Methodology for detecting movements of interest in elderly people

Luz Angela Sucerquia Vega

Universidad de Antioquia UDEA
Facultad de Ingeniería
Medellín, Colombia
2016

Methodology for detecting movements of interest in elderly people

Luz Angela Sucerquia Vega

In Partial Fulfillment of the
Requirements for the Degree of
Master in Engineering

Advisor:
Prof. Jesús Francisco Vargas Bonilla, PhD.

Field:
Electronics
Research Group:
Sistemas Embebidos e Inteligencia Computacional –SISTEMIC–

Universidad de Antioquia
Facultad de Ingeniería
Medellín, Colombia
2016

Acknowledgments

There are many people I would like to thank; people who helped me make possible developing this work. I would like to thank to my advisor, Prof. Jesus Francisco Vargas Bonilla, for his support, guidance and dedication during the development of this work.

I would like to thank Monica Rodriguez for her collaboration and support in the long and arduous data acquisition process, Camilo Ocampo for developing the device, and Darwin Agudelo for programing the algorithm. I also would like to thank to all those anonymous participants, PROSA UDEA, the Judo and Aikido martial arts groups at the UDEA for their insightful contribution in the generation of the SisFall dataset, especially to Felipe Toro for his help acting for the videos.

I thank my family for their invaluable support during all these years; especially to my mother, she always kept up the hope on difficult times. Additionally, I want to thank all my friends and partners for making this process less difficult and more pleasant. Finally, I would like to thank my boyfriend for his unconditional and constant encouragement and support, without him, I would never have done this.

Angela Sucerquia

This project was partially supported by a grant from Regalías de la nación: "Plataforma tecnológica para los servicios de teleasistencia, emergencias médicas, seguimiento y monitoreo permanente a los pacientes y apoyo a los programas de promoción y prevención", contract number RutaN512C-2013.

Resumen

El aumento en la expectativa de vida, tanto en Colombia como a nivel mundial, requiere un mayor uso de tecnologías dentro del área de la salud que permita a los adultos mayores conservar su independencia y mejorar su calidad de vida. En esta tesis se analiza la problemática de caídas en adultos mayores independientes, cuyas consecuencias pueden minimizarse mediante un sistema portable de detección automática que envíe una alarma de forma oportuna. Como punto de partida se elaboró una base de datos con 38 participantes que realizaron 19 actividades de la vida diaria y simularon 15 tipos de caídas. Para ello se utilizó un dispositivo portable con un acelerómetro triaxial.

Pruebas preliminares con algoritmos de extracción de características comúnmente usados en la literatura para discriminar entre caídas y actividades de la vida diaria presentaron una precisión de hasta 96 %. Para ello se utilizó un clasificador de bajo costo computacional basado en umbral que pudiese funcionar en tiempo real en sistemas embebidos. Un análisis individual de actividades con cada uno de los algoritmos de extracción de características demostró que algunas de ellas son complementarias entre sí, este análisis se usó como punto de partida para desarrollar métricas no lineales que mejoraron la discriminación a un 99 %. También se observó que muchos de los falsos positivos son debidos a actividades periódicas de alta aceleración, que pudieron ser detectados a partir de su periodo.

Con el fin de garantizar que la metodología desarrollada fuese implementable en sistemas embebidos sin que ello signifique una alta carga computacional (y el consecuente consumo de batería), en este trabajo se propone un algoritmo basado en un filtro de Kalman, un pre procesamiento basado en un filtro Butterworth de cuarto orden, una métrica no lineal basada en dos características de extracción comúnmente usadas, y un clasificador basado en umbral. Este algoritmo fue implementado en un dispostivo embebido y validado mediante la simulación de las mismas actividades de la base de datos adquirida en este trabajo, además de una prueba piloto en condiciones reales con adultos mayores. Ambas pruebas presentaron una tasa de error inferior al 1 %.

Palabras clave:
Adultos mayores, detección de caídas, actividades de la vida diaria, acelerómetro, procesamiento de señales.

Abstract

The increase in life expectancy, both in Colombia and globally, requires higher use of health-care technology to allow elderly adults maintain their independence and improve their quality of life. In this thesis, we analyze the problem of falls in independent elderly people. The consequences of a fall can be minimized by a portable automatic detection system, wich sends an alarm right after an event. We started by creating a dataset with 38 participants that conducted 19 activities of daily life and simulated 15 types of falls. They used a portable device with a triaxial accelerometer.

Preliminary tests with feature extraction algorithms commonly used in the literature to discriminate between falls and activities of daily living presented up to 96 % of accuracy. They were implemented with a low computational cost threshold-based classifier, which can operate in real-time on embedded systems. An individual activity analysis with each feature extraction algorithm demonstrated that some of them are complementary to each other. This analysis was used as a starting point to develop nonlinear discrimination metrics that improved the accuracy to 99 %. We also noted that most false positives are due to high acceleration periodic activities, and we could detect them solely based on their period.

In order to guarantee that the developed methodology can be implemented on embedded systems without affecting their computational capability (and the consequent battery consumption), we propose an algorithm based on a Kalman filter, with a pre-processing stage based on a 4-th order Butterworth filter, a non-linear feature based in two commonly used feature extraction characteristics, and a threshold-based classifier. This algorithm was implemented in an embedded device and validated by simulating the same activities of the dataset acquired in this work, along with a pilot test in real conditions with elderly adults. Both tests presented an error rate below 1 %.

Keywords:
Elderly people, fall detection, activities of daily living, accelerometer, signal processing.

Contents

1 Introduction

1.1 Problem description

The number of elderly people living alone has been continuously growing worldwide. This independence comes with the risk of not receiving prompt attention if an accident occurs. A third of the people over 65 years old suffer in average one fall per year [1], and it grows with age [2] and previous falls, where about one third develop fear of falling again [3, 4].

Not receiving attention in the first hour of the accident increases the risk of dead and chronic affections [5]. This issue has been widely addressed in recent years with systems that detect falls in elderly people, and generate a prompt alert that can reduce the consequences related to medical attention response time [6]. These systems have acceptance among the objective population as a way to support their independence and reduce their fear of falling [7].

Falls are commonly detected with wearable or ambient-based systems (see [6, 8–10] for reviews in the field). Ambient-based sensors such as cameras are intrusive and do not solve the problem for independent adults, who are not confined to closed spaces [11–13]. According to [2] up to 50 % of the falls in independent elderly people occur outside the home premises. Wearable devices offer portability as they can be used regardless of the user location. Available wearable devices include smartphone apps and self-developed systems. In both cases the preferred sensor is the triaxial accelerometer because of its low cost, small size, and because it is built-in in almost all smartphones [6].

Independently of the device used, other authors have faced problems with energy consumption, battery life, false positives (the alarm turns on with activities of daily living), false negatives (the alarm does not turn on with falls), and user comfort. Solving these problems is still an open issue due to the long-term real-time operation needed, and the low computational capabilities that these systems have. In this work, we propose to develop a fall detection methodology based on movements of interest, we will test it with a self-developed dataset, and validate it with elderly people using a wearable device.

1.2 Objectives

1.2.1 General

To develop a methodology to identify movement focused on detecting falls in elderly people, and to implement it in a wearable system able to generate the corresponding alarm.

1.2.2 Specific

- To report the analysis of algorithms that identify movements of interest on elderly people with acceleration data.
- To develop a fall detection methodology to generate alerts using acceleration signals and movement identification.
- To design a test protocol and generate a dataset with movements of interests and falls on young and elderly people.
- To implement the developed methodology in a wearable device with transmission, and validate it in an elderly people population.

1.3 Literature overview

Strategies to opportunely detect falls in elderly people have spread in recent years [6]. This phenomena corresponds to the known consequences associated to a large waiting time to receive medical attention after falling [6, 14]. Our analysis of these strategies begins listing the different wearable devices and sensors used to detect falls and movement. Then, we explore the wide variety of fall detection approaches available in the literature. We finish this analysis with a review of walk and jog detection algorithms, as an input to the main achievement of this thesis work: detecting specific daily living activities to reduce fall detection errors, which is something that other authors have not addressed before.

As stated in the problem description, ambient based devices are not feasible for independent elderly people given that up to 50 % of falls may occur in outside premises [2]. Consequently, we will focus exclusively on wearable devices.

1.3.1 Sensors used for fall detection

The most popular sensors used for wearable devices are accelerometer and gyroscope; sometimes used together with pressure sensors or magnetometers. However, most works found in the fall and motion detection literature are solely based on capacitive accelerometers because of their small size, low weight, low energy consumption, and low cost [15]. These

characteristics have popularized accelerometers in several electronic devices for a wide range of uses (stabilizing cameras, video-game pads, smartphone positioning, etc.), and recently in smart-watches or directly attached to clothes.

Specifically for fall detection, accelerometer based systems are highly popular. From 197 studies with portable devices reviewed in [6], 186 used them. Some accelerometer devices include a gyroscope for detecting the position of the subject. However, as mentioned in Bushching et al. [16], the power consumption in medical devices is critical; and as referred in Igual et al. [6], the inclusion of a gyroscope does not significantly improves the accuracy of the system.

Another commonly tested possibility for detecting falls is the use of several accelerometers distributed through the body. Cleland et al. [15] stated that a single accelerometer is enough for these tasks. However, Allen et al. [17] previously stated otherwise, and Gao et al. [18] also stated that more sensors improve the energy efficiency of the device, due to the lower complexity of the algorithms. Later, Gao et al. [19] analyzed multiple vs. single sensor systems for detecting activities, and they demonstrated that elaborated classifiers and feature sets are not required to obtain high accuracies on a multi-sensor system.

Unfortunately, there are not standardized methodologies for determining the best fitted number and location of the sensors for both fall detection and motion capture, and few works focus on it. In [15] the authors analyzed different locations for placing a triaxial accelerometer for fall and motion detection. The authors concluded that most of the commonly used positions in the body (waist, chest, leg, etc.) provide similar information, and they highlighted the waist as preferred for being in the center of mass of the body. But as an example of the widespread of works in this field, Yuan et al. [20] proposed the wrist as the best placement for detecting falls, despite the higher false positive rates they acknowledge this position implies. For a wider overview, the review presented in Pannurat et al. [9] resumes the most common locations of the sensors, reviewing 12 works using the chest, 8 the head, 29 the waist, 13 the thigh, 6 the wrist, 3 the back, and 5 the ankle or foot. In conclusion, the waist is the best single location as the chest and head (that provide similar information) are more uncomfortable for a device.

1.3.2 Wearable devices

Wearable devices used for fall detection can be divided in three categories: smartphones, self-developed ones, and commercial systems. Smartphones are a popular selection for implementing fall detection algorithms because they include a robust hardware, a powerful processor, and they are economically affordable [6, 10, 21, 22]. However, the low cost of the individual components and design tools has encouraged authors to develop their own em-

bedded devices. This alternative has opened possibilities of devices with more than one accelerometer, and other sensors such as gyroscopes, magnetometers, and barometric pressure sensors, among others [19, 23–25]. There also exist commercial options that can be purchased in the market [26–29]. But they are usually exclusive of private healthcare companies [9, Table 6] preventing their use in research.

Authors face similar problems independently on the selected device. The main issues authors report are: false alarm percentages, undetected falls, energy consumption (mainly affected by the computational cost of the classification algorithms), and user comfort [6]. These issues are more complex for smartphone based systems because they are multipurpose (more energy consumption), and their free movement within the pocket highly increases the noise (more classification errors) [10]. Igual et al. [30] performed a comparative analysis of different widely used algorithms with three public databases acquired with smartphones. The authors demonstrated the low robustness of detecting falls due to the large variability among datasets, i.e., the way people carry the smartphone.

For this project, we require a device permanently attached to the body of independent elderly people. As presented above, smartphones are prone to be forgotten, to fall from the hand of the user, and they have limited battery life. Then, in this work we developed an wearable embedded device to record data and test our methodology. It is presented in Chapter 2.

1.3.3 Datasets

With the proper wearable device technology selected, candidate fall detection algorithms must be evaluated before being implemented. It requires to acquire datasets with common types of falls and activities of daily living –ADL–. In this sense, some authors analyzed how elderly people fall. Back in 1993, authors in [5] performed a wide survey with 704 women over 65 years old. They found that the most common causes reported were trips, slips and lost of balance. About the conditions of the fall, in [31] the authors found that women were three times more likely to hit the ground in the hips than men, and that most people fell in forward direction with 60 % of prevalence. Most activities currently selected for testing algorithms are based on results of these studies.

Once the selected ADL and falls are simulated and acquired, the raw acceleration data must be processed and classified. Authors commonly filter the data, apply a feature extraction method, and classify activities as falls or ADL. The literature provides a wide number of features (Pannurat et al. [9, Table 4] has a complete list). Unfortunately, the absence of public datasets has prevented fair comparisons among them, making it impossible to replicate results [9, 32]. Additionally, most works have not been tested with the objective population (elderly people, see [9, Table 1]). In [32] for example, authors tested 13 state-of-the-art

approaches with real elderly people falls, and found that indeed their performance is affected by the training datasets used. However, they did not release the validation dataset preventing other authors to analyze why those features reduced their performance, and more important how to solve it. To our knowledge, there only exist three public datasets, all acquired using smartphones [25,33,34]. In Chapter 2, we present a new dataset with falls and ADL acquired with a wearable device.

1.3.4 Fall detection algorithms

Most fall detection algorithms follow a three stages methodology: preprocessing, feature extraction, and classification. Preprocessing is critical in the performance of the classification algorithms and their computational burden, but most authors just mention it and most works just include a simple low-pass filter followed by a baseline removal. In Chapter 3 we analyzed the effect of this kind of filtering and we conclude that it is effectively enough to this purpose.

On the other hand, there is a wide amount of feature extraction algorithms proposed in the literature [6,9]. These algorithms process the acceleration signal in order to obtain discriminative features among falls and ADL, such as acceleration peaks, variance, angles, etc. (see [9, Table 4] for a complete list). These features are initially divided between static and dynamic. Static algorithms compute point-wise acceleration, and dynamic ones commonly use a sliding window. Among dynamic features it is usual to find statistic metrics (mean, variance, etc.), changes in orientation (sometimes using time, as falls change faster than common activities), and area under the curve (computed with integrals). In Chapter 3, we provide results of some of the most commonly used detection features tested with both young and elderly people.

Regarding classification (between falls and ADL), threshold based detection is still the most opted choice over other alternatives such as nearest neighbors (NN) and support vector machines (SVM), mainly because the latter ones are not practical for real-time implementation. Habib et al. [10] show various examples of SVM approaches consuming the battery in a few hours. In recent years, authors have focused on tackling the issues of threshold based approaches. A simple but clever solution to avoid false positives was proposed by Koshmak et al. [22], allowing the user to cancel the alarm signal in case of false alarm. However, it does not solve the main problem of these algorithms: the high number of false positives and negatives. Because of that reason authors insist in using SVM. Some of the many examples are the proposal of Kau. et al. [35] for solving the movement-in-the-pocket problem of smartphones, and the detection of falls as novelties of Medrano et al. [34]. But none of them addresses the battery issues.

Vavoulas et al. [33] published and tested a dataset (called Mobifall) acquired with a smart-

phone and achieved good detection and classification results, but again with SVM. More recently, Igual et al. [30] tested that dataset and other two public ones with NN and SVM strategies. Despite the fact that their algorithms are too complex for implementation in embedded devices (in terms of battery life), their conclusion that algorithms are highly dependent on the specific dataset used for training is an important issue not addressed by other authors. In Chapter 4, we achieved 99 % of accuracy with a threshold based algorithm implemented over Mobifall dataset. We tried to test the other two datasets tested in [30] but we found another issue, their files are too short and they do not provide enough information about the specific activities.

1.3.5 Identification of specific activities

The individual activity analysis performed in Chapter 3 shows us that most failures in fall detection are focused on few activities. Most of these activities coincide in periodic waveforms and high peak accelerations. But motion capture with accelerometers has not previously been analyzed with fall detection purposes. The closest approach comes from Cola et al. [36], who detected gait deviation as fall risk feature. However, there are previous approaches in the literature for detecting jog and walk with accelerometers. Oner et al. [37] used the peaks of the acceleration signal measured with a smartphone to find steps, and subsequently the kind of activity based on the period between steps. Wundersitz et al. [38] also used the acceleration peaks but using their own device. Other authors used more elaborated metrics but all peak based. Clements et al. [39] computed principal components of the Fast Fourier Transform (FFT) to cite an example.

One alternative to detect non-peak based measures is using wavelets. In Godfrey et al. [40], the authors used wavelets for classifying activities and postural transitions in young and elderly people. The wavelet was initially used to eliminate low frequency drift, and then to determine with vertical acceleration the type of postural transition. Yazar et al. [41] proposed a wavelet based algorithm called single-tree complex wavelet to detect falls versus ordinary activities, with a feature vector composed by wavelet energy. Both wavelet algorithms presented good performance metrics (up to 99 %), but complexity issues were not addressed. In this sense, authors have used Discrete Wavelet Transform (DWT) in portable devices. For example, in [42] and [43] the authors implemented 1D and 2D DWT for image processing and compression in smartphones and embedded systems. Being the image analysis more computationally intensive than the 1D acceleration data used here.

1.3.6 Kalman-filter-based detection

Finally, in Chapter 6 we propose a fall detection algorithm based on Kalman filtering, which simultaneously detects orientation changes and periodic activities. The Kalman filter is a

well-known quadratic optimal estimator [44], widely used in several research fields. The Kalman filter is Markovian (avoiding large memory storage) and linear (simple computations for lower energy consumption). Both characteristics are desirable for implementation in embedded devices.

The Kalman filter has been previously used to identify movements of interest with accelerometers. [45] used a Kalman filter to determine the lie-to-sit-to-stand-to-walk states, which are commonly used to measure the risk of falling in elderly people (with the Berg Balance Scale BBS for example [46]). In that work, the authors used an Extended Kalman filter to determine the orientation of the device. [47] proposed a novel user context recognition using a smartphone. There, the Kalman filter was used to obtain the orientation of the device based on its multiple sensors (not only the accelerometer). But the authors did not specify how they did it. Finally, [48] used a multiple sensor system to determine gait initiation and termination. In their work, the Kalman filter was used again to obtain the orientation of the devices.

The aforementioned works coincide in the objective of the Kalman filter (identifying locomotion activities) but they differ on the way they implement it, and none of them is interested in detecting falls. In [49], the authors again used the Kalman filter to obtain the device angle, but with the purpose of detecting falls with three sensors (including gyroscope). All previously mentioned works demonstrate that the orientation of the device computed with a Kalman filter is a strong feature extraction characteristic, and that the Kalman filter is useful to detect periodic activities such as walking or jogging. However, none of them combine these capabilities as we propose in Chapter 6.

1.4 Publications

As result of this work, five publications were developed. One of them was already published, two are under review, and the final two are in preparation. A copy of all papers and white papers is included as Supplementary Material.

- J.D. López, A. Sucerquia, L. Duque-Muñoz, and F. Vargas, "Walk and jog characterization using a triaxial accelerometer", in IEEE International Conference on Ubiquitous Computing and Communications, 2015, pp. 1406–1410.
- A. Sucerquia, J.D. López, and F. Vargas,"Two-threshold energy based fall detection using a triaxial accelerometer", in 38th Annual International Conference of the IEEE-EMBS, 2016.
- J.D. López, A. Sucerquia, F. Vargas, "Analyzing Multiple Accelerometer Configurations to Detect Falls and Motion", CLAIB 2016 (Accepted for publication).

- A. Sucerquia, J.D. López, and F. Vargas,"SisFall: A Fall and Movement Dataset", 2016 (Submitted to Sensors).
- A. Sucerquia, J.D. López, and F. Vargas, "Kalman filter based elderly fall detection with a triaxial accelerometer", (In preparation).

2 Datasets

2.1 Introduction

Research on fall and movement detection with wearable devices has witnessed promising growth. However, there are few publicly available datasets, all recorded with smartphones, that prevent authors to evenly compare their new proposals. Additionally, most works are not tested with the objective population (elderly people), reducing their accuracy in real-life applications [32]. In order to tackle this issue, the movement and fall detection approaches proposed in this thesis were tested with two datasets, the publicly available Mobifall dataset [50], and a novel dataset (called Sistemic Research Group Fall and movement dataset –SisFall–) specifically developed for this work. SisFall dataset is available for download at http://sistemic.udea.edu.co/wp-content/uploads/2016/03/SisFall_dataset.zip.

2.2 SisFall

In this section, we present the Sistemic Research Group Fall and movement dataset –SisFall, a dataset of falls and activities of daily living –ADL– acquired with a self-developed device composed of two types of accelerometer and one gyroscope. It consists of 19 ADL and 15 fall types performed by 23 young adults, 15 ADL types performed by 14 participants over 60 years old, and data from one additional participant of 60 years old that performed ADL and falls.

2.2.1 Selection of activities

Prior to define the activities of the dataset, we analyzed those falls and ADL commonly tested in the literature (see [9, Table 4]). Then, we completed the information by performing a survey with elderly people living alone and administrative personnel from retirement homes. The survey consisted in three main questions: For each fall incident, (*i*) which activity were you performing when the fall happened? (*ii*) What produced the fall? a sliding, a faint, a trip, other? (*iii*) In which orientation the fall happened? what part of the body received the impact?. The survey was conducted with 15 elderly people from the psycho-physic program of the Universidad de Antioquia (between July and August 2014, in Medellín, Colombia), and 17 retirement homes (between October 2014 and January 2015, in

Medellín and Manizales, Colombia).

As a result of the survey, the independent elderly people fall more when walking, taking a shower, and walking up or down stairs; and fall less when trying to get-up or sit-down in a chair or a bed, or bending. On the other hand, elderly people living in retirement homes fall more when walking and when trying to get-up from a chair or a bed; and fall less when walking up or down stairs. The answers given by the participants were consistent with the results presented by [5]. Table **2-1** shows the types of fall we selected for this work.

Table **2-1**: Types of fall selected to this work.

Code	Activity	Trials	Duration
F01	Fall forward while walking caused by a slip	5	15s
F02	Fall backward while walking caused by a slip	5	15s
F03	Lateral fall while walking caused by a slip	5	15s
F04	Fall forward while walking caused by a trip	5	15s
F05	Fall forward while jogging caused by a trip	5	15s
F06	Vertical fall while walking caused by fainting	5	15s
F07	Fall while walking, with use of hands in a table to dampen fall, caused by fainting	5	15s
F08	Fall forward when trying to get up	5	15s
F09	Lateral fall when trying to get up	5	15s
F10	Fall forward when trying to sit down	5	15s
F11	Fall backward when trying to sit down	5	15s
F12	Lateral fall when trying to sit down	5	15s
F13	Fall forward while sitting, caused by fainting or falling asleep	5	15s
F14	Fall backward while sitting, caused by fainting or falling asleep	5	15s
F15	Lateral fall while sitting, caused by fainting or falling asleep	5	15s

ADL of Table **2-2** were selected based on: common activities, activities that are similar (in acceleration waveform) to falls, and activities with high acceleration that can generate false positives. All ADL and falls selected for this work were approved by a physician specialized in sports. According to the survey, falling when walking up or down stairs is a common type of fall, but we did not include it in our protocol because of the high risk of having an accident.

Table **2-2**: Types of ADL selected to this work.

Code	Activity	Trials	Duration
D01	Walking slowly	1	100s
D02	Walking quickly	1	100s
D03	Jogging slowly	1	100s
D04	Jogging quickly	1	100s
D05	Walking upstairs and downstairs slowly	5	25s
D06	Walking upstairs and downstairs quickly	5	25s
D07	Slowly sit in a half height chair, wait a moment, and up slowly	5	12s
D08	Quickly sit in a half height chair, wait a moment, and up quickly	5	12s
D09	Slowly sit in a low height chair, wait a moment, and up slowly	5	12s
D10	Quickly sit in a low height chair, wait a moment, and up quickly	5	12s
D11	Sitting a moment, trying to get up, and collapse into a chair	5	12s
D12	Sitting a moment, lying slowly, wait a moment, and sit again	5	12s
D13	Sitting a moment, lying quickly, wait a moment, and sit again	5	12s
D14	Being on one's back change to lateral position, wait a moment, and change to one's back	5	12s
D15	Standing, slowly bending at knees, and getting up	5	12s
D16	Standing, slowly bending without bending knees, and getting up	5	12s
D17	Standing, get into a car, remain seated and get out of the car	5	25s
D18	Stumble while walking	5	12s
D19	Gently jump without falling (trying to reach a high object)	5	12s

Fig. 2.2.1 includes a video of a participant jogging, tripping and falling (activity F05). It is clear how the amplitude of the accelerometer is highly sensitive to each state, and how the amplitude raises during the fall. This video has low resolution for technical issues, but high resolution videos of each type of fall and ADL performed by the participants are available at http://sistemic.udea.edu.co/wp-content/uploads/2016/03/SisFall_videos.zip. They were recorded as an effort to solve another drawback in the literature: showing the exact conditions of the recordings.

Figure **2-1**: Video example of activity F05 (jog-trip-fall). High definition videos of all activities are available at http://sistemic.udea.edu.co/wp-content/uploads/2016/03/SisFall_videos.zip

2.2.2 Participants

This database was generated with collaboration of 38 volunteers divided in two groups: Elderly people and young adults. Elderly people group was formed by 15 participants (8 male and 7 female), and young adults group was formed by 23 participants (11 male and 12 female). Table **2-3** shows age, weight, and height of each group. Individual information is available in the Readme of the dataset.

Table **2-3**: Age, height and weight of the participants.

	Sex	Age	Height [m]	Weight [kg]
Elderly	Female	62 – 75	1.50 – 1.69	50 – 72
	Male	60 – 71	1.63 – 1.71	56 – 102
Adult	Female	19 – 30	1.49 – 1.69	42 – 63
	Male	19 – 30	1.65 – 1.83	58 – 81

Young adults performed ADL and falls while elderly people performed only ADL, except the participant of 60 years old identified by code SE06, who is an expert in Judo and also simulated falls. Elderly people did not perform activities D06, D13, D18, and D19 from Table **2-2** due to recommendations of the physician specialized in sports. Additionally, some elderly people did not perform some activities due to personal impairments (or medical recommendation).

2.2.3 Experimental set-up

The dataset was recorded with a self-developed embedded device composed of a Kinets MKL25Z128VLK4 microcontroller, an Analog Devices ADXL345 accelerometer (configured

for ±16 g, 13 bits of ADC), a Freescale MMA8451Q accelerometer (±8 g, 14 bits of ADC), an ITG3200 gyroscope (±2000 °/s, 16 bits of ADC), a SD card for recording, and a 1000 mA/h battery. The device was fixed to the waist of the participants (Fig. **2-2**). This location provides high distinction among activities for a single accelerometer system.

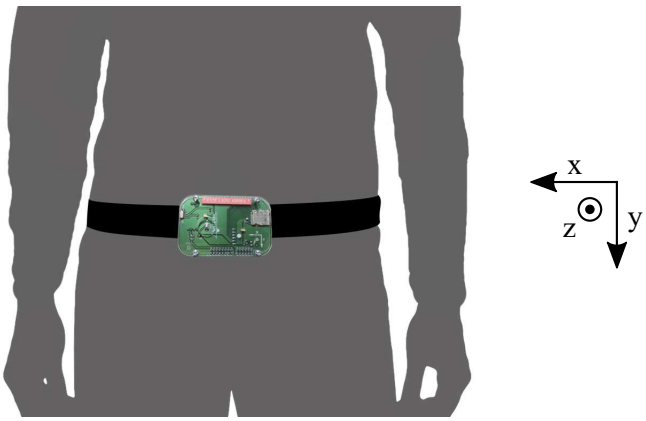


Figure **2-2**: **Device used for acquisition.** The self-developed embedded device included two accelerometers and one gyroscope. It was fixed to the waist of the participants.

Only acceleration data acquired with the ADXL345 sensor was used in this work, as it is the most energy efficient of the three sensors used. However, the data recorded with the other accelerometer and the gyroscope are also publicly available for further studies. The orientation of the sensor (see Fig. **2-2**) presents the positive z axis in forward direction, the positive y axis in the gravity direction, and the positive x axis pointing to the right side of the participant. All activities were acquired with a frequency sample of 200 Hz.

The classrooms and open spaces of a coliseum at the Universidad de Antioquia were used for recording the activities. In order to guarantee health safety conditions, falls were simulated using safety landing mats. Activity D17 from Table **2-2** was recorded using the copilot chair of a Renault Logan car. The time required for recording all trials was approx. 1.5 hours for each elderly person and 3.5 hours for each young adult.

2.3 MobiFall

Some of the algorithms proposed in this thesis were also validated with a third-party public dataset. Mobifall dataset was acquired by Vavoulas et al. [50] and is publicly available in the website http://www.bmi.teicrete.gr/index.php/research/mobifall. The second version of MobiFall dataset was formed by 24 participants (17 male with ages between 22–47, and 7 female with ages between 22–36), 9 of these subjects simulated falls and ADL,

and 15 subjects simulated only falls. The dataset consists of 9 ADL and 4 fall types (Table **2-4**). The signals were acquired with a Samsung Galaxy S3 cellphone, inertial module LSM330DLC. The device was located in a trousers pocket with random direction.

Table **2-4**: Types of ADL and Falls included in the MobiFall dataset

Code	Activity	Trials	Duration
STD	Standing with subtle movements	1	5m
WAL	Normal walking	1	5m
JOG	Jogging	3	30s
JUM	Continuous jumping	3	30s
STU	Stairs up (10 stairs)	6	10s
STN	Stairs down (10 stairs)	6	10s
SCH	Sitting on a chair	6	6s
CSI	Step in a car	6	6s
CSO	Step out a car	6	6s
FOL	Fall Forward from standing, use of hands to dampen fall	3	10s
FKL	Fall forward from standing, first impact on knees	3	10s
BSC	Fall backward while trying to sit on a chair	3	10s
SDL	Fall sidewards from standing, bending legs	3	10s

This dataset includes acceleration, orientation and rotation data, but only acceleration was used in this work. Acceleration signals were acquired with a sampling rate around 87 Hz. File `SDL_acc_16_01.txt` was removed from the dataset (the recording begins at the middle of the fall).

2.4 Summary

Research on elderly fall detection lacks of public datasets with enough activities and falls simulated by elderly people. Available datasets have few activities and none includes falls performed by elderly people. In this chapter we presented and made publicly available the SisFall dataset. It consisted of up to 34 activities (falls and ADL) performed by 38 participants, acquired with a self-developed wearable device.

This dataset includes an elderly person (subject SE06) that performed ADL and falls. Subject SE06 is to our knowledge the only publicly available dataset of an elderly person simulating falls. Together with a wide variety of activities and subjects, the SisFall dataset will help the scientific community as a complete benchmark in future years. As supplementary

material, we included videos of all simulated activities as an effort to help other researchers to replicate this work.

3 Fall detection algorithms from the literature

3.1 Introduction

In this chapter, we test the SisFall dataset presented in the previous chapter with widely used features and a simple to implement threshold based classification. We achieved up to 96 % of accuracy in fall detection with young adults and 99 % in ADL with elderly people. An individual activity analysis demonstrates that most errors coincide in a few number of activities, suggesting authors to develop algorithms focused on these particular ones. Finally, validation tests with the elderly participant that simulated falls significantly reduced the performance of the features. This validates findings of other authors and encourages to develop new strategies with this new dataset as benchmark.

3.2 General fall detection scheme

Preprocessing stage: Preprocessing is critical in the performance of the classification algorithms and their computational burden. In this work, we performed a comparison between using preprocessing or not in fall detection. The preprocessing stage consisted of a 4-th order IIR Butterworth low-pass filter with cut-off frequency of 5 Hz. This filter was selected due to its low computational cost, as preliminary tests with more elaborated IIR and FIR filters did not improve the classification accuracy. Some features based on peak amplitude required bias (DC Level) rejection. This was performed using the first derivative between two consecutive samples.

Feature extraction: The objective of this stage is to maximize the separation between ADL and falls. We tested 28 features listed in [9, Table 4] (coded in the original paper as F1–F11, F16, F21–F29, F32–F35, F44–F46). From the 28 features, only six achieved over 90 % of accuracy on preliminary tests (not shown here). Within this group, we selected three of them for presenting in this work, they are listed in Table **3-1**. The sum vector magnitude on horizontal plane was selected because it was the best static feature, the standard deviation magnitude presented the higher accuracy among all features, and the activity signal magnitude was the best feature among the area under the curve approaches. Not all tested

features were initially intended for detecting falls, which may explain why only six achieved high accuracy.

Feature C_1 from Table **3-1** is static. The other two features require a sliding window selected for this paper of 0.5 s, being this the minimum value that includes the critical phase of the fall [51]. Here, acceleration signal of one sample in the three axis is defined as the vector $\vec{a} = [a_x, a_y, a_z]^T \in \Re^3$, the sliding window used for computing the dynamic features is denoted with $\tilde{a}[k] = [\vec{a}^T[k - N_v], \ldots, \vec{a}^T[k]]^T \in \Re^{N_v \times 3}$, at time step k, with $N_v = 100$ samples (0.5 s). The standard deviation operator was defined as $\text{std}(\cdot)$, and the integral in C_3 was computed with trapezoid method.

Table **3-1**: Features used to test the proposed dataset.

Code	Feature	Equation
C_1	Sum vector magnitude on horizontal plane	$C_1[k] = \sqrt{a_x^2[k] + a_z^2[k]}$
C_2	Standard deviation magnitude	$C_2[k] = \sqrt{\sigma_x^2[k] + \sigma_z^2[k]};\ \text{with } \sigma_i = \text{std}\big(\tilde{a}_i[k]\big)$
C_3	Activity signal magnitude area	$C_3[k] = \int_{k-N_v}^{k} \left(\sqrt{\tilde{a}_x^2[n] + \tilde{a}_y^2[n] + \tilde{a}_z^2[n]}\right) dn$

For the tests with preprocessing, the Butterworth filter was applied to the raw data before computing features. Additionally, C_1 and C_3 behaved better with bias removal (with derivative) implemented after the filtering stage.

Classification: A simple to implement threshold-based classifier was selected for this work. Threshold-based classification is still the most widely used strategy for fall detection, as it is less computationally intensive than support vector machines and similar classification algorithms [10].

For each feature, the threshold was obtained as follows:

1. A set of candidate thresholds was established between the minimum acceleration in a fall, and the maximum acceleration in ADL.

2. The sensitivity and specificity of training data for each candidate threshold value was obtained.

3. The threshold that guaranteed a minimum difference between sensitivity and specificity was chosen.

4. Validation data was tested with the chosen threshold. This procedure was performed in ten cross-validation rounds.

Sensitivity (SEN) and specificity (SPE) were calculated as specified in [52]:

$$\text{SEN} = \frac{TP}{TP + FN} \qquad \text{SPE} = \frac{TN}{TN + FP} \tag{3-1}$$

where TP are falls correctly classified, FN are falls that the algorithm did not detect, TN are ADL correctly classified, and FP indicates false falls. The accuracy (ACC) was calculated using Eq. (3-2), where N is the number of files of the dataset:

$$\text{ACC} = \frac{TN + TP}{N} \tag{3-2}$$

This balanced computation of the accuracy is selected due to the large difference between the number of ADL and fall files.

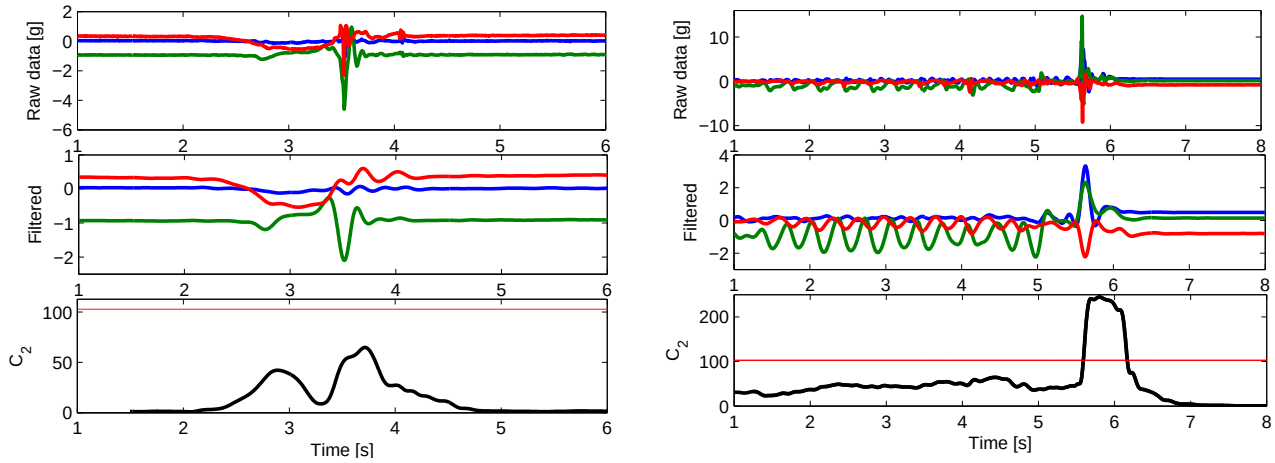
Fig. **3-1** shows an example of the preprocessing stage and the computation of feature C_2 for ADL D11 (trying to get-up from a chair and fail – figures on left) and fall F05 (trip and fall while jogging – figures on right). This ADL was selected because of its high peak acceleration. Despite this, C_2 peak was around 40 % below the threshold value (Fig. 3.1(a) –bottom left). On the other hand, feature C_2 far crossed the threshold during fall F05 (Fig. 3.1(b) –bottom right). Note that while jogging before the fall, which is a high acceleration activity, feature C_2 was always below the threshold.

Cross-validation: The robustness of the classification stage was analyzed with a cross-validation set-up. A first analysis was performed including only young adults guaranteeing the same proportion of falls and ADL in all groups (3542 files randomly divided into 10 groups). Ten independent cross-validation rounds were performed, each with 3195 files for training and 347 files for validation. Each group was used in one round as validation data. The elderly people group was not initially included in the cross-validation, in order to determine the effect of setting-up an algorithm intended for elderly people only with young adults.

A second cross-validation analysis was performed with all participants except SE06. In this case, we analyzed the performance of the algorithm when including ADL of elderly people. Participant identified by code SE06 was not included in the cross-validation because he is the only elderly that simulated falls. SE06 data were only used in a final validation as the closest to real-life conditions example.

3.3 Results and discussion

We initially tested the features of Table **3-1** on the young adults group data without performing preprocessing. Features C_2 and C_3 presented average accuracies over 90 % in 10-fold



(a) ADL: Trying to get-up from a chair and fail.

(b) Fall: Trip and fall while jogging.

Figure **3-1**: Example of processing and classification. The features are computed after the filtering process of the raw data. (a) ADL D11 gives C_2 values (bottom) below the threshold (horizontal red line). (b) C_2 feature of fall F05 crosses the threshold when the fall is detected.

cross-validation (Table **3-2**). Feature C_1 (which is one of the most used in the literature, mostly for comparison purposes) presented the lowest performance (88.8 %), but it required 1/100 of memory and simpler computations than C_2 and C_3 because it is static. This fact may be counterproductive, as poor features require more elaborated classifiers, which are commonly more computationally intensive [10].

Table **3-2**: Cross-validation over young adults data without preprocessing.

	C_1	C_2	C_3
Specificity [%]	88.85 ± 2.40	90.32 ± 2.53	91.68 ± 1.35
Sensitivity [%]	88.74 ± 2.44	90.23 ± 2.27	92.13 ± 1.60
Accuracy [%]	88.79 ± 1.88	90.26 ± 2.05	91.93 ± 1.00

All features increased their performance after applying preprocessing on the young adults group data (Table **3-3**). Feature C_2 obtained the best accuracy with 96.36 % in validation. This result is consistent with the literature [9, Table 1]. Feature C_3 presented a small improvement in accuracy with respect to Table **3-2**, which is expected as it is an integration based feature (it can reduce high frequency noise as a low-pass filter). Table **3-3** also includes the mean and percentage of variation (abs-error, compared as absolute error) of obtained thresholds. For C_1 and C_2 the thresholds varied around 2.5 % along the 10-fold cross-validation. Note that the accuracy presented lower variations (in percentage) than the thresholds, meaning that the tested methodology is not sensitive to small changes in the

parameters. C_3 presented a considerably lower threshold variation, but it also had the lower accuracy.

Table **3-3**: Cross-validation with preprocessing.

	C_1	C_2	C_3
Specificity [%]	92.53 ± 2.62	96.34 ± 1.28	91.91 ± 1.20
Sensitivity [%]	92.60 ± 2.38	96.31 ± 1.67	91.85 ± 2.36
Accuracy [%]	92.60 ± 1.19	96.36 ± 0.78	91.90 ± 1.12
Threshold	14.42	107.20	7.30
abs-error [%]	2.71	2.49	0.14

The inclusion of the filtering stage defines the minimum allowed frequency sample. A preliminary analysis demonstrated that more elaborated filters (or with higher cut frequency values) did not improve the accuracy. This result is meaningful as it may suggest that a frequency sample of up to 11 Hz could be enough for fall detection (for a cut frequency of 5 Hz), with its respective computational cost reduction. The frequency sample is critical in wearable devices because (*i*) the system remains more time in idle state, and (*ii*) more separation among samples allows more computations (however, it may compromise the battery life).

3.3.1 Tests with elderly people

The group "Elderly" of Table **3-4** shows the specificity (ADL estimation) of testing the thresholds of Table **3-3** in the elderly people group (they only performed ADL, participant SE06 was excluded of this analysis). All features presented higher performance than the overall results of Table **3-3**, and again feature C_2 obtained the best performance with 99.14 % of specificity. An additional test performed after including elderly people in the training process gave similar threshold and accuracy values than those of Table **3-4**, i.e., training an algorithm with ADL of young adults is feasible to use in ADL of elderly people. This result is expected given that young adults usually get higher accelerations when performing the same activities.

A new test consisted in using the thresholds obtained in Table **3-3** to validate with subject SE06 data (the subject from elderly group that simulated falls). Group "SE06" in Table **3-4** shows the specificity, sensitivity, and accuracy of this test. Similar to [32], all features severely decreased their fall detection performance (sensitivity). C_2 maintained high specificity (100 %) and presented the higher sensitivity (80.00 %). Unfortunately, it is still unfeasible for real applications (1 in 5 falls was not detected), and it is far from the original results obtained with young adults. This result may be biased by the way subject SE06

Table **3-4**: Validation tests with only elderly people.

Group		C_1	C_2	C_3
Elderly	Specificity	97.79	99.14	97.17
SE06	Specificity	91.14	100	92.41
	Sensitivity	64.00	80.00	62.67
	Accuracy	77.92	90.26	77.92
SE06 Full	Specificity	91.14	97.47	92.41
	Sensitivity	65.33	81.33	64.00
	Accuracy	78.57	89.61	78.57

performed the falls (softening the fall with the arms).

Finally, thresholds obtained with cross-validation over young adults and elderly people (excluding SE06) were used as an attempt to increase the sensitivity values with subject SE06 (group "SE06 full" in Table **3-4**). However, none of the features presented a significant improvement, i.e., other authors must take into account that not including falls with elderly people will severely bias the behavior of fall detection algorithms.

3.3.2 Individual activity analysis

A final close review of individual activities of SisFall provided the following findings: (*i*) ADL and falls simulated by elderly people were smaller in amplitude than those simulated by young people. This suggests that algorithms trained with young people data tend to bias the thresholds upwards in amplitude, with the consequent increase in false negatives. (*ii*) As a preparation to this work we tested up to 28 features and most tended to fail in the same activities [9, Table 4]. Fig. **3-2** shows box-plots of the maximum value obtained per activity with C_2 feature (with young adults group exclusively). Note that only few activities severely crossed the threshold (horizontal red line): jogging quickly (D04), jump (D18), and fall backward when trying to sit (F11).

There are not many works focused on the types of falls elderly people suffer (most authors limited to perform same activities of previous works). But based on our survey, previous works [5] and our findings, other authors may focus their own work on some representative activities where the algorithms are prone to fail.

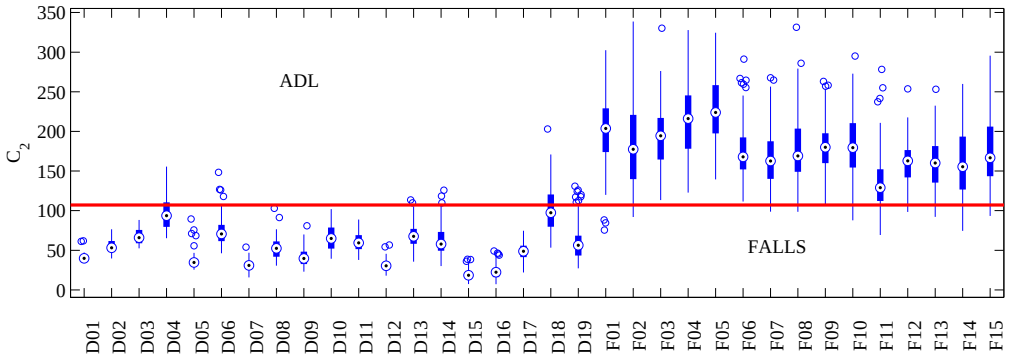


Figure **3-2**: **Box-plots of C_2 per activity.** Maximum value per activity obtained with C_2. Most threshold crossings (horizontal red line) are contained in activities D04, D18 and F11.

3.4 Summary

In this chapter, the SisFall dataset was tested with some of the most widely used features to detect falls. Two proof-of-principle experiments were performed: the effect of the preprocessing stage, and the importance of including elderly people data. Preprocessing explanations are commonly simplified in most approaches available in the literature. However, with a simple 4-th order Butterworth filter we obtained a 6 % increment of accuracy with the best feature (C_2). Preprocessing is crucial in fall detection as it defines the minimum acquisition frequency, that in this work we found at 11 Hz.

The classification threshold values obtained with young people data behaved well when we included elderly people ADL data (with up to 99 % of specificity). An expected result because young people tend to be faster than elderly people doing the same activities (higher peak acceleration expected). But when the algorithms were tested with subject SE06, a judo expert of 60 years old that performed falls, the sensitivity (fall detection) values significantly dropped down to 80 % and less. Bagalà et al. [32] obtained the same conclusion when testing state-of-the-art algorithms with real-life falls (but they did not release the dataset).

4 Energy based algorithm for fall detection

4.1 Introduction

In this chapter, we propose a double threshold fall detection strategy that can be implemented either in fixed self-developed devices or smartphones. The first threshold is a combination of the widely used acceleration peak and area under the curve features. This combination was selected after analyzing on which activities each feature was weak. The second threshold corresponds to a novel energy based strategy. We focused on two premises:

- A not too computationally complex solution (composed of only $O(n)$ computations and 1 s recording windows).
- The possibility of implementing it in different hardware systems (smartphones and self-developed embedded ones).

This methodology was validated with the MobiFall and SisFall datasets.

4.2 Individual activity analysis

Daily activities such as jumping and jogging have high acceleration peaks comparable with those obtained in some falls. Additionally, some falls such as falling from a chair may not have higher acceleration that several ADL. These are the main problems threshold based approaches face. Fig. **4-1** shows box-plots with the maximum values achieved when testing two different features per file of the 13 activities of MobiFall. The first feature was the sum vector magnitude, which is the root mean square (RMS) value of the acceleration signal in the three axis $\vec{a} = [a_x, a_y, a_z]^T \in \Re^3$ for a given time step k [9]:

$$E_1[k] = \sqrt{a_x[k]^2 + a_y[k]^2 + a_z[k]^2} \tag{4-1}$$

The second feature was the signal magnitude area, computed as the sum of integrals per axis [9]:

$$E_2[k] = \frac{1}{N_v} \sum_{i=x,y,z} \left(\int_{k-N_v}^{k} |a_i[n]| dn \right) \tag{4-2}$$

where N_v is the size of a sliding horizon window (here $N_v = 1$ s).

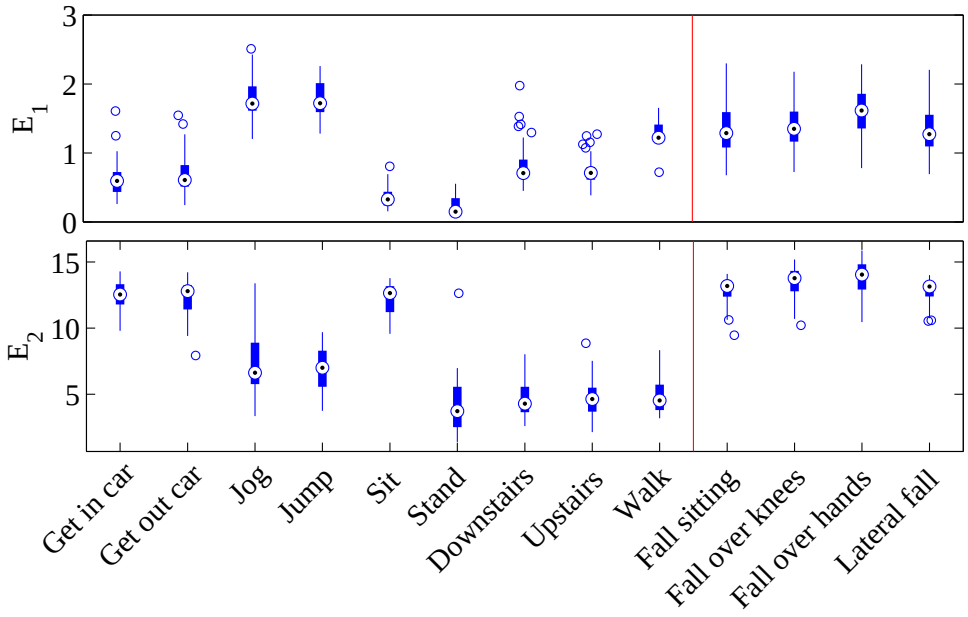


Figure **4-1**: Box-plots of two features separated by activity in the dataset MobiFall. Note how in both cases the maximum values of some ADL overlap with those of falls, meaning that a threshold is not a good classification option. However, also note that both features fail in the opposite activities.

Fig. **4-1**(Top) shows that for E_1 walk, jog and jump had higher overall acceleration than all falls. However, they presented good performance with E_2 (Fig. **4-1**(Bottom)), being getting in and out of a car and sitting the ones that failed. These results suggest that a product between both features would improve the separation among classes, as it will be shown in the following section.

4.3 Energy based fall detection

The proposed fall detection methodology consists of three stages: pre-processing, feature extraction and classification. The pre-processing stage consists of a simple to implement 4-th order Butterworth filter, and for E_1 a derivative is applied for removing the bias.

Once the acceleration data is pre-processed, both E_1 and E_2 features are computed, and a novel non-linear feature is also computed: $E_3 = E_1 \cdot E_2$. Given that both E_1 and E_2 are consistent in falls and fail in different activities, their multiplication reduces all ADL amplitude values while keeps high the ones of falls. Fig. **4-2**(Top) shows the same box-plot

of Fig. **4-1** with this new feature. Even all ADL reduced their values (and dispersion), jog and jump still overlap with the falls, being necessary a new feature to fix this problem.

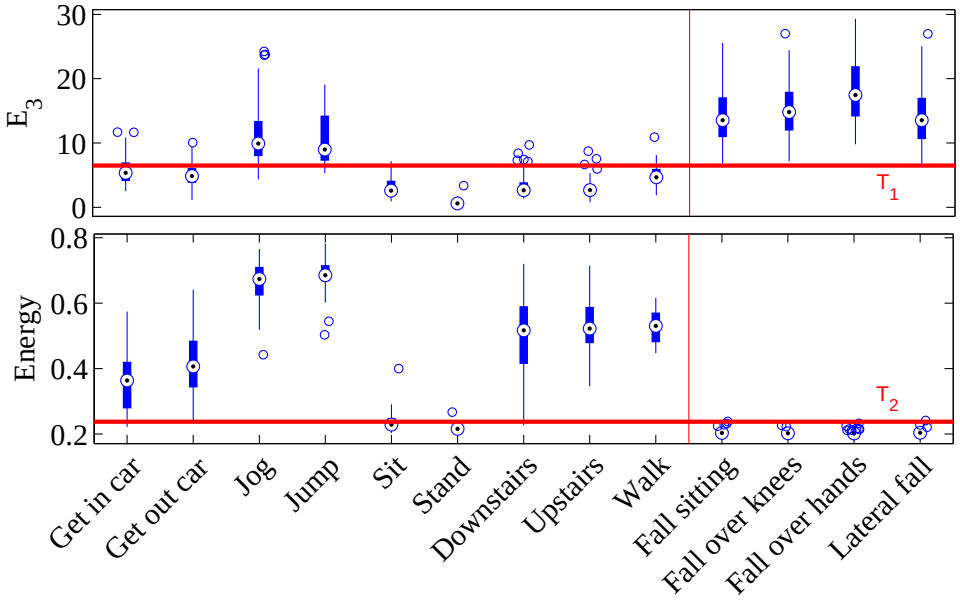


Figure **4-2**: Box-plots of the two new proposed features separated by activity in the dataset MobiFall. The Energy feature separates falls with almost all ADL except those of low energy (such as sit and stand, see the horizontal red lines –Thresholds), but E_3 does separate those, being the interception of both features ideal for classification.

A second new feature is proposed in order to avoid false positives due to high energy activities. The proposed feature computes the normalized energy of a set of time windows, with the purpose of differentiating short duration events from repetitive activities of high energy (such as walk or jog). First, the signal magnitude area E_2 of five consecutive non-overlapping windows is stacked in a vector $\vec{e}$:

$$\vec{e} = \left[E_2[k-4N_v], E_2[k-3N_v], \ldots, E_2[k]\right]^T \in \Re^5 \tag{4-3}$$

i.e., in practice it is computed each second (The same size N_v of the sliding window). Then, instant high energy activities (falls) are separated from periodic ones by normalization:

$$p = \frac{1}{5}\frac{\sum \vec{e}}{\max(\vec{e})} \tag{4-4}$$

giving a single value every second that can be compared with a threshold. Finally, a fall is detected if during each second a value of E_3 is higher than a threshold: $E_3[k] > T_1$, and p is below a second threshold: $p < T_2$. The need of five 1 s windows implies that a fall will be detected 5 s after it occurs (see Fig. **4-3**(Bottom)). A preliminary analysis with smaller

number and size of windows showed less accuracy.

Fig. **4-2**(Bottom) shows the key component of this strategy. All falls have lower normalized energy than most ADL, being simple to set a threshold just above them. Those activities with low energy (sit and stand) have also low acceleration values, being discarded by a threshold in E_3 (none overlaps with falls).

The selection of both thresholds can be automatically performed with a bi-level optimization, as follows:

1. Find the maximum E_3 value of each fall, and set T_1 as the minimum among these maxima (the one with most failure probability).
2. Move T_2 within its training span and select the value that provided the higher accuracy.
3. With the selected T_2 value, move T_1 within its current value (it only grows) and the maximum magnitude of ADL activities.
4. Iterate until convergence (both T_1 and T_2 vary less than 1 %). It is achieved after 3 or 4 iterations.

The first threshold (T_1) avoids false positives caused by low power energy activities, and the second (T_2) avoids false positives caused by high energy periodic activities. Because of this, both thresholds will converge close to the limits of fall activities (low false negative rate).

Fig. **4-3** shows an example of a fall after jogging caused by tripping, in SisFall dataset. Remember that a fall is present if both thresholds are crossed by their respective features. However, due to the 5 s window it may be tricky to see. Note how feature E_3 crosses the threshold (red horizontal line) at 4.7 s (tripping) and then during the fall with a maximum at 5.2 s. Meanwhile, in the Energy plot the threshold is crossed 5 s later.

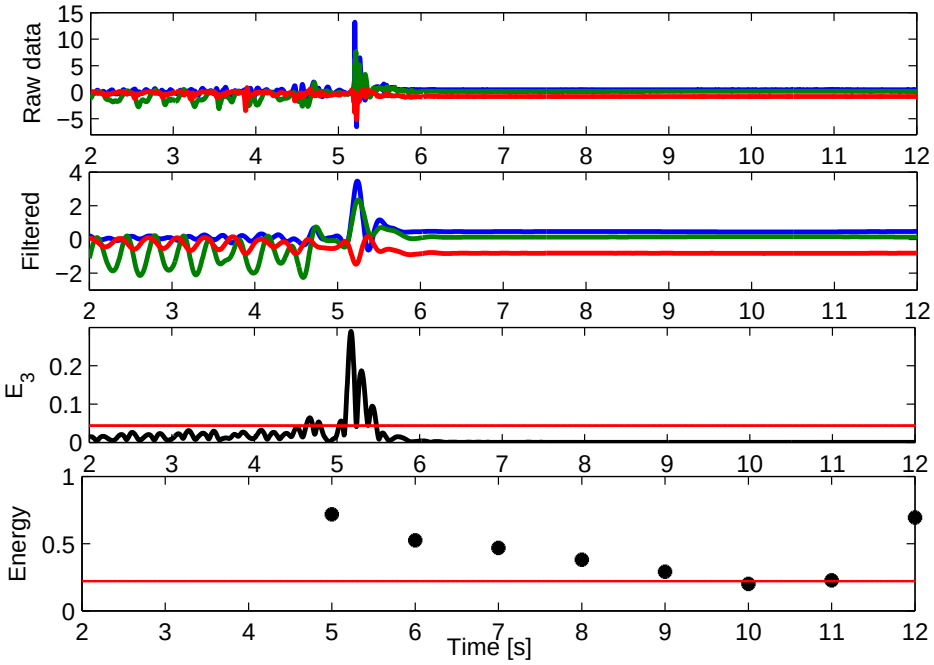


Figure **4-3**: Example of a fall detection with the proposed algorithm. The raw data of a person jogging, tripping and falling (top) is initially filtered. Then, features E_3 and Energy are computed and if both thresholds (horizontal red lines) are crossed (with 5 s difference) a fall is detected.

4.4 Results

In this section, the performance of the proposed methodology to differentiate between falls and ADL is shown. Sensitivity (SEN) and specificity (SPE) were calculated as specified in Eq. (3-1), and accuracy (ACC) was calculated using Eq. (3-2).

Table 4-1 shows mean and standard deviation of specificity, sensibility and accuracy achieved with a 10-fold cross-validation test with MobiFall dataset. The achieved accuracy over 99 % is comparable with results presented in [30, 33], but it was obtained with our less computationally intensive threshold-based approach.

Table **4-1**: Proposed methodology with MobiFall dataset. 10-fold cross validation.

Specificity [%]	99.35 ± 1.37
Sensitivity [%]	98.54 ± 2.48
Accuracy [%]	99.03 ± 1.13

The accuracy standard deviation in validation was 1.128. The mean values of the thresholds were $T_1 = 6.499$ and $T_2 = 0.2375$ (arbitrary units), and their variations along the 10-fols were 2.76 % and 2.45 % respectively. These results show stability in the solution with MobiFall dataset.

Table **4-2** shows mean and standard deviation results of training and validation with SisFall dataset, in same conditions as for Table **4-1**: 10-fold of cross-validation, 3924 files for training and 432 for validation at each row. For comparison purposes, we also included the results obtained with the features tested in Chapter 3. The only change made to the algorithm with respect to the parameters used for MobiFall dataset was that E_1 was implemented only over the horizontal plane (Eq. (4-1) without y component), this because unlike the smartphone the acquisition device was fixed at the center of the abdomen.

Table **4-2**: Proposed methodology evaluated in SisFall dataset compared to the features analyzed in Chapter 3. Same conditions of tests with MobiFall were guaranteed.

	C_1	C_2	C_3	Proposed
Specificity [%]	93.25 ± 1.53	96.83 ± 1.27	92.81 ± 1.21	98.55 ± 0.51
Sensitivity [%]	93.28 ± 2.80	96.66 ± 1.85	92.69 ± 2.56	95.31 ± 1.44
Accuracy [%]	93.27 ± 1.20	96.77 ± 0.726	92.76 ± 1.37	**97.27 ± 0.86**

Results of Table **4-2** show that despite the larger number of activities (13 in MobiFall vs. 34 in SisFall), the inclusion of elderly people, and the change in the device and its location did not severely affect the algorithm performance, with accuracy levels over 97 % and standard deviations lower than 1.4 %. The mean threshold values were $T_1 = 4.28$ and $T_2 = 0.22047$ (arbitrary units), with variations of 2.83 % and 1.68 % respectively. The lower performance in SisFall compared to MobiFall is expected as this database has almost 3 times more activity types (each dataset trained with its own data). However, they are still higher than those presented in the previous chapter.

Compared to the traditional features, the proposed algorithm provides higher accuracy values, with bias over specificity (desired in fall detection). These results together with those of MobiFall demonstrate the robustness of the proposed approach, and its robustness among different types of recorded data.

An additional validation was performed only with elderly people of SisFall dataset. A 99.39 % of specificity was achieved (only ADL). This means that the algorithm is feasible for implementation in the objective population.

4.5 Summary

In this chapter, a methodology for discriminating between falls and ADL was presented. The methodology is based on a simple to implement real-time pre-processing algorithm, and two thresholds for classification. Two novel threshold based features were proposed for detecting falls. The first one (E_3) comes from the product of the well known sum vector magnitude (E_1) and signal magnitude area (E_2). The second feature comes from the normalization of the signal magnitude area over five 1s-windows.

The proposed methodology was validated with the self developed SisFall dataset and the publicly available MobiFall dataset. It achieved similar results obtained by other authors using SVM (over 99 % with MobiFall and 97 % with SisFall), but with the low energy consumption provided by threshold based algorithms. In addition, the problem of classification caused by activities of the daily living of high acceleration was addressed and analyzed as a way to solve the specific problems this type of strategies have.

5 Walk and jog identification

5.1 Introduction

In previous chapters we observed how most failures on fall detection algorithms occur in few activities, and how most of these activities are periodic with high acceleration, such as walk and jog. The objective in this chapter is to determine if a subject is walking or jogging no matter the dataset used, i.e., the way the data was acquired. This knowledge will help (see Chapter 6) fall detection algorithms to reduce false positives due to high acceleration periodic activities. Additionally, as an interesting result of this work, the algorithms here proposed allow the user to determine in real-time the quality of walk or jog in terms of repeatability, i.e., a subject could maintain a regular movement and could be advised every time the speed changes.

This chapter begins with a wavelet based algorithm that provides a kurtosis metric. Then, given that the wavelet coefficient used is similar to the auto-correlation of the signal, a more efficient auto-correlation algorithm is proposed. Finally, a statistical analysis of the period of the activity (extracted from the auto-correlation) was performed for determining the robustness of the period as a measure of walk and jog, and their regularity.

5.2 Participants and datasets

The proposed methodology was tested with three datasets: a walk and jog dataset specifically generated for this work (called Sistemic), samples from MobiFall, and a third private dataset from Gepar Research Group (Gepar, Universidad de Antioquia, available at http://sistemic.udea.edu.co/investigacion/datos-caminar-y-trotar/?lang=en).

Sistemic dataset (recorded as part of this work, available at http://sistemic.udea.edu.co/investigacion/datos-caminar-y-trotar/?lang=en) consisted of ten volunteers (6 men and 4 women) that recorded five 20 s repetitions of walk and jog. Table **5-1** shows age, height, and weight details of the participants. The data were acquired with the same device used with SisFall (Chapter 2). But in this case both accelerometers were configured for ±8 g, 10 bits of ADC (scaling factor of 15.6 mg/LSB). The sampling frequency was 125 Hz according to [53].

The methodology was then validated with the other datasets. Gepar consisted of 2 women and 10 men that performed five repetitions per activity acquired with an embedded device [53]. The sample from MobiFall consisted of 4 women and 5 men, with a single record of 5 min walk, and three 30 s jog repetitions. Table **5-1** shows age, height, and weight details for both datasets.

Table **5-1**: Characteristics of the participants separated by dataset.

Dataset	Sex	Age	Height [m]	Weight [kg]
Sistemic	Female	21 – 40	1.55 – 1.66	48 – 58
	Male	23 – 43	1.64 – 1.80	56 – 73
Gepar	–	18 – 56	1.64 – 1.85	56 – 85
MobiFall	Female	26 – 36	1.60 – 1.70	50 – 90
	Male	22 – 32	1.69 – 1.89	64 – 102

5.3 Wavelet based detection

Given the periodic but not stable characteristics of walk and jog acquired with accelerometers (see Fig **5-4** for an example), an exhaustive analysis of wavelet coefficients was performed in order to determine common characteristics of the signal. Non-overlapping windows of 2 s length were used to estimate walk and jog in the same way a real-time device would do. The length of the windows was selected as the (rounded) minimum time required to walk three consecutive steps. The following algorithm was implemented:

1. Select the vertical or forward axis direction of the acceleration dataset (y or z in the device shown in Fig. **2-2**).

2. Decompose the 2 s window with a six levels mother wavelet *Biorthogonal 6.8*; and select detail coefficients depending on the sampling frequency of the acceleration signal. Preliminary tests showed that D_3 or D_4 are adequate for sampling frequencies below 100 Hz, and D_5 or D_6 for more than 100 Hz.

3. Compute the kurtosis of the selected detail coefficients. A boundary must be determined in order to differentiate jog from walk. This boundary can be easily selected for each specific device (dataset) by placing it between the 25 % quartile of walk, and the 75 % quartile of jog (see Fig. **5-1**).

Figure **5-1** shows the kurtosis based metric for the three datasets. Note how in all cases it is possible to set a threshold between walking and jogging. The three datasets show a similar

behavior for walking in terms of variability, but significant differences for jogging. A deeper analysis of the raw data shows how placing the device in the middle of the body provides a better waveform (see Chapter 2). Gepar dataset was acquired with a device fixed at one side of the participants, and MobiFall dataset was acquired with a non-fixed smartphone. Walking is not too affected because of less movement than jogging.

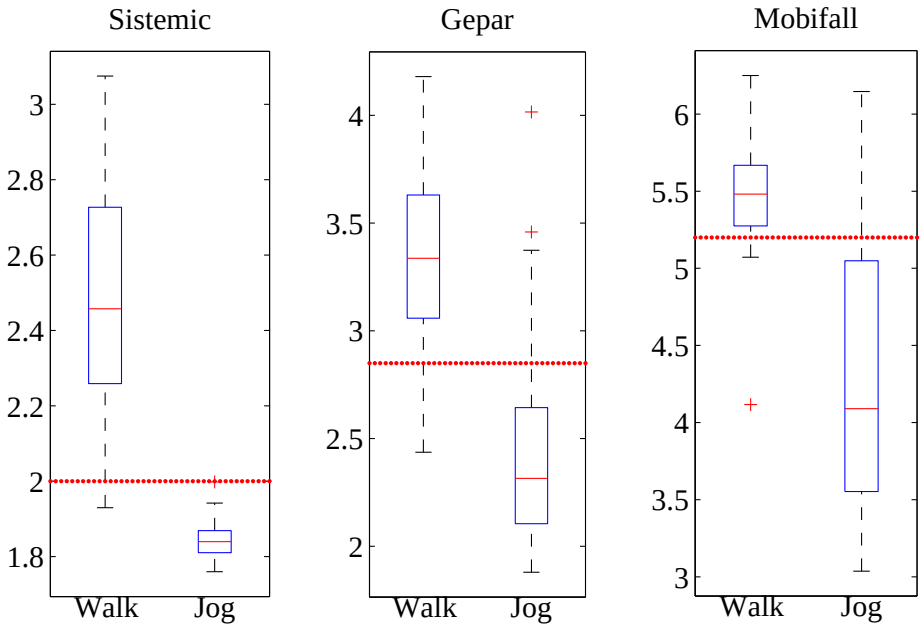


Figure **5-1**: Box-plots of the kurtosis based measures for the three datasets. The horizontal dotted red line shows the selected threshold value for each dataset. The inter-quartile range for walk is similar among datasets, but the medians differ obligating to re-calculate the threshold for each dataset.

Table **5-2** shows the accuracy achieved with the three datasets and the thresholds used (based on Fig. **5-1**). The variability of jogging affected the MobiFall dataset reducing its accuracy to 83 %. For Sistemic dataset, both accelerometers were tested but no differences were detected. Unfortunately, the implementation of this algorithm may not be feasible in real-time low-power devices. Then, a novel auto-correlation based strategy is proposed in the following section.

5.4 Auto-correlation based detection

A close analysis of the selected wavelet detail coefficient shows that it is similar to the auto-correlation of the signal, i.e., the same kurtosis-based metric provided with wavelets can be achieved with a different strategy:

1. Select the forward or vertical acceleration axis, and compute the algorithm for 2 s windows.
2. Perform a mean filter over short overlapped windows of approx. 0.1 s.
3. Obtain the auto-correlation $r(\tau)$ of the selected signal (z-axis for example). An efficient and not too computationally intensive option is the Wiener-Khinchin theorem, that computes the auto-correlation in the frequency domain signal, obtained with the FFT:

   $$Z(f) = \mathrm{FFT}[z(t)]$$
   $$S(f) = Z(f) \times Z^*(f)$$
   $$r(\tau) = \mathrm{IFFT}[S(f)]$$

   where IFFT$[\cdot]$ is the inverse FFT, and $Z^*(f)$ is the complex conjugate of $Z(f)$.
4. Compute the kurtosis of the auto-correlation function $r(\tau)$, and find the boundary that differentiates between jog and walk.

Fig. **5-2** shows how a 1 s window of walk raw data from Sistemic dataset is processed in order to obtain the kurtosis value. Fig. **5-2** (top) shows the 3-axis raw data of a walk activity, there is not evident periodicity of the signal in any of the three axis (x: blue, y: green, z: red). The y axis signal is selected for filtering with a mean filter, then it is normalized and average removed in order to obtain the result of Fig. **5-2** (middle), a period can be now observed but still local maxima and minima avoid a robust kurtosis estimation. Fig. **5-2** (bottom) shows the auto-correlation of the mean filtered signal, the resultant signal is clear and allows obtaining not also a stable kurtosis, but also the period of the walk with a simple zero crossing algorithm.

The same accuracy analysis of the wavelet algorithm was performed for the auto-correlation based algorithm. Again the accuracy was determined by a kurtosis threshold. The results are shown in Table **5-2**. This algorithm provided better results for Gepar and MobiFall, but reduced its accuracy for the Sistemic dataset. The overall improvement over the wavelet based algorithm can be explained with the robustness provided by the frequency domain analysis for calculating the auto-correlation, compared to the use of a single wavelet coefficient.

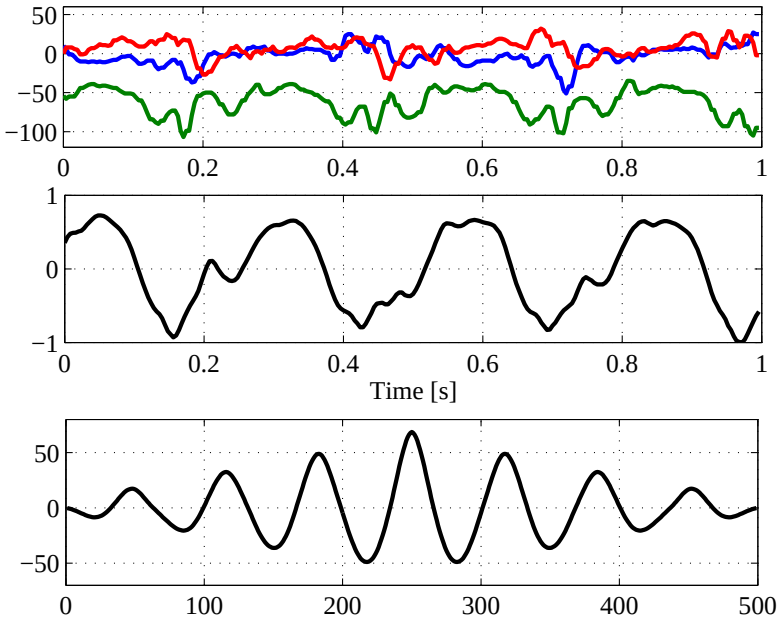


Figure **5-2**: Example of the preprocessing stage for a 1 s window of walk. (Top) Raw data in the three axis with amplitude measured in bits, there is not an evident periodicity of the signal. (Middle) Normalized y-axis mean filtered data, a periodic but not clean activity is observed. (Bottom) Auto-correlation of the filtered signal, the period may be easily computed from the zero crossing and the kurtosis gives stable values.

Table **5-2**: Accuracy achieved and threshold used for both algorithms validated with the three datasets.

Algorithm	dataset	Accuracy	Threshold
Wavelet	Sistemic	99 %	2
	Gepar	95 %	2.8
	MobiFall	83 %	5.4
Auto-correlation	Sistemic	95 %	2.9
	Gepar	100 %	9
	MobiFall	89 %	4.8

5.5 Statistical analysis of walk and jog

The auto-correlation performed above also provides the period of the original signal by applying a zero-crossing algorithm. Based on the periodicity of the signal, a statistical analysis of walk and jog was performed in order to analyze how the kurtosis provides a quality metric,

and how robust the threshold used is.

A factorial experimental design for type of activity and number of subjects in Sistemic dataset was performed in order to determine if there is a significant difference among subjects and activities. Subject 5 did not walk in the same way for all repetitions and was removed for this analysis. The resultant model was described by:

$$y_{ijk} = \mu + \tau_i + \beta_j + (\tau\beta)_{ij} + \epsilon_{ijk}$$

where y_{ijk} is the acceleration of activity $i = 1, 2$ (1 walk, 2 jog), for subject $j = 1, \ldots, 9$ and repetition $k = 1, \ldots, 5$. μ is the global mean, τ_i is the effect of the i-th activity, β_j is the effect of the j-th subject, $(\tau\beta)_{ij}$ is the multiplicative effect (interaction) between activities and subjects, and ϵ_{ijk} is the random error component (assumed Gaussian). The model was used to test the following hypothesis:

- $H_0 = \tau_1 = \tau_2 = 0$: There are not significant differences between walk and jog.
- $H_0' = \beta_1 = \beta_2 = \cdots = \beta_9 = 0$: There are not significant differences between the 9 subjects of the experiment.
- $H_0'' = (\tau\beta)_{1,1} = (\tau\beta)_{1,2} = \cdots = (\tau\beta)_{2,9} = 0$: There is not significant multiplicative effect between activities and subjects.

Table **5-3** shows the summary of the data computed with Statgraphics® software package. Note that the minimum walk value and the maximum jog value do not overlap, i.e., those activities can be accurately differentiated solely using their period.

Table **5-3**: Summary of the statistical analysis performed for Sistemic data.

Activity	Min	1st Q	Median	Mean	3rd Q	Max
Walk	**0.46**	0.50	0.55	0.55	0.58	0.71
Jog	0.32	0.36	0.37	0.37	0.38	**0.41**

A power transformation was performed in order to guarantee the two way variance assumptions: normality and constant variance. The residuals had white noise behavior, revealing independence among individuals and good model specification. Table **5-4** shows the analysis of variance. The p-values of zero indicate that there are significant differences among activities, among subjects, and that there exists multiplicative effect, i.e., H_0 is rejected in all cases.

Table **5-4**: Analysis of variance for Sistemic dataset.

Source	Sum Sq.	Df.	Mean Sq.	F-ratio	P-value
A: Activity	427.21	1	427.21	40553	0.0000
B: Subject	12.73	8	1.59	151.06	0.0000
AB	21.42	8	2.68	254.19	0.0000
Residuals	8.29	787	0.01		
Total	469.26	804			

Fig. **5-3** (left) shows the confidence intervals for walk and jog. It is clear that using a boundary is adequate for this task. Finally, Fig. **5-3** (right) shows the variance component plot for variability among subjects and activities. As expected the mean between subjects presented low variability compared to the mean between activities. These figures were computed with the data transformed back.

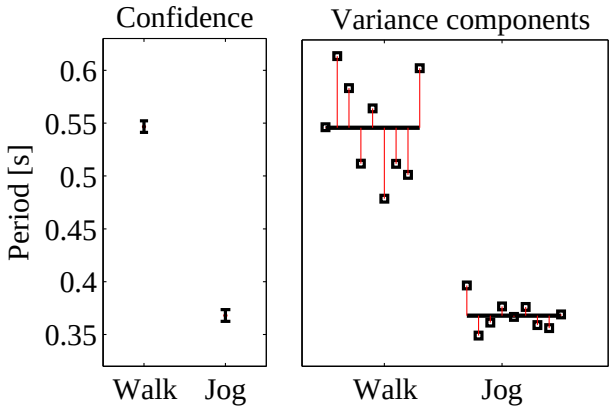


Figure **5-3**: (Left) Confidence intervals for walk and jog; significant differences are observed. (Right) Mean of each subject per activity referenced to the global means (horizontal lines); note how walking varies more than jogging.

5.6 Estimating the quality of the activity

Given the statistical analysis results, it is expected that variations in a single activity should be accounted by the proposed methodology. Fig **5-4** shows a failed 20s-jog repetition, where the participant stopped and started again, with the consequent lost of quality in the jogging.

Although the kurtosis can be used as quality measure, for visualization purposes the following quality measure was computed: $Q(T) = K_{\min}/K(T)$. with $K(T)$ the kurtosis of the 2 s

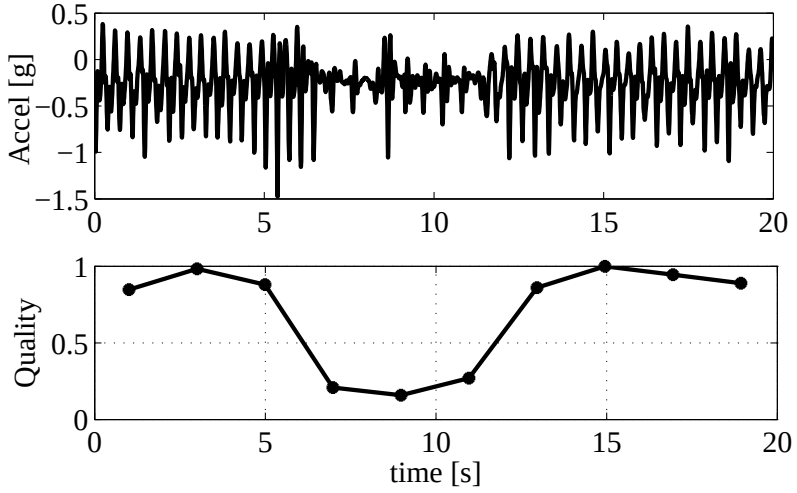


Figure **5-4**: Sistemic y-axis signal where the participant stopped jogging (top), and the quality measure detected the abnormal activity (bottom). Every 2 s the algorithm gives a measurement that besides the obvious stop in 6–12 s, also provides information about the regularity of the jog activity.

window, and $K_{\min}$ the minimum kurtosis measured for the participant, which is the more stable jogging detected. A value of 1 means that the activity is regular and for values close to 0 there is loss of periodicity. Note that several other quality measures can be implemented in the same way this measure was selected.

Finally, the same measure can be used in fall detection algorithms based on peak accelerations, in order to avoid false positives. A possible improvement could be to vary the threshold depending on the activity, given that it is expected to have higher accelerations when jogging.

5.7 Summary

In this chapter, a novel methodology for detecting and characterizing walk and jog based on non-peak acceleration features was proposed. This methodology is based on the kurtosis of wavelet coefficients, or with the auto-correlation of the acceleration signal. Both alternatives were validated with publicly available datasets (one developed as part of this thesis) and presented feasible results (over 83 % of accuracy and up to 100 %). The auto-correlation signal also provided a simple way to obtain the period of the activity with simple zero-crossing detection. One of the datasets was recorded with a smartphone in a pocket, i.e., the methodology is able to be implemented in different embedded systems.

As a secondary result, the period of the activity was used to perform a statistical analysis of walk and jog. This analysis demonstrated that walk and jog are significantly different, i.e., a threshold algorithm is robust; and that the participants are significantly different to each other, but not enough to overlap among activities allowing obtaining quality measures. In the following chapter, detection of periodic activities will be used to reduce false positives in elderly fall detection.

6 Kalman-filter-based fall detection

6.1 Introduction

Results of previous chapters show that a threshold based classification correctly separates most falls and ADL; and that most failures focus in periodic activities that can be independently detected. In this final chapter we make use of a Kalman-filter that provides an input to the feature extraction stage and a clean signal for a periodicity detector. The proposed approach combines bias variations of the signal and acceleration peaks. This increases the robustness of the feature extraction and allows simpler classifiers (Threshold-based instead of more elaborated ones). Fig. **6-1** shows a flow chart of the proposed methodology.

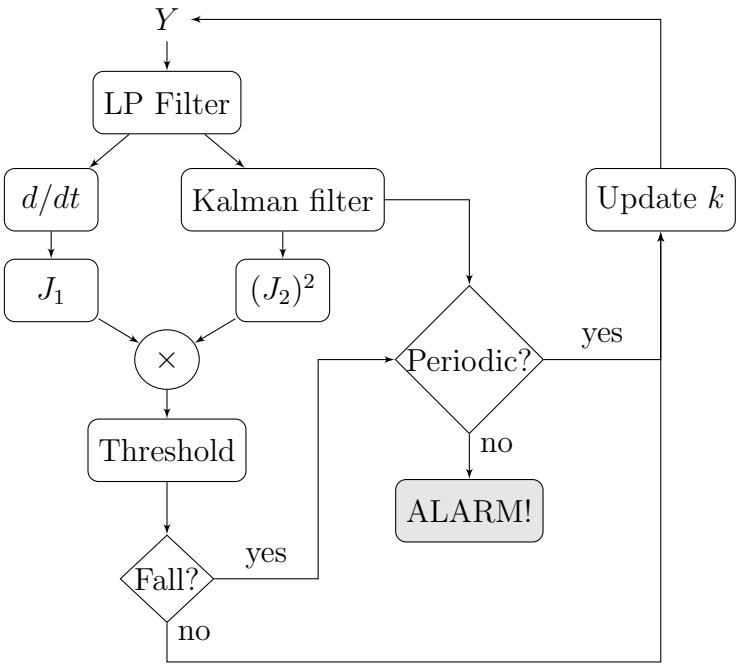


Figure **6-1**: **Proposed methodology.** The key feature of this approach is to detect individual (periodic) activities before taking a decision.

The proposed methodology consists of four stages: Preprocessing, feature extraction, classification, and periodic activity detection. For each time sample k, the raw data Y is initially

low-pass filtered. Then, it splits in bias removal and Kalman filtering, that feed both features (J_1 and J_2 respectively, see Eqs. (6-7) and (6-8)). The threshold-based classification is performed over an indirect feature: $J_3 = J_1 \cdot (J_2)^2$. If the resultant value crosses the threshold, the periodicity of the signal (extracted from the Kalman filter and a zero-crossing algorithm) is analyzed in order to determine if it is a false fall alert, or if indeed the alarm should be turned on. This methodology is explained in the following section.

6.2 Methods

6.2.1 Preprocessing and periodicity detection

The same 4-th order IIR Butterworth low-pass filter with a cut-off frequency of 5 Hz used in Chapter 3 removes instant peaks, as shown in Fig. **6-2**a. This filter is selected because: (*i*) It can be implemented in hardware; (*ii*) It does not require large computations in software; and (*iii*) Increasing the order or the cut-off frequency does not improve the accuracy, i.e., the sampling frequency remains low. The filtered data is then bias removed with a simple differentiation of consecutive samples, as it is needed to compute one feature (J_1). SisFall dataset was initially acquired at 200 Hz, however, the proposed methodology only requires 25 Hz to feed the filter. Then, all results presented in this chapter correspond to the proper downsampled signals.

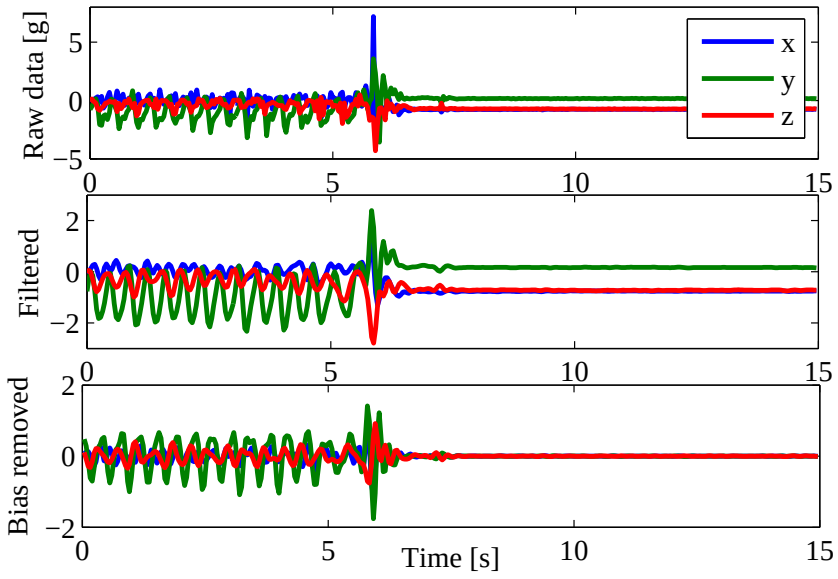


Figure **6-2**: **Example of preprocessing**. (Top) Raw acceleration data of jogging, tripping and finally falling (activity F05). (Middle) Filtered data. Note how the peak is smaller in amplitude and the signal is smoothed. (Bottom) A simple discrete differentiation is enough to remove the bias.

The second feature (J_2) is computed over the bias level, which is obtained with a Kalman filter. A Kalman filter [44] is an optimal quadratic estimator able to recover hidden states of a state-space model. We use the Kalman filter with two purposes: to recover the bias level variation, and to find the periodicity of the signal.

Let define the filtered acceleration data as $\vec{a}[k] = [a_x, a_y, a_z]^T \in \Re^{3\times 1}$ for time instant k. This data feeds the following autonomous state-space model:

$$\begin{aligned} \vec{x}[k] &= A\vec{x}[k-1] + \eta \\ \vec{y}[k] &= C\vec{x}[k] + \epsilon \end{aligned} \tag{6-1}$$

where the first three states of $\vec{x} \in \Re^{4\times 1}$ are used for classification and the fourth state is the basis of the periodicity detector (see Fig. **6-3**). As this Kalman filter is exclusively used for filtering (and not for feature extraction or classification), the state transition $A \in \Re^{4\times 4}$ and output $C \in \Re^{4\times 4}$ matrices are identity matrices. Finally, the output is defined as $\vec{y} = [a_x, a_y, a_z, a_y - b_{a_y}]^T \in \Re^{4\times 1}$, where the first three terms are the low-pass filtered acceleration data in the three axis, and the fourth output is the acceleration on vertical axis minus its current bias b_{a_y}, updated together with the feature.

This state-space model is affected by Gaussian measurement noise $\epsilon = \mathcal{N}(0, R)$, and Gaussian state uncertainty $\eta = \mathcal{N}(0, Q)$. The objective of the Kalman filter is to minimize the variance of the states $P \in \Re^{4\times 4}$, considering them as random variables with a Gaussian distribution: $\vec{x} = \mathcal{N}(\overline{x}, P)$.

The Kalman filter consists of five equations divided in two stages. The prediction stage of the Kalman filter *predicts* the current value of the states and their variance solely based on their previous values:

$$\vec{x}[k]^- = A\vec{x}[k-1] \tag{6-2}$$

$$P[k]^- = AP[k-1]A^T + Q \tag{6-3}$$

both $\vec{x}[k]^-$ and $P[k]^-$ are intermediate values that must be corrected based on the current data values:

$$G[k] = CP[k](CP[k]^-C^T + R)^{-1} \tag{6-4}$$

$$\vec{x}[k] = \vec{x}[k]^- + G[k](\vec{y}[k] - C\vec{x}[k]^-) \tag{6-5}$$

$$P[k] = (I_4 - G[k]^T C)P[k]^- \tag{6-6}$$

where $I_4 \in \Re^{4\times 4}$ is an identity matrix.

We only have two parameters to control the Kalman filter: the variance matrices Q and R. There are not rules to determine their values, but specifically for this problem they are not

difficult to define. Both are usually diagonal (no interaction among states), large values of Q and R tend to the original data: $\vec{x} \approx \vec{y}$, and they are also complementary, i.e., reducing any of them flats the states. As shown in Fig. **6-3** (Second and Third panels), the first three states are flat, and the fourth one seeks for periodic (sinusoidal shape) waveforms.

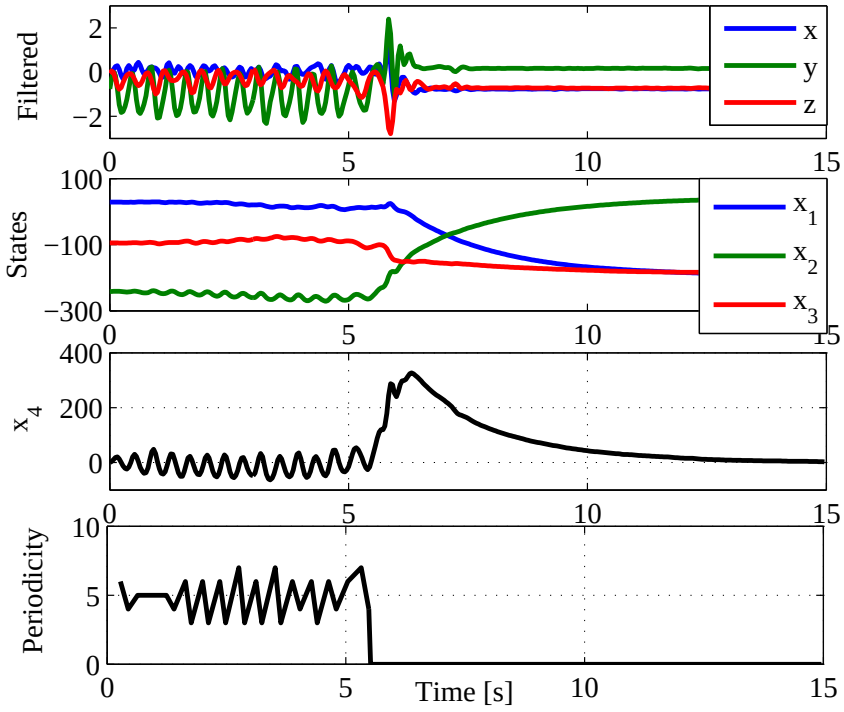


Figure **6-3**: **Kalman filtering**.(Top panel) Reference filtered acceleration data (Activity F05). (Second panel) First three states of the Kalman filter. It is clear that the filter recovers the variations in bias level of the signal. (Third panel) The forth state of the Kalman filter recovers a quasi sinusoidal signal during the first 6 s. Its objective is to remove bias to allow posterior zero crossing detection. (Bottom) Periodicity detector. The first 6 s the subject is jogging with a period of 10 time samples (twice zero crossing); when the subject falls it stops detecting periodicity.

Fig **6-3** (Bottom) shows how state x_4 tends to a zero-bias sinusoidal shape when the person walks or jogs. This allows implementing a simple zero-crossing periodicity detector. Note how the periodicity is lost when the person trips and falls. The periodicity detector focuses on the three seconds after a possible fall event. If during these 3 s window the periodicity is guaranteed we may expect that it was not a fall. The size of the window is selected as the minimum one to guarantee that the person continues slowly walking.

6.2.2 Feature extraction and classification

The feature extraction consists of a non-linear feature composed of two widely used ones, the sum vector magnitude and the standard deviation magnitude. The static sum vector magnitude is computed as the root-mean-square (RMS) of the static acceleration with previous bias removal:

$$J_1[k] = \text{RMS}(\tilde{a}[k]) \tag{6-7}$$

where in practice the bias is rejected with a simple differentiation: $\tilde{a}[k] = \vec{a}[k] - \vec{a}[k-1]$.

The standard deviation magnitude is dynamic and is computed at each time step k over a 1 s sliding window: $\tilde{x}[k] = [\vec{x}[k-N], \ldots, \vec{x}[k]] \in \Re^{3\times N}$, with $N = 25$ the size of the window (for a frequency sample of 25 Hz). This second feature is computed as follows:

$$J_2[k] = \text{RMS}(\text{std}(\tilde{x}[k])) \tag{6-8}$$

where $\text{std}(\cdot)$ is the standard deviation operator. The size of the sliding window is the same selected in Chapter 3.

The same sliding window can be used to determine the current bias value: $b_{a_y}[k] = \text{mean}(\tilde{x}[k])$. Fig. **6-4** shows both features with the jog-trip-fall example. The maximum values during jogging are half way of the fall in J_1, but they get clearly distant in J_2.
Finally, the classification stage is performed over an indirect feature:

$$J_3[k] = \max(\tilde{J}_1[k]) \cdot \max(\tilde{J}_2[k])^2 \tag{6-9}$$

With $\tilde{J}_i[k] \in \Re^{N\times 1}$ a sliding window with the last N values of the corresponding feature. This window is necessary as the Kalman filter takes some time to achieve the maximum, i.e., not always both metrics present a maximum at the same time. The objective of this product of features is to amplify the values of those activities where both features agree, and to minimize those where both features disagree. The square of J_2 gives it priority over J_1 as it is more accurate (note that both are greater than one as they are in bits).

The classification consists of a single threshold over $J_3[k]$ computed at each time step k. The value of the threshold is defined after a training process.

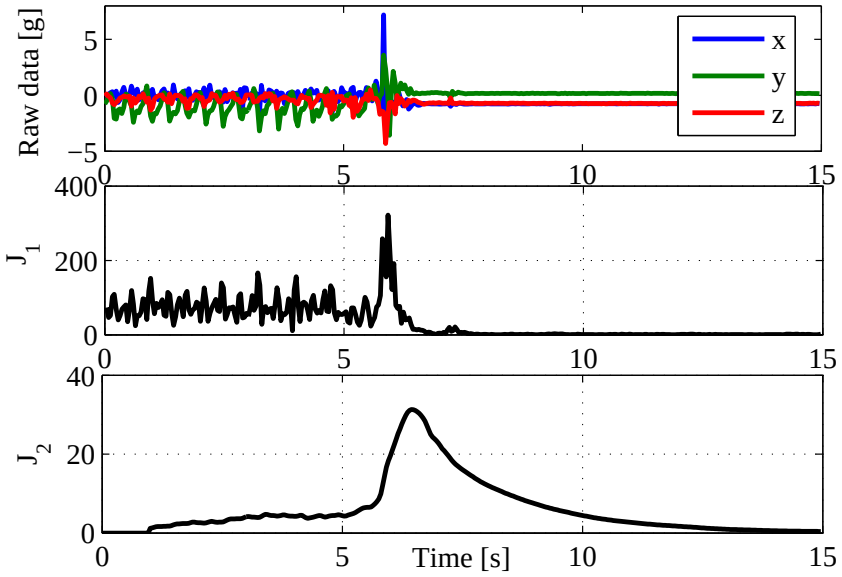


Figure **6-4**: **Feature extraction**. (Top) Reference filtered data (The subject is running, trips and falls). (Middle) Feature J_1 detects the fall as a large difference between its peak and jogging peaks. (Bottom) Feature J_2 has a similar shape but with a larger percentual difference.

6.2.3 Parameter selection

- **Kalman filter initialization:** The states can be initialized with zero and $P[0] = Q$, i.e., selecting uninformative priors. However, for faster convergence $x_2[0]$ and $b_{a_y}[0]$ can be initialized with -1 g (approx. -258 in bytes for the device configuration used with SisFall), which is the initial condition of the accelerometer in our device. Q and R can be computed with a simple heuristic process: For the first three states, initialize Q and R with identity matrices and reduce their standard deviation in scales of 10 until the accuracy stops increasing. For the fourth state, reduce Q and R until x_4 shows a sinusoidal shape in periodic signals (walk and jog). The final values used in this work were:

$$Q = 0.001^2 \times I_4 \qquad R = \begin{bmatrix} 0.05^2 & 0 & 0 & 0 \\ 0 & 0.05^2 & 0 & 0 \\ 0 & 0 & 0.05^2 & 0 \\ 0 & 0 & 0 & 0.01^2 \end{bmatrix} \tag{6-10}$$

  Note that all computations in both Matlab and the embedded device are performed in bits and not in gravities.

- **Threshold:** The same 10-fold cross-validation presented in Chapter 3 was performed here in order to obtain the optimal value of the threshold.

6.3 Results

6.3.1 Fall detection

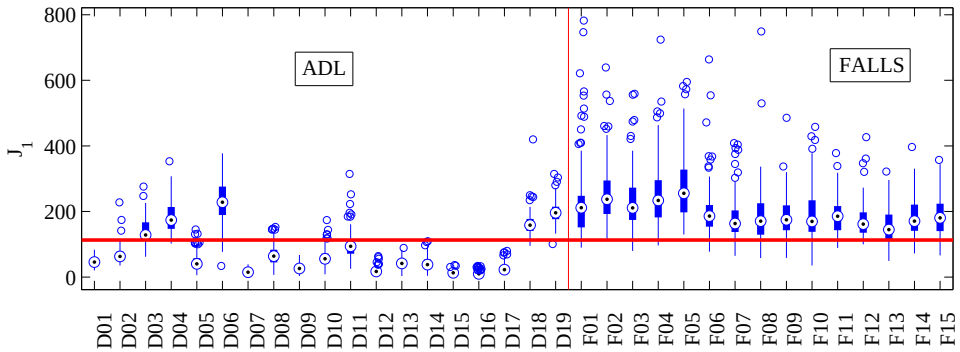
We initially tested the performance of the proposed algorithm without detecting periodic activities. Table **6-1** shows the validation results with SisFall dataset over a 10-fold cross-validation (451 files each). All subjects and activities available in the dataset were included in the cross validation. The low detection accuracy obtained with J_1 (around 86 %) would raise questions about its usefulness. However, note how J_3 is significantly higher than J_2 (99.3 % vs. 96.5 %), i.e., even J_1 is not a good metric, combined with J_2 it improves the individual accuracy values.

Table **6-1**: Test on SisFall dataset without periodicity detector.

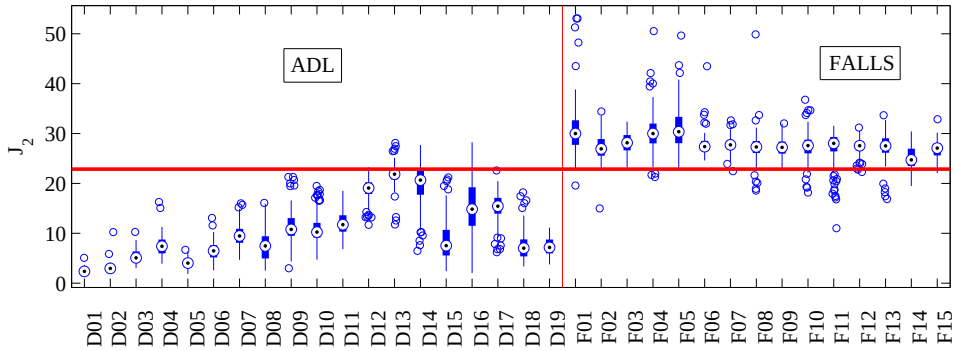
	J_1	J_2	J_3
Accuracy [%]	86.14 ±1.36	96.50 ±0.84	99.33 ±0.28
Threshold	110.88 ±3.23	22.88 ±0.027	42628 ±511.59

Fig. **6-5** shows an activity-by-activity analysis for the three metrics. The horizontal red line is the threshold for the best accuracy value, and the vertical red line divides ADL and falls. Comparing J_1 (Fig. 6.5(a)) and J_2 (Fig. 6.5(b)) we observe that J_1 largely fails in periodic ADL (D03, D04, D06, D18, and D19) but J_2 does not, and J_2 goes closer to the threshold in activities where J_1 does not (D16 for example). This separation was the basis to create J_3, it combines their results with a product but giving priority to J_2 (computed with square), given that it is more accurate. The small box in Fig. 6.5(c) shows how all activities are more separated from the threshold; and importantly, there are less outsiders in Falls (false negatives).

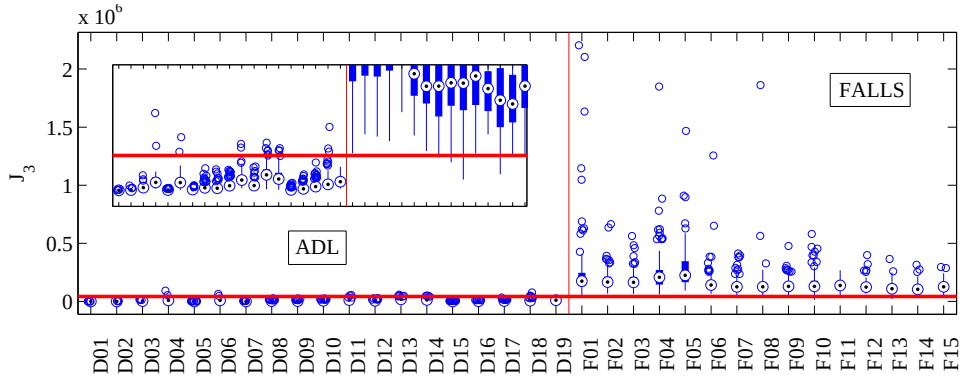
This initial result significantly improves those obtained with previous approaches tested in Chapter 3 (none of them achieved more than 96 %). In this case we included subject SE06 (an elderly that performed falls) in the cross validation, which means that this methodology is able to be used in the objective population.



(a) J_1



(b) J_2



(c) J_3

Figure **6-5**: Individual activity analysis of the proposed algorithm tested with SisFall. The horizontal red line corresponds to the optimal threshold value, and the vertical one separates ADL and falls. (a) J_1 has large errors on periodic activities, while (b) J_2 fails in those that change the body angle. (c) They provide J_3 with a better discriminant capability (the small box at the left shows a vertical zoom).

6.3.2 Fall detection with periodicity detector

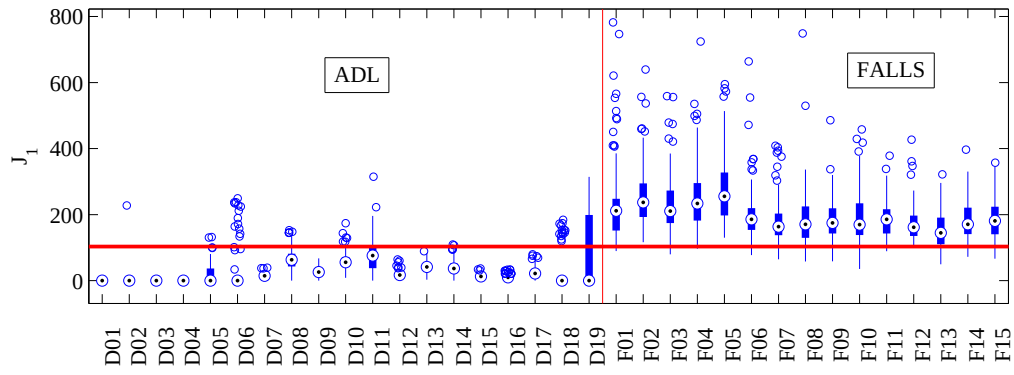
We then performed the same analysis but including the periodicity detector. The main purpose of this detector is to take J_1 to zero if a periodic activity is observed after a possible fall (false positive). Table **6-2** shows the validation results after a 10-fold cross-validation. Compared to the previous analysis, J_1 has a significant improvement (94.32 %). J_2 remains the same as the periodicity detector does not affect it. Even one would expect a significant improvement in J_3, this is not the case (however it is higher, with 99.4 % of accuracy) given that most of the outsiders were not located in the periodic activities (e.g., jump). However, Fig. 6.6(b) shows how the possibility of errors is lower given the larger distance from the threshold.

Table **6-2**: Test on SisFall dataset with periodicity detector.

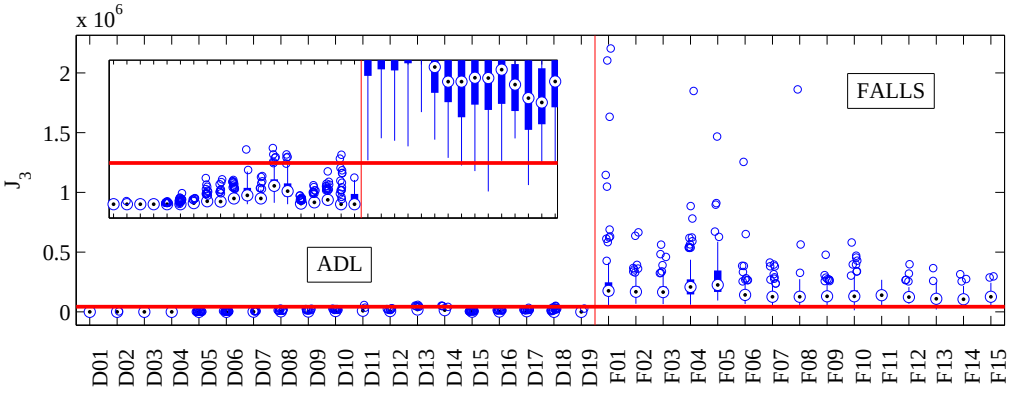
	J_1	J_2	J_3
Accuracy [%]	94.32 ±0.86	96.43 ±0.81	99.4 ±0.36
Threshold	103.03 ±0.02	22.914 ±0.11	42230 ±985.01

Fig. **6-6** shows the same individual activity analysis of Fig. **6-5** but with the periodicity detector in J_1. Feature J_2 was not included as it is not affected by the detector. Fig. 6.6(a) shows how activities D01 to D04 were turned to zero, as the detector confirmed that the subject was walking or jogging. Even there is a single outsider, Fig. 6.6(b) shows that it was cleared in J_3. In this case, J_3 shows more distance from the threshold than the previous test. This indicates that even the cross-validation did not show a significant improvement on accuracy, the inclusion of the periodicity detector increases the robustness of the algorithm to non-simulated unexpected situations (hits, for example). Importantly, none fall was turned to zero in Fig. 6.6(b), indicating that the periodicity detector was turned off in all periodic activities that finished in a fall.

The periodicity detector was active in 606 files (13.5 %). In SisFall dataset each activity has a limited number of repetitions. However, it is expected that a walk will last more than one minute, i.e., the posibility of failure is higher with activities that the subject performs regularly (such as walking).



(a) J_1 with periodicity detector



(b) J_3 with periodicity detector

Figure **6-6**: Individual activity analysis of the proposed algorithm including the periodicity detector. The horizontal red line corresponds to the optimal threshold value, and the vertical red line separates ADL and falls. (a) J_1 was turned to zero in all periodic ADL. (b) This allowed J_3 to increase the distance between most ADL and Falls.

6.4 On-line validation

We implemented the Kalman filter proposed above on the same device used to acquire SisFall (see Chapter 2 for details). We implemented it with the same parameters and sample frequency (25 Hz) defined above. The threshold for J_3 was set at 40,000. All calculations performed in the device were compared to results computed in Matlab. We obtained no significant differences.

Two validation tests were performed with the algorithm implemented on the device, in order to verify the off-line results presented in Table **6-2**:

- Individual activities: Six young adults (subjects SA03, SA04, SA05, SA06, SA09,

SA21) and one elderly person (subject SE06) performed three trials of all activities (except D17, getting in and out of a car, due to logistic issues). We verified on-line if the alarm was turned on (with a led light incorporated to the device). Additionally, all raw data and the device computations were recorded in text files.

- Full-day: Three elderly adults (they were not part of SisFall) carried the device for full days. They used the device permanently except during night sleep and shower (as the device is not water-proof). The files were cut in segments to avoid computational overloads (one hour of recording implied a text file of around 10 MB).

6.4.1 Individual activities

The volunteers performed a total of 18 ADL and 15 falls in the same way SisFall dataset was acquired (around 99 total trials per subject). They presented a total of 4 false positives and 1 false negatives. Subject SE06 did not show errors. All false positives were in D13 and D14 (bed related ones). Following Fig. 6.6(b), it is clear that these activities are commonly close to the threshold. A deeper analysis of this problem (which is not reflected in the following test) demonstrated that when a person moves on the bed, it is usual to separate the hip from the mattress and let it fall in the new position. The pad used for this experiment is harder than a mattress increasing the false positive probability. The overal results coincided with the expected statistics.

6.4.2 Full-day (pilot) tests

We invited three elderly people that were not part of SisFall acquisition (in order to avoid biases). Table **6-3** shows their age, weight and height. We asked them to behave normally while carrying the device during at least two days, and we checked the integrity of the devices every couple of hours.

Table **6-3**: Sex, age, height and weight of the participants in full-day activities.

Code	Sex	Age	Height [m]	Weight [kg]
SM01	Female	60	1.56	54
SM02	Female	68	1.46	56
SM03	Male	79	1.62	68

In the following, we present a summary of the activities they performed and the overall behavior of the system:

- SM01: She assisted to a Tae-Bo for adults class (INDER Medellín), and stayed at home cooking, cleaning and resting. She did not present false positives.

- SM02: She stayed most of the time cooking at home, cleaning and sit on the dinning room. She usually supports her belly against the kitchen or the table, it caused some false positives (4) of the system. She went out of her home two times, unfortunately both times the device got hits and lost the SD card, loosing all data. This is worrying as after an interview we concluded that she strongly hit the device in both cases pressumably againts furniture. We presumed that her low height together with the shape of her belly (rounded) incremented the risk of direct hits to the device.

- SM03: He did some trips to a business in the downtown and to the church. The rest of the time he stayed at home in bed or the dinning room. He did not have false positives in any activity. His trip to the downtown included stairs, two train trips and two bus trips. This trip is presented in Fig. **6-7**, note that despite the wide amount of activities, the levels of feature J_3 were not close to the threshold (40,000).

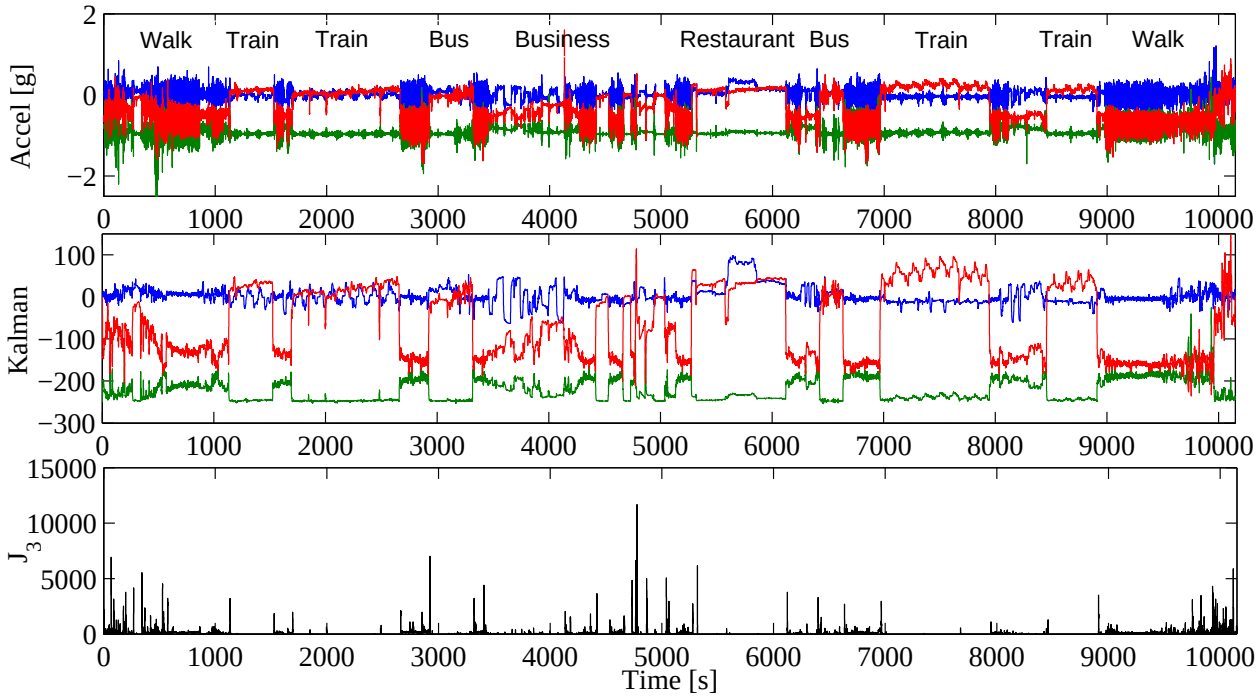


Figure **6-7**: **Trip to the downtown of SM03**. (Top panel) Raw acceleration data, 2 hours and 45 minutes of recording. (Second panel) First three states of the Kalman filter. (Third panel) feature J_3. It was always below the threshold (set at 40,000).

6.5 Summary

In this chapter we proposed a fall detection methodology, based on the conclusions obtained from previous chapters:

1. It uses the low-pass filter proposed in Chapter 3.
2. We selected two complementary features from Chapter 3, and we combined them in similar way as we already did in Chapter 4.
3. We included a periodic activity detector similar to those of Chapter 5, but with lower computational requirements.
4. We continued with a simple and non-computationally intensive threshold-based classification.

The basis of this methodology is a novel Kalman filter. The Kalman filter is not computationally intensive as it is Markovian, and it demonstrated to be stable with acceleration data. The proposed methodology was tested with SisFall dataset, we then validated it on-line after implementing it on the embedded device, and we finally performed real-life tests with three elderly people who used the device in their common life.

The most significant technical improvement of this approach is the way that a combined non-linear feature (J_3) provided higher accuracy (99.4 %) than the individual ones (94.3 % and 96.4 %). We obtained this feature after analyzing individually several features with each activity (finally keeping J_1 and J_2). J_1 comes from feature C_1 (but with 3 axis) in Chapter 3, and J_2 is the same feature C_2 but computed after the Kalman filtering stage. This pair was selected as they were highly complementary (each fails in different activities than the other one).

This methodology allowed reducing the frequency sample to just 25 Hz (as results of Chapter 3 suggested). The battery allowed more than 17 hours of continuous acquisition in the full-day tests (without saving data to a SD, it is expected to get longer). This final validation demonstrated that the proposed methodology can be used in real-life with objective population. However, even it behaved well with on-line simulated falls and real-life use, only real falls that may occur at any moment will show its real accuracy.

7 Conclusions and further research

7.1 General conclusions and main contributions

SisFall dataset In Chapter 2, we presented the new SisFall dataset acquired as part of this work. This responded to the need of a fall detection dataset with a large number of activities determined after a literature review and our own survey; well documented acquisition conditions (we recorded videos of each activity); and data from an embedded device fixed to the body (other datasets publicly available were recorded with smartphones). SisFall dataset consisted of 34 activities (falls and ADL) performed by 38 participants (15 of them had more than 60 years old). We made all files publicly available for the scientific community.

The dataset includes an elderly person (subject SE06) that performed ADL and falls. Even this person is a Judo expert (cannot be considered as representative of the elderly population), and that only one subject is not enough for most purposes, the acquired falls are wide more close to real-life conditions than what we could find available in the literature.

The dataset was tested in Chapter 3 with the most widely used features to detect falls. With this work, we demonstrated that a simple 4-th order Butterworth filter with a cut frequency of 5 Hz is enough to detect falls without loosing information. Additionally, we found that dynamic features based on statistical moments are the most accurate to classify among falls and ADL. However, we also found that training algorithms with young people is insufficient to obtain acceptable accuracy with the objective population.

Energy based fall detection In Chapter 4 we presented an energy-based fall detection algorithm. With this work, we used a static feature extraction characteristic (sum vector magnitude) together with an energy-based feature. This algorithm was tested in data from a Smartphone and an embedded device, with acceptable results. Moreover, our most important finding was that the combination of different features provides higher discrimination capabilities than the individual ones. This result led to a second conclusion, a threshold based classifier is enough to achieve accuracy levels of up to 99 %. The importance of this final finding relays in the low complexity (and consequently energy consumption) that threshold-based classifiers require.

Analysis of individual activities In Chapters 3 and 4 we found that most errors in the threshold based algorithms were focused in some individual activities, such as periodic ADL with high energy, namely walk, jog, or going up or down stairs. Consequently, in Chapter 5 we developed a novel methodology for detecting and characterizing walk and jog based on non-peak-based acceleration features. We demonstrated that with the Kurtosis of wavelet coefficients it was possible to obtain a measure to correctly identify these activities. However, we found that it was more stable to obtain the period of the acceleration signal using its auto-correlation.

Our methodology worked correctly with a fixed device (SisFall) and a smartphone (Mobifall). A posterior statistical analysis demonstrated that the period provides statistical significant differences among walk and jog. This methodology proved to be sensitive enough to provide a "quality of the activity" measure. We were able to determine on-line the regularity of the activity when the subject walked or jogged. This result could be useful for sports, allowing the person to maintain a regular jog rithm for long periods of time.

Kalman-filter-based fall detection In Chapter 6 we took advantage of the main improvements obtained during this work: Simple frequency filtering, a non-linear feature based on commonly used ones, theshold-based classification, and a periodicity detector to avoid false positives. With that, we generated a novel fall detection algorithm centered on a Kalman filter stage.

We selected the Kalman filter because its low computational cost and robustness, it provided an orientation level to a variance feature and at the same time a sinusoidal signal when the subject performed a periodic activity. This last result highly reduces the computational cost to obtain the period of the signal, as it avoids to compute Wavelets or auto-correlation.

The new non-linear feature used for this work was obtained in an intuitive way, and together with a theshold based classifier it achieved 99.4 % of accuracy with SisFall dataset. We then implemented this methodology in the embedded device and tested with full-day tests with objective population (two female and one male, all over 60 years all). We asked them to do what they use to, including traveling in train and bus, making exercise and cooking or cleaning. With a sampling frequency of 25 Hz (lower than most works in the literature), we obtained more than 17 continuous hours of acquisition (we recorded on-line, increasing the consumption) and the devices behaved as expected, just with a couple of false positives due to hits of the device during cooking. This final point is out of the scope of this work, and a good starting point for a future work.

7.2 Future work

Acquisition of real falls In Chapter 3 we demonstrated that setting an algorithm with young adults does not perform well with falls of elderly people. Even the falls were tested with a Judo expert who tried to minimize the impact, we would expect the same behavior of an elderly person trying to avoid the fall. So, even the methodology proposed in Chapter 6 seemed to solve this issue (the validation test with this subject presented 100 % of accuracy), it is necessary to increase the number of falls with elderly people to have a representative sample. However, with the impossibility of performing simulated falls with elderly people (the risk of accident is too high), with an average of one fall per year, it is unrealistic to expect acquiring confident data of real falls. Then, this remains as an open issue that must be solved in the near future.

Detection of hits on the device With the full-day tests performed in Chapter 6, we found that the position of the device (in the front) makes it propense to hits with some people. Our specific scenario was a 1.46 m height person that use to lean against the kitchen when cooking, hitting the device without noticing it. We acknowledge that with more intensive tests other situations may arise. A work focused on this kind of hit would help determining its differences with falls and avoiding false positives.

Sports and other uses of individual activity detection The "quality of activity" detector developed in Chapter 5 explores a growing research area: detecting individual activities for a wide variety of porpuses. They include sports performance, tracking people in closed spaces, rehabilitation, gait detection, among others.

Bibliography

[1] J. Masdeu, L. Sudarsky, and L. Wolfson, *Gait Disorders of Aging. Falls and Therapeutic Strategies.* Lippincot -Raven, Philadelphia, 1997.

[2] S. Lord, C. Sherrington, and H. Menz, *Falls in Older People: Risk Factors and Strategies for Prevention.*, 1st ed. Cambridge University Press, 2001.

[3] B. Vellas, S. Wayne, L. Romero, R. Baumgartner, and P. Garry, "Fear of falling and restriction of mobility in elderly fallers," *Age and Ageing*, vol. 26, no. 3, pp. 189–193, 1997.

[4] K. Delbaere, G. Crombez, G. Vanderstraeten, T. Willems, and D. Cambier, "Fear-related avoidance of activities, falls and physical frailty. a prospective community-based cohort study," *Age and Ageing*, vol. 33, no. 4, pp. 368–373, 2004.

[5] S. Lord, J. Ward, P. Williams, and K. Anstey, "An epidemiological study of falls in older community-dwelling women: the randwick falls and fractures study," *Australian Journal of Public Health*, vol. 17, no. 3, pp. 240–245, 1993.

[6] R. Igual, C. Medrano, and I. Plaza, "Challenges, issues and trends in fall detection systems," *BioMedical Engineering OnLine*, vol. 12, pp. 1–24, 2013.

[7] S. Brownsell, D. Bradley, R. Bragg, P. Catlin, and J. Carlier, "Do community alarm users want telecare?" *Journal of Telemedicine and Telecare*, vol. 6, no. 4, pp. 199–204, 2000.

[8] T. Shany, S. J. Redmond, M. R. Narayanan, and N. H. Lovell, "Sensors-based wearable systems for monitoring of human movement and falls," *IEEE Sensors Journal*, vol. 12, no. 3, pp. 658 – 670, 2012.

[9] N. Pannurat, S. Thiemjarus, and E. Nantajeewarawat, "Automatic fall monitoring: A review," *Sensors*, vol. 14, pp. 12 900–12 936, 2014.

[10] M. A. Habib, M. S. Mohktar, S. B. Kamaruzzaman, K. S. Lim, T. M. Pin, and F. Ibrahim, "Smartphone-based solutions for fall detection and prevention: Challenges and open issues," *Sensors*, vol. 14, no. 4, pp. 7181–7208, 2014.

[11] G. Diraco, A. Leone, and P. Siciliano, "An active vision system for fall detection and posture recognition in elderly healthcare," in *Design, Automation & Test in Europe Conference & Exhibition (DATE)*, 2010, pp. 1536 – 1541.

[12] M. Humenberger, S. Schraml, C. Sulzbachner, A. N. Belbachir, g. Srp, and F. Vajda, "Embedded fall detection with a neural network and bio-inspired stereo vision," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2012, pp. 60 – 67.

[13] M. Yu, A. Rhuma, S. M. Naqvi, L. Wang, and J. Chambers, "A posture recognition-based fall detection system for monitoring an elderly person in a smart home environment," *IEEE Transactions on Information Technology in Biomedicine*, vol. 16, pp. 1274 – 1286, 2012.

[14] G. M. Henao, C. L. Curcio Borrero, and J. F. Gómez Montes, "Consecuencias de las caídas en ancianos institucionalizados," *Revista de la asociación Colombiana de Gerontología y Geriatría*, vol. 23, pp. 1221–1233, 2009.

[15] I. Cleland, B. Kikhia, C. Nugent, A. Boytsov, J. Hallberg, K. Synnes, S. McClean, and D. Finlay, "Optimal placement of accelerometers for the detection of everyday activities," *Sensors*, vol. 13, no. 7, pp. 9183–9200, 2013.

[16] F. Büsching, U. Kulau, M. Gietzelt, and L. Wolf, "Comparison and validation of capacitive accelerometers for health care applications," *Computer Methods and Programs in Biomedicine*, vol. 106, no. 2, pp. 79–88, 2012.

[17] F. R. Allen, E. Ambikairajah, N. H. Lovell, and B. G. Celler, "Classification of a known sequence of motions and postures from accelerometry data using adapted gaussian mixture models," *Physiological Measurement*, vol. 27, pp. 935–951, 2006.

[18] L. Gao, A. K. Bourke, and J. Nelson, "A comparison of classifiers for activity recognition using multiple accelerometer-based sensors," in *11th International Conference on Cybernetic Intelligent Systems (CIS), IEEE*, 2012, pp. 149–153.

[19] L. Gao, A. Bourke, and J. Nelson, "Evaluation of accelerometer based multi-sensor versus single-sensor activity recognition systems," *Medical Engineering & Physics*, vol. 36, pp. 779–785, 2014.

[20] J. Yuan, K. K. Tan, T. H. Lee, and G. C. H. Koh, "Power-efficient interrupt-driven algorithms for fall detection and classification of activities of daily living," *IEEE Sensors Journal*, vol. 15, no. 3, pp. 1377 – 1387, 2015.

[21] M. Kaenampornpan, T. Anuchad, and P. Supaluck, "Fall detection prototype for thai elderly in mobile computing era," in *8th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTICON)*, 2011, pp. 446 – 449.

[22] G. A. Koshmak, M. Linden, and A. Loutfi, "Evaluation of the android-based fall detection system with physiological data monitoring," in *35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2013, pp. 1164 – 1168.

[23] M. Tolkiehn, L. Atallah, B. Lo, and G.-Z. Yang, "Direction sensitive fall detection using a triaxial accelerometer and a barometric pressure sensor," in *33rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBC*, 2011, pp. 369 – 372.

[24] A. Sorvala, E. Alasaarela, H. Sorvoja, and R. Myllylä, "A two-threshold fall detection algorithm for reducing false alarms," in *6th International Symposium on Medical Information and Communication Technology (ISMICT)*, 2012, pp. 1 – 4.

[25] K. Frank, M. J. Vera, P. Robertson, and T. Pfeifer, "Bayesian recognition of motion related activities with inertial sensors," in *12th ACM International Conference on Ubiquitous Computing UbiComp*, 2010, pp. 445–446.

[26] Smarthip. Firebird Medical. Last view March 31, 2016. [Online]. Available: http://www.firebirdmedical.com/products.html

[27] Vigi'fall. Consorcio de empresas tecnológicas FallWatch, Vigilio Telemedical. Last view March 31, 2016. [Online]. Available: http://www.vigilio.fr/index.php?lang=EN

[28] Philips lifeline. Philips. Last view March 31, 2016. [Online]. Available: https://www.lifeline.philips.com/safety-solutions

[29] Fall-bracelet. Neat Group. Last view March 31, 2016. [Online]. Available: http://www.neat-group.com/es/en/radio-transmitters/fall-bracelet/

[30] R. Igual, C. Medrano, and I. Plaza, "A comparison of public datasets for acceleration-based fall detection," *Medical Engineering and Physics*, vol. 37, no. 9, pp. 870–878, 2015.

[31] T. O'Neill, J. Varlow, A. Silman, J. Reeve, D. Reid, C. Todd, and A. Woolf, "Age and sex influences on fall characteristics," *Annals of the Rheumatic Diseases*, vol. 53, pp. 773–775, 1994.

[32] F. Bagalà, C. Becker, A. Cappello, L. Chiari, K. Aminian, J. M. Hausdorff, W. Zijlstra, and J. Klenk, "Evaluation of accelerometer-based fall detection algorithms on real-world falls," *Plos one*, vol. 7, no. 5, p. e37062, 2012.

[33] G. Vavoulas, M. Pediaditis, C. Chatzaki, E. Spanakis, and M. Tsiknakis, "The MobiFall dataset: Fall detection and classification with a smartphone," *International Journal of Monitoring and Surveillance Technologies Research*, vol. 2, no. 1, pp. 44–56, 2014.

[34] C. Medrano, R. Igual, I. Plaza, and M. Castro, "Detecting falls as novelties in acceleration patterns acquired with smartphones," *Plos One*, vol. 9, no. 4, p. e94811, 2014.

[35] L. J. Kau and C. S. Chen, "A smart phone-based pocket fall accident detection, positioning, and rescue system," *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 1, pp. 44–56, 2014.

[36] G. Cola, M. Avvenuti, A. Vecchio, G.-Z. Yang, and B. Lo, "An on-node processing approach for anomaly detection in gait," *IEEE Sensors Journal*, vol. 15, no. 11, pp. 6640 – 6649, 2015.

[37] M. Oner, J. A. Pulcifer-Stump, P. Seeling, and T. Kaya, "Towards the run and walk activity classification through step detection - an android application," in *34th Annual International Conference of the IEEE EMBS*, 2012, pp. 1980 – 1983.

[38] D. W. T. Wundersitz, P. B. Gastin, C. Richter, S. J. Robertson, and K. J. Netto, "Validity of a trunk-mounted accelerometer to assess peak accelerations during walking, jogging and running," *European Journal of Sport Science*, vol. 2014, pp. 382–390, 2014.

[39] C. M. Clements, M. J. Buller, A. P. Welles, and W. J. Tharion, "Real time gait pattern classification from chest worn accelerometry during a loaded road march," in *34th Annual International Conference of the IEEE EMBS*, 2012.

[40] A. Godfrey, A. Bourke, G. Ólaighin, P. van de Ven, and J. Nelson, "Activity classification using a single chest mounted tri-axial accelerometer," *Medical Engineering and Physics*, vol. 33, pp. 1127–1135, 2011.

[41] A. Yazar, F. Keskin, B. U. Toreyin, and A. E. Cetin, "Fall detection using single-tree complex wavelet transform," *Pattern Recognition Letters*, vol. 34, pp. 1945–1952, 2013.

[42] M. Nagabushanam and S. Ramachandran, "Fast implementation of lifting based 1D/2D/3D DWT-IDWT architecture for image compression," *International Journal of Computer Applications*, vol. 51, pp. 35–41, 2012.

[43] C.-H. Hsia and J.-M. Guo, "Efficient modified directional lifting-based discrete wavelet transform for moving object detection," *Signal Processing*, vol. 96, pp. 138–152, 2014.

[44] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.

[45] F. Bagalà, J. Klenk, A. Cappello, L. Chiari, C. Becker, and U. Lindemann, "Quantitative description of the lie-to-sit-to-stand-to-walk transfer by a single body-fixed sensor," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 21, no. 4, pp. 624–633, 2013.

[46] K. Berg, S. Wood-Dauphinee, J. Williams, and B. Maki, "Measuring balance in the elderly: validation of an instrument." *Canadian Journal of Public Health*, vol. 83, no. Suppl 2:S7-11, 1992.

[47] A. M. Otebolaku and M. T. Andrade, "User context recognition using smartphone sensors and classification models," *Journal of Network and Computer Applications*, vol. 66, pp. 33–51, 2016.

[48] D. Novak, P. Rebersek, S. M. M. De Rossi, M. Donati, J. Podobnika, T. Beravs, T. Lenzi, N. Vitiello, M. C. Carrozza, and M. Muniha, "Automated detection of gait initiation and termination using wearable sensors," *Medical Engineering & Physics*, vol. 35, no. 12, pp. 1713–1720, 2013.

[49] X. Yuan, S. Yu, Q. Dan, G. Wang, and S. Liu, "Fall detection analysis with wearable mems-based sensors," in *16th International Conference on Electronic Packaging Technology (ICEPT)*, 2015, pp. 1184–1187.

[50] G. Vavoulas, M. Pediaditis, E. G. Spanakis, and M. Tsiknakis, "The MobiFall dataset: An initial evaluation of fall detection algorithms using smartphones," in *13th International Conference on Bioinformatics and Bioengineering (BIBE)*, 2013, pp. 1 – 4.

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

[52] N. Noury, A. Fleury, P. Rumeau, A. Bourke, G. Laighin, V. Rialle, and J. Lundy, "Fall detection – principles and methods," in *29th Annual International Conference of the IEEE EMBS*, 2007, pp. 1663 – 1666.

[53] M. Vallejo, C. V. Isaza, and J. D. López, "Artificial neural networks as an alternative to traditional fall detection methods," in *35th Annual International Conference of the IEEE EMBS*, 2013, pp. 1648–1651.