83	98.92	97.84	97.83	96.75	36
Ensemble	99.98	99.98	99.99	99.98	99.98	99.98	57
KNN Equal	99.26	99.26	99.63	99.26	99.26	98.89	7
KNN Inverse	99.26	99.26	99.63	99.26	99.26	98.89	7

Table 47 - Continued

Classifier	ACC	SENS	SPEC	PREC	F1-score	MCC	Number of Features
KNN Squared Inverse	99.26	99.26	99.63	99.26	99.26	98.89	7
SVM Gaussian	86.02	85.98	93.01	86.01	85.98	79.00	53
SVM Linear	59.78	59.91	79.94	59.96	59.78	39.87	39
SVM Polynomial	97.36	97.36	98.68	97.35	97.35	96.05	47

Table 48 - Cross-validation evaluation performance results of different machine learning models trained with the features ranked by the Relieff method to detect the NF direction

Classifier	ACC	SENS	SPEC	PREC	F1-score	MCC	Number of Features
LDA	53.89	53.66	76.91	53.75	53.57	30.63	27
QDA	57.74	57.57	78.79	57.85	57.61	36.52	29
DT	99.17	99.16	99.58	99.17	99.16	98.75	21
Ensemble	100	100	100	100	100	100	24
KNN Equal	99.90	99.90	99.95	99.90	99.90	99.85	52
KNN Inverse	99.90	99.90	99.95	99.90	99.90	99.85	52
KNN Squared Inverse	99.90	99.90	99.95	99.90	99.90	99.85	52
SVM Gaussian	99.18	99.17	99.59	99.18	99.18	98.76	60
SVM Linear	64.80	64.84	82.40	64.80	64.80	47.23	56
SVM Polynomial	99.91	99.91	99.96	99.91	99.91	99.87	36

Table 49 - Cross-validation evaluation performance results of different machine learning models trained with the features ranked by the PCA method to detect the NF direction

Classifier	ACC	SENS	SPEC	PREC	F1 - score	MCC	Number of Features
LDA	51.86	52.03	76.02	52.30	51.84	28.17	58
QDA	54.08	54.16	77.05	54.07	53.9	31.21	58
DT	98.65	98.65	99.33	98.64	98.65	97.97	27
Ensemble	100	100	100	100	100	100	39

Table 49 - Continued

Classifier	ACC	SENS	SPEC	PREC	F1 - score	MCC	Number of Features
KNN Equal	89.90	89.88	94.95	89.88	89.88	84.83	59
KNN Inverse	89.90	89.88	94.95	89.88	89.88	84.83	59
KNN Squared Inverse	89.90	89.88	94.95	89.88	89.88	84.83	59
SVM Gaussian	92.77	92.74	96.35	92.77	92.70	89.13	59
SVM Linear	63.05	63.03	81.49	63.12	63.05	44.56	59
SVM Polynomial	97.05	97.04	98.52	97.05	97.05	95.57	58

Table 50 – Cross-validation evaluation performance result of the SVM Gaussian Kernel model with the hyperparameters optimized to detect the NF direction

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1-score	MCC	Number of Features
Relieff	SVM Gaussian	99.95	99.95	99.97	99.94	99.94	99.92	60

Appendix 3

This appendix presents the cross-validation results concerning chapter 6.

Table 51 - The sixty most relevant features ranked by MRMR, Relieff and PCA methods to detect an INF. Each number corresponds to the respective feature of Table 35

Rank	MRMR	Relieff	PCA
1	117	122	72
2	123	119	71
3	152	118	74
4	99	134	69
5	24	120	73
6	132	117	70
7	105	136	132
8	113	141	131
9	137	139	133

Table 51 - Continued

Rank	MRMR	Relieff	PCA
10	162	121	10
11	56	138	8
12	27	140	9
13	57	135	7
14	25	148	170
15	82	165	119
16	53	168	122
17	98	137	166
18	146	162	169
19	58	157	163
20	147	159	130
21	71	153	17
22	150	158	19
23	14	166	156
24	59	169	18
25	60	163	165
26	54	123	168
27	16	9	162
28	142	7	118
29	65	167	121
30	70	161	141
31	148	164	153
32	74	10	134
33	55	8	138
34	158	155	140
35	157	156	31
36	64	98	117
37	166	41	47
38	22	97	39

Table 51 - Continued

Rank	MRMR	Relieff	PCA
39	90	145	126
40	97	124	148
41	69	49	3
42	13	99	159
43	124	44	135
44	89	36	139
45	72	33	145
46	15	81	161
47	129	52	167
48	88	12	157
49	23	152	164
50	34	40	75
51	95	34	14
52	115	154	44
53	149	5	81
54	83	95	52
55	136	32	12
56	21	94	97
57	85	4	84
58	73	48	154
59	41	42	36
60	66	114	123

Table 52 – Cross-validation evaluation performance results of different machine learning models trained with the features ranked by the MRMR method to detect an INF

Classifiers	ACC	SENS	SPEC	PREC	F1 - score	MCC	Number of Features
DT	94.23	81.48	96.73	82.99	82.23	78.79	57
Ensemble	96.22	77.96	99.80	98.68	87.11	85.74	10

Table 52 - Continued

Classifier	ACC	SENS	SPEC	PREC	F1 - score	MCC	Number of Features
KNN Equal	88.95	65.59	93.53	66.48	66.03	59.44	3
KNN Inverse	88.95	65.59	93.53	66.48	66.03	59.44	3
KNN Squared Inverse	88.95	65.59	93.53	66.48	66.03	59.44	3

Table 53 - Cross-validation evaluation performance results of different machine learning models trained with the features ranked by the Relieff method to detect an INF

Classifier	ACC	SENS	SPEC	PREC	F1 - score	MCC	Number of Features
DT	99.65	98.80	99.82	99.06	98.93	98.72	11
Ensemble	99.91	99.68	99.95	99.77	99.73	99.67	40
KNN Equal	99.91	99.72	99.95	99.72	99.72	99.66	9
KNN Inverse	99.91	99.72	99.95	99.72	99.72	99.66	9
KNN Squared Inverse	99.91	99.72	99.95	99.72	99.72	99.66	9

Table 54 - Cross-validation evaluation performance results of different machine learning models trained with the features ranked by the PCA method to detect an INF

Classifier	ACC	SENS	SPEC	PREC	F1 - score	MCC	Number of Features
DT	99.46	98.18	99.71	98.54	98.36	98.04	44
Ensemble	99.91	99.60	99.97	99.82	99.71	99.65	38
KNN Equal	87.69	61.57	92.80	62.61	62.09	54.74	60
KNN Inverse	87.69	61.57	92.80	62.61	62.09	54.74	60
KNN Squared Inverse	87.69	61.57	92.80	62.61	62.09	54.74	60

Table 55 – Cross-validation evaluation performance result of the Ensemble model with data oversampling to detect an INF

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1 - score	MCC	Number of Features
Relieff	Ensemble	99.95	99.97	99.94	99.84	99.90	99.86	47

Table 56 – Cross-validation evaluation performance result of the Ensemble model with z-score normalization to detect an INF

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1 - score	MCC	Number of Features
Relieff	Ensemble	99.91	99.71	99.94	99.71	99.71	99.65	20

Appendix 4

This appendix presents the cross-validation results concerning chapter 7.

Table 57 - The sixty most relevant features ranked by MRMR, Relieff and PCA methods to detect human gait events. Each number corresponds to the respective feature of Table 35

Rank	MRMR	Relieff	PCA
1	48	158	74
2	2	155	71
3	154	159	72
4	155	153	73
5	40	120	69
6	158	156	70
7	46	40	132
8	4	48	131
9	76	32	133
10	75	4	21
11	32	157	24
12	127	152	122
13	159	118	118
14	50	136	119

Table 57 - Continued

Rank	MRMR	Relieff	PCA
15	39	166	166
16	152	163	169
17	29	169	163
18	37	34	121
19	69	50	168
20	30	162	162
21	136	165	165
22	153	168	156
23	45	42	130
24	72	6	20
25	3	122	19
26	34	134	17
27	41	38	18
28	128	46	141
29	80	119	117
30	71	145	153
31	6	30	148
32	120	127	97
33	101	117	138
34	123	154	52
35	38	2	81
36	165	121	36
37	73	140	140
38	1	135	139
39	107	41	31
40	43	139	44
41	87	49	47
42	77	137	39
43	104	148	134

Table 57 - Continued

Rank	MRMR	Relieff	PCA
44	111	123	126
45	49	99	40
46	74	33	48
47	99	161	12
48	93	167	155
49	106	5	16
50	115	76	3
51	78	39	4
52	143	75	32
53	63	31	158
54	42	3	157
55	102	47	159
56	114	94	14
57	166	164	161
58	162	126	167
59	61	97	145
60	151	98	23

Table 58 – Cross-validation evaluation performance results of different machine learning models trained with the features ranked by the MRMR method to detect human gait events

Classifier	ACC	SENS	SPEC	PREC	F1 - score	MCC	Number of Features
LDA	75.76	69.13	91.62	69.65	69.36	61.04	32
QDA	74.24	63.83	90.75	67.53	64.96	56.57	16
DT	91.27	89.33	97.00	89.16	89.24	86.23	36
Ensemble	97.47	96.68	99.11	97.12	96.90	96.03	48
KNN Equal	98.15	97.49	99.35	97.79	97.64	97.01	18
KNN Inverse	98.15	97.49	99.35	97.79	97.64	97.01	18
KNN Squared Inverse	98.15	97.49	99.35	97.79	97.64	97.01	18

Table 59 – Cross-validation evaluation performance results of different machine learning models trained with the features ranked by the Relieff method to detect human gait events

Classifier	ACC	SENS	SPEC	PREC	F1 - score	MCC	Number of Features
LDA	74.82	69.12	91.52	68.26	68.64	60.11	54
QDA	74.84	67.24	91.23	68.36	67.74	59.15	54
DT	93.76	92.20	97.85	92.30	92.25	90.10	30
Ensemble	98.10	97.41	99.33	97.82	97.61	96.96	52
KNN Equal	98.18	97.40	99.35	97.94	97.67	97.05	50
KNN Inverse	98.18	97.40	99.35	97.94	97.67	97.05	50
KNN Squared Inverse	98.18	97.40	99.35	97.94	97.67	97.05	50

Table 60 – Cross-validation evaluation performance results of different machine learning models trained with the features ranked by the PCA method to detect human gait events

Classifier	ACC	SENS	SPEC	PREC	F1 - score	MCC	Number of Features
LDA	68.76	61.07	89.54	60.62	60.73	50.25	60
QDA	69.03	62.26	89.73	61.45	61.65	51.37	60
DT	92.98	91.21	97.58	91.32	91.26	88.85	59
Ensemble	97.73	96.72	99.18	97.64	97.17	96.39	59
KNN Equal	80.15	77.25	93.10	76.58	76.90	69.97	59
KNN Inverse	80.15	77.25	93.10	76.58	76.90	69.97	59
KNN Squared Inverse	80.15	77.25	93.10	76.58	76.90	69.97	59

Table 61 – Cross-validation evaluation performance results of the Ensemble model with 18, 25 and 30 features to detect human gait events

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1-score	MCC	Number of Features
Relieff	Ensemble	97.44	96.50	99.09	97.09	96.79	95.91	18
		97.62	96.76	99.16	97.26	97.01	96.19	25
		97.68	96.84	99.18	97.35	97.09	96.29	30

Table 62 – Cross-validation evaluation performance result of the Ensemble model with the hyperparameters optimized to detect human gait events

Feature Selection Method	Classifier	ACC	SENS	SPEC	PREC	F1-score	MCC	Number of Features
Relieff	Ensemble	98.31	97.81	99.41	97.91	97.56	97.28	52