M 2015

U.PORTO
FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

POSITION REFERENCE WITH DIRECTION ESTIMATOR FOR INDOOR LOCATION SYSTEMS

JOÃO MANUEL JORGE BARBOSA
DISSERTAÇÃO DE MESTRADO APRESENTADA
À FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO EM
ÁREA CIENTÍFICA

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Position reference with direction estimator for Indoor Location Systems

João Manuel Jorge Barbosa

Mestrado Integrado em Engenharia Eletrotécnica e de Computadores

Supervisor at FEUP: Rui Esteves Araújo (PhD)

Supervisor at Fraunhofer AICOS: Filipe Sousa (M.Sc.)

July 30, 2015

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO

MIEEC - MESTRADO INTEGRADO EM ENGENHARIA ELETROTÉCNICA E DE COMPUTADORES 2014/2015

A Dissertação intitulada

"Position Reference with Direction Estimator for Indoor Location Systems"

foi aprovada em provas realizadas em 24-07-2015

o júri

Presidente Professor Doutor Paulo José Cerqueira Gomes da Costa
Professor Auxiliar do Departamento de Engenharia Eletrotécnica e de Computadores da Faculdade de Engenharia da Universidade do Porto

Professor Doutor Adriano Jorge Cardoso Moreira
Professor Associado do Departamento de Sistemas de Informação da Escola de Engenharia da Universidade do Minho

Professor Doutor Rui Manuel Esteves Araújo
Professor Auxiliar do Departamento de Engenharia Eletrotécnica e de Computadores da Faculdade de Engenharia da Universidade do Porto

O autor declara que a presente dissertação (ou relatório de projeto) é da sua exclusiva autoria e foi escrita sem qualquer apoio externo não explicitamente autorizado. Os resultados, ideias, parágrafos, ou outros extratos tomados de ou inspirados em trabalhos de outros autores, e demais referências bibliográficas usadas, são corretamente citados.

Autor - João Manuel Jorge Barbosa

Faculdade de Engenharia da Universidade do Porto

Resumo

Nas últimas décadas os Sistemas de Localização Indoor sofreram um progresso considerável. Gradualmente têm vindo a desempenhar um papel cada vez mais importante em todos os aspectos da vida quotidiana das pessoas incluindo, por exemplo, assistente de navegação, detecção de emergência, vigilância/acompanhamento de um alvo de interesse e também fornecendo informações de localização e publicidade. Por essas razões, soluções em tempo real precisas e fidedignas para serviços de localização indoor são necessárias mais que nunca. No entanto, nenhum dos sistemas já desenvolvidos é capaz de satisfazer todas as exigências. Como consequência, há um interesse permanente na concepção de um sistema que permita atingir todos os requisitos.

Uma vez que os smartphones estão equipados com vários sensores como acelerómetros, giroscópios e magnetómetros, é possível ter um sistema de localização indoor a funcionar no nosso telémovel pessoal. Contudo, há alguns problemas associados a esta solução baseada em sensores inerciais uma vez que estes sensores ruidosos vão introduzir erros de posição devido ao próprio ruído. Consequentemente, esses erros cumulativos vão gerar uma estimativa errada para a posição atual.

Esta dissertação tem como objectivo melhorar a precisão do Sistema de Localização Indoor, usando uma superficie inteligente que vai recolher os dados e calcular em tempo real a direcção do utilizador.

Neste documento será apresentado o estado de arte dos campos intervenientes desta solução e a metodologia adoptada para a implementação desta abordagem como uma referência de posição com estimador de direção para Sistemas de Localização Indoor.

A superficie inteligente não só funciona como uma referência absoluta de posição mas também permite detectar a direção da caminhada do utilizador. No final, o sistema poderá enviar todas as informações para o smartphone (utilizando por exemplo Bluetooth).

7 individuos com pesos entre 65 Kg e 95 Kg e com tamanho do sapato entre 39 e 44 (Euro sizes) testaram o sistema final. Os resultados obtidos foram bastantes satisfatórios. Foi aplicada uma solução de baixo custo que no final foi capaz de detectar com sucesso 4 direções e orientações diferentes. Além disso, o sistema foi desenhado por forma a diferenciar o local onde o pé atinge a superficie inteligente (centro, direita ou esquerda).

Abstract

In the past decades Indoor Location Systems were undergone considerable progress. Gradually they have been playing an increasingly important role in all aspects of people's daily lives including, for example, living assistant, navigation, emergency detection, surveillance/tracking of target-of-interest and also providing location based information and advertisement. For these reasons, reliable, accurate and real-time solutions for indoor tracking services are required even more strongly than ever. However, none of the systems already developed is capable of satisfy all the demands. As a consequence, there are a permanent concern on designing a system that will achieve all the requirements.

Since smartphones are equipped with several sensors like accelerometer, gyroscope and magnetometer it is possible to have an indoor location system running in our personal smartphone. Nevertheless, there are some problems associated to this solution based in inertial sensors once these noisy sensors will introduce position errors due to noise itself. Consequently, these cumulative errors will generate a wrong estimation for current position.

This dissertation aims to improve Indoor Location System accuracy by using a smart floor that will be collecting data and calculating in real-time the user's direction.

In this document, the state of the art of the intervening fields of this solution and the methodology adopted for the implementation of this approach as a position reference with direction estimator for Indoor Location Systems will be showcased.

The intelligent surface works as an absolute position reference and also detects person's walking direction. In the end, the system could send all the information to the smartphone (using for example Bluetooth).

7 People with weights between 65 Kg and 95 Kg, and shoe sizes from 39 to 44 (Euro sizes) tested the final system. The results were quite satisfactory. It was applied a low cost solution that in the end was capable of successfully detect 4 different directions and orientations. Besides, the system was design in order to differentiate the point where the foot reaches the intelligent surface (centre, right or left).

Acknowledgements

Esta é provavelmente um dos momentos mais complicados da dissertação pois existe uma grande probabilidade de me esquecer de alguém nos agradecimentos.

Para começar, gostaria de agradecer aos meus orientadores pelo apoio e supervisionamento prestado ao longo destes meses. Ao Professor Rui Esteves Araújo, agradeço os conselhos oportunos que acabaram por ser essenciais para o sucesso deste trabalho. Relativamente ao Eng. Filipe Sousa, queria agradecer também o seu acompanhamento quase diário. Com o seu supervisionamento, fui capaz de manter todas as tarefas dentro dos prazos bem como os seus conselhos ajudaram-me a manter o trabalho no caminho certo e a atingir os objectivos propostos.

De seguida, queria agradecer à minha familia pelo apoio incondicional prestado. Pelo carinho e pelo conforto, e acima de tudo pela compreensão, principalmente nos momentos de maior tensão. Obrigado aos meus pais e irmão.

Não esquecer ainda de todos os meus amigos em geral. Em momentos de tanta responsabilidade, algum lazer é fundamental. Queria agradecer a todos vós com especial obrigado ao Cruz, Maketista e Mike. Obrigado pelos 5 anos a aturarem-me. Que venham muitos mais. Também queria agradecer aos Fanfarrões por me terem estrategicamente desencaminhado em alturas de maior tensão e stress. Sem vocês muito provavelmente teria desistido.

Aos velhos amigos Pedro, Rui e Ferreira por terem estado sempre presentes não só ao longo destes meses mas também ao longo de todo o meu curso académico. Obrigado por aquelas noites míticas em Coimbras.

À pessoa mais importante, e porque não precisa de muitas palavras, deixo o meu mais profundo obrigado à minha namorada.

A toda a equipa da Fraunhofer, queria também demonstrar o meu obrigado pelo excelente ambiente proporcionado ao longo destes meses. Foi gratificante trabalhar diariamente nesta instituição.

A todos um grande obrigado,

João Barbosa

"If I have seen further,
it is by standing on the shoulders of giants."

Isaac Newton

Contents

List of Figures

List of Tables

List of Abbreviations

ADC	Analog-to-Digital Converter
AOA	Angle of Arrival
AP	Access Point
API	Application Programming Interface
CDMA	Code Division Multiple Access
COP	Centers of Pressure
CoM	Center of mass
FDMA	Frequency Division Multiple Access
FSR	Force Sensitive Resistor
GPIO	General-purpose input/output
GRF	Ground Reaction Force
GPS	Global Position System
HF	High Frequency
ILM	Iterative Landweber Method
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
IPNS	International Conference on Information Processing in Sensor Networks
LPF	Low Pass-Filter
MT	Mobile Terminal
NFC	Near Field Communication
NN	Neural Network
PCoMA	Parallel Center of Mass algorithm
PIL	Precise Indoor Location
PFT	Polymer Thick Film
PNS	Pedestrian Navigation Systems
POF	Plastic Optical Fiber
RFID	Radio-Frequency Identification
RF	Reference Points
RPi	Raspberry Pi
RSS	Received Signal Strength
SNR	Signal-to-Noise ratio
TDE	Time Delay Estimation
TDMA	Time Division Multiple Access
TDOA	Time Difference of Arrival
TOA	Time of Arrival
TOF	Time of Flight
TOI	Target-of-Interest
UHF	Ultra High Frequency
ULF-MC	Ultra Low-Frequency Magnetic Field Communication
WLAN	Wireless Local Area Network

Chapter 1

Introduction

1.1 Motivation and Context

The localization and tracking of users in a specific space has in recent years become a goal of computer science researchers. With the advent of smart environments, transparent user localization has become even more pressing purpose than before the rise of these paradigms. If a system or environment could transparently follow the movement of the user, it could customize its interface and behaviour to match the references, history, and context of that particular ambient to intervene in the case of necessity or danger.

Since all dead reckoning solutions are subject to cumulative errors, navigational aids are required in order to give accurate information not only to correct positional errors but also to calibrate dead reckoning algorithms.

When we talk about outdoor environments, GPS plays a dominant role in localization. On the other hand, GPS is inefficient for indoor scenarios due to the weakness of signals emitted by this system and their disability to penetrate most building materials. According to the U.S. Environmental Protection Agency (EPA):

> "People spend approximately 93% of their time indoors, 2% outdoors, and 5% in transit (e.g car, train, bus). Thus, from a time budget standpoint, indoor environments dominate the total exposure spectrum." [1]

Considering that people spend most of their time in indoor environments, other effective technologies are demanded for indoor human/object location and some techniques have been developed these last decades. However, due to restrictions of position timing, position accuracy, and complex indoor environments, a flawless positioning technique has not been achieved yet.

In our particular case, the main issue is associated with estimation errors from an inertial navigation system. At the moment, Portuguese researchers from *Fraunhofer AICOS* have already developed a location system that using advanced dead reckoning algorithms based on fused data provided by an Inertial Measurement Unit (IMU) [2] is capable of calculate the user's position in indoor environments. Since this approach creates undesirable estimation errors, our mission is to develop a system that provides absolute reference points with increased resolution.

1.2 Project Presentation

> "Indoor location systems are an important enabling technology for applications such as indoor navigation, public safety, security management and ambient intelligence, representing huge potential regarding advertisement and retail businesses. Pedestrian navigation systems (PNS) have recently emerged as a solution for the indoor positioning problem regarding the lack of accuracy. These systems rely on dead reckoning algorithms which are solutions based on the fused data provided by the Inertial Measurement Unit (IMU) on the smartphone that can then be used to evaluate one's current position by using a previously known one."[3]

The lack of reliable GPS signals inside buildings has been an obstacle to determine accurate indoor positioning. In order to overcome this issue Fraunhofer AICOS has developed the PIL project. The Precise Indoor Location (PIL) is an accurate indoor location technology with sub-meter level accuracy that allows indoor navigation using a smartphone. The goal of the PIL project is to provide a commercial grade location solution, that is not only free of any infrastructure requirements, but also does not require efforts to generate and maintain maps of buildings for the purpose of indoor location, in order to contribute to the fast spread of indoor location solutions on the consumer level. Target markets are e.g. retail, solutions for public and private safety providers or elderly people.

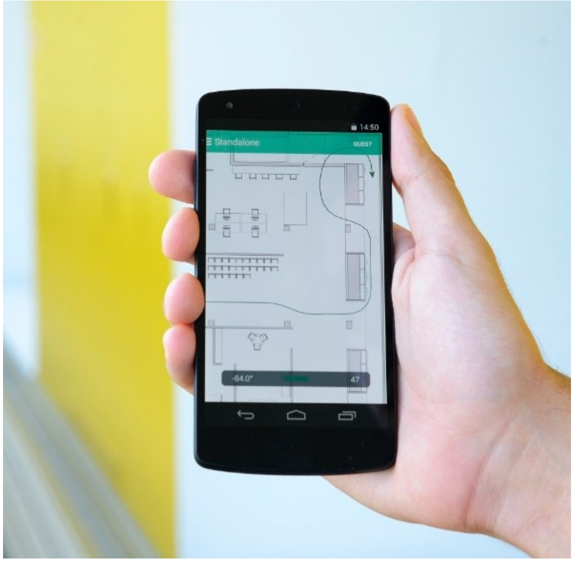


Figure 1.1: Precise Indoor Location (PIL) developed by Fraunhofer AICOS Portugal [3].

Recently Fraunhofer Portugal won the 3rd place with PIL solution in the Microsoft Indoor Location Competition [4], which was held last April 15 during the IPSN 2015 Conference in Seattle. With more than 40 submissions from industry and academia this is currently the most renowned competition in the field.

In Chapter 2 will be presented different approaches that due to their potential could be used for position reference. However, we opted for a solution based in a smart floor with pressure sensors. The smart floor was designed concerning the existing PIL project. Since the project already have a performance above the average, it was designed a solution that will work as an add-on for the main system and so the performance won't decrease. Taking into account these aspects, in Figure 1.2 is presented a block diagram describing the system concept:

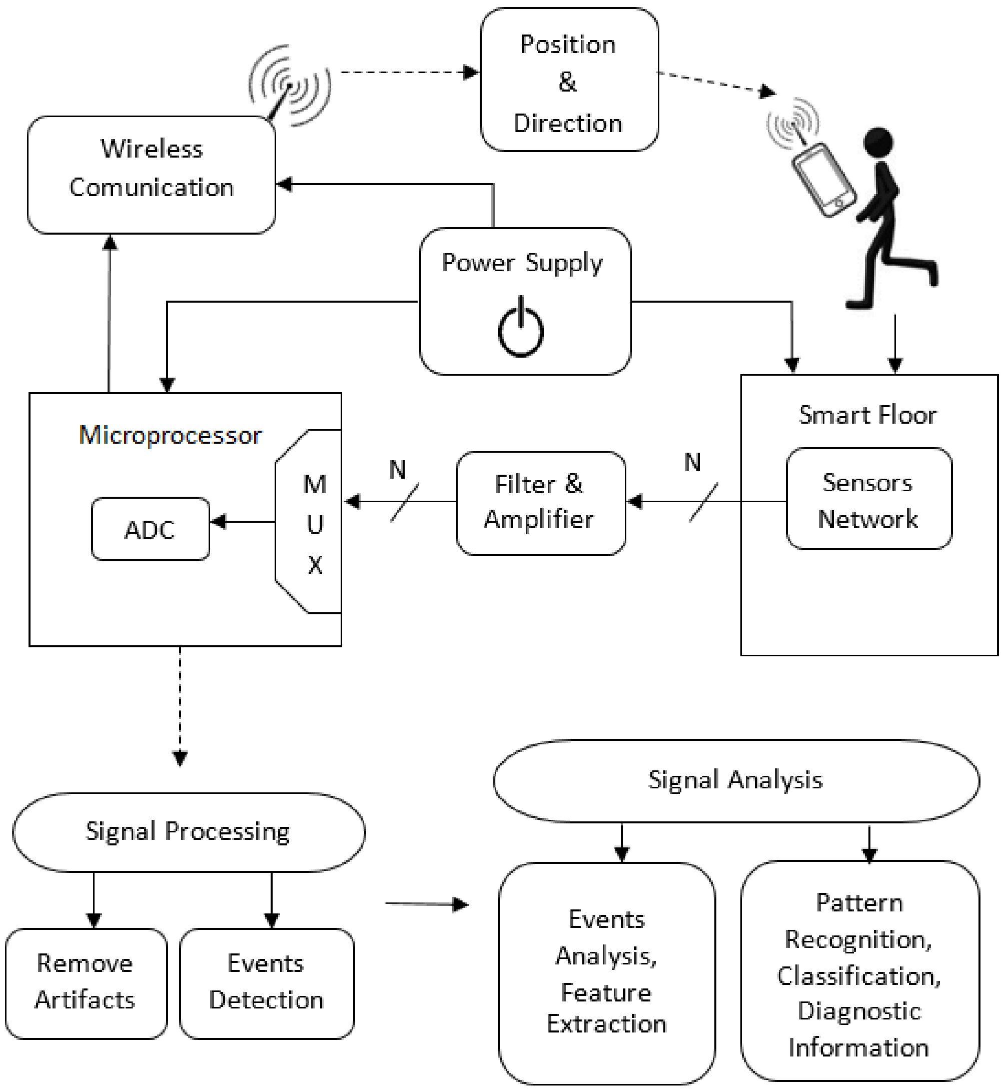


Figure 1.2: Floor Sensing System Concept.

1.3 Objectives

The goal of this dissertation is to develop a new solution that will work as a position reference for the main system. Concerning the main goal, we propose a floor sensing system. Basically, our solution consists in a hardware prototype with built in sensors that analyses the user's footprint and estimates the walking direction. Regarding these objectives, the main requirements of the system are:

1. **Low-cost solution**. Since our solution will work as a position reference, in a real indoor environment we will need a few prototypes distributed along the building. For this reason, the total cost of our prototype should be less than 100€;

2. **Compatible with Android environment** since Fraunhofer AICOS implements the PIL solution based on this Operative System;

3. **High accuracy** for position reference. The higher the accuracy, the better the system will perform. However, we think that sub-one-meter accuracy should be enough;

4. Estimate the **walking direction**. Considering that the gyroscope from the smart phone introduces drafting errors we need a solution that improves these undesirable estimation errors;

5. **Detect and correctly match the data**. The system need to be prepared to work in situations where multiple users are simultaneously using the smart floor;

6. **Transfer data** to the corresponding smartphone via wireless. Afterwards acquiring all the data, the system must send this information to the smartphone;

7. **Low energy consumption.** Since a considerable number of these prototypes will be required to cover a wide area, it is in our best interest to have a low energy consumption;

8. **Real-time solution.** All the information needs to be processed in real-time in order to immediately be available for PIL system.

1.4 Structure

In addition to the Introduction this report have 4 more chapters. In Chapter 2 we describe the State Of The Art and the research effort regarding Indoor Location Systems. In Chapter 3 we illustrate our approach to achieve the goals described in this chapter. The Tests and Results are presented in Chapter 4. Finally, Chapter 5 presents the major conclusions and also the future work.

Chapter 2

Background and State of the Art

In this chapter we will present the state of the art on indoor location systems. Although our approach will be based on floor sensing, we also start by introducing some remarkable techniques and point out a few recent applications.

For all the different techniques presented, we will present brief research in order to reveal if it fulfils the requirements of the project. The smart floor solution is suitable for most of the objectives illustrated in section 1.3 and for that reason we conclude that a floor sensing system should be implemented. To support our choice we expose some examples of smart floors that were successfully applied to foot detection and tracking.

At the end we will discuss the major topics of floor sensing application and in section 2.9 we will summarize all the background and research described in this chapter. We will also present some current solutions for similar problems and enumerate the fields of study associated to our final solution (which is based on force sensing principles).

2.1 IEEE 802.11 Wi-Fi

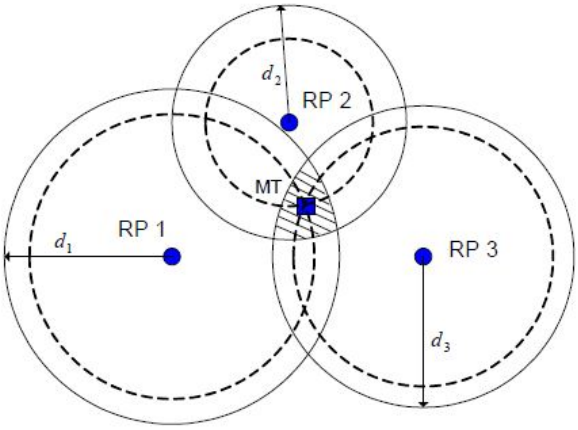
Wireless Internet has become the norm over the last years thanks to the IEEE 802.11 Wi-Fi standards and since wireless Internet has increased in use, so has the number of base stations (also known as access points or APs). Given that most smartphones have built-in support for Wi-Fi wireless Internet and once we are surrounded by Wi-Fi signals most of the time, it seems like using Wi-Fi as a position reference would be an acceptable choice.

There are two fundamental methods used in indoor location determination with Wi-Fi signals: TOA – time of arrival and AOA – angle of arrival. There also the RSS (Received Signal Strength) based techniques which are applied to estimate the position of the target-of-interest.

The explanation of both techniques can be found in the paper [5]. Nonetheless, the findings from [6] illustrate how difficult it is to accurately measure the AOA, RSS and carrier signal phase in indoor radio propagation channels. For this reason, most of the independent indoor position systems mainly use the TOA-based techniques.

As a result of estimation errors of distance to reference points (RP) receivers, caused by an inaccurate TOA measurement, the geometrical trilateration technique can only provide a region of uncertainly, for the estimated location of the mobile terminal (MT), instead of a single set of position coordinates.

Figure 2.1: Distance-based geolocation method.[5].



2.2 Bluetooth

As most mobile devices support the Bluetooth wireless standard, location determination methods using this technology can be considered complementary to using Wi-Fi.

Although Bluetooth requires 5 to 10 seconds to discover local Bluetooth devices and has the fastest data throughput speeds only at approximately 0.7 Mbps [7], Bluetooth has an added advantage of being a short range radio technology. With typical devices having a range of 30 feet, this provides the potential for using these devices as low cost radio beacons that give a more precise location by covering smaller regions than Wi-Fi APs.

An example that combines both Wi-Fi and Bluetooth technologies can be found in [8]. As we can see in figure 2.1, the final results confirm the concept idea.

Table 2.1: Data of Experimental Results from [8].

Type	**Average Error (m)**	**Maximum Error (m)**	**Minimum Error (m)**
Without Global Searching	4.0140	10.8	0
WI-FI Only	3.0380	9.4	0
WI-FI & Bluetooth	2.9148	8.9	0

2.3 InfraRed

Starting in 1992, we can find examples of location determination using InfraRed light, known to be used in television remote controls. *The Active Badge Location System* [9] estimates a location based on an Active Badges that would be worn by the participants. Basically, this badge would broadcast a unique code every 15 seconds using InfraRed light and this burst of invisible light would be received by sensors strategically placed. Approaches based on InfraRed have some advantages compared to techniques depended on Wi-Fi and Bluetooth technologies. In Table 2.2 we have a summary around these 3 approaches:

Table 2.2: A comparison of wireless technologies [10].

	Area of interaction	**Throughput**	**Time of connection**
WLAN	Range of 100 m around the access point (in indoor environments without obstacles)	Theoretically 11 Mbps but in practise closer to 4-5 Mbps	Immediately
Bluetooth	Range of 10 m around another Bluetooth device	Theoretically 1 Mbps but in practise closer to 700 Kbps	Between 5 and 10 s to discover other Bluetooth devices
Infrared	Sender and receiver should have the corresponding ports tilted at a 30ngle each other, point and shot style application	Currently, according to the protocol used by devices, up to 4 Mbps	Immediately after two devices are lined up

We could adapt our system in order to use the smartphone's camera to work as a receiver of the invisible light. Although this is an inexpensive procedure we consider that it is not convenient for the user to take out his phone, turn on the camera and search for the transmitter. The aim of our solution is to have the minimum physical (and psychological) disturbance to our daily life and for that reason the sensing devices should be unnoticeable and totally autonomous.

2.4 RFID

RFID (Radio Frequency Identification) is expected to be a good way of achieving position based services economically indoors. The normal procedure is to measure the radio power of the signals from three RFID readers, similar to GPS, and to calculate position. This can be accurate but on the other hand it is quite complex.

In [11], a novel indoor location estimation technique based on UHF band RFID is proposed. The person's location is estimated through the radio field strength of the tag signals received, as shown in Figure 2.2. In this technique, 950 MHz band passive system is used since it offers communication over several meters. This approach is far more cost effective than other techniques because the passive tags do not need any battery which greatly improves serviceability.

Figure 2.2 shows the system outline. Each tag has a unique ID number. Tag IDs of read tags are transferred to the server. The server has a location information mapping table that pairs each set of tags with coordinates.

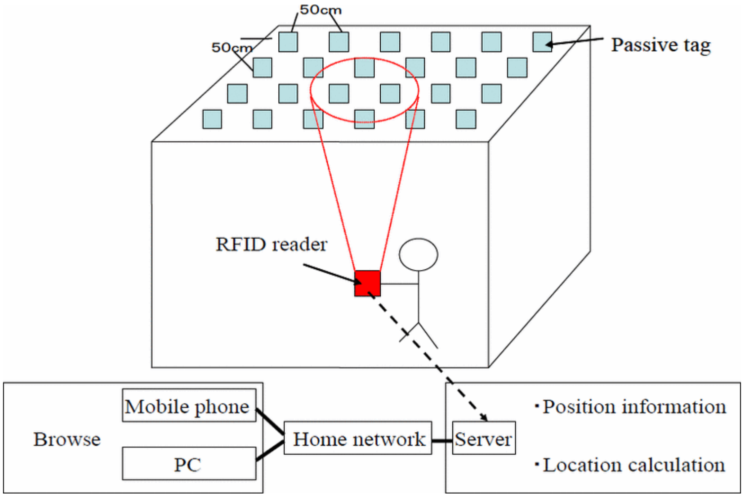


Figure 2.2: RFID System outline [11].

2.5 Ultrasonic

Another potential source of location data is ultrasonic sound. In *The Cricket Indoor Location System* [12], ultrasonic beacons are placed strategically throughout a building. The system is a decentralized one, with the beacons communicating with each other to determine their distance from each other by measuring the travel time of sound waves. Several such distance measurements from other beacons provide the device's overall position relative to the others. A mobile device can then discover its current position using the same method. Figure 2.3 describes the system.

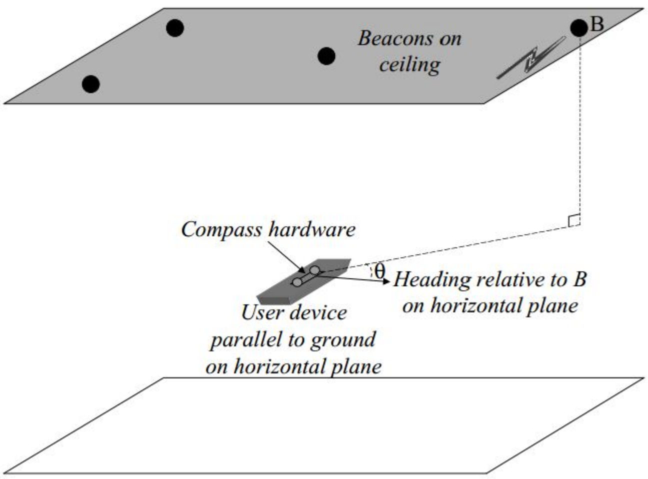


Figure 2.3: *The Cricket Indoor Location System* Concept [12].

The idea is quite ingenious as it provides a system that requires minimal setup and maintenance while being fault tolerant. However, the authors consider that the results raise some important issues that need to be addressed in the implementation phase before a production system can be realized. Some of those improvements are related to problems with reflections and diffractions.

Even this approach has accurately measures (in the last experiment the authors improved precision up to 5 cm) it isn't a practical solution because once again the system needs user's intervention.

In the previous sections of this chapter, we illustrated some applicable solutions for indoor location systems. Even though most of them are very accurate, there are always disadvantages associated to those solutions and for that reason we think that none of the approaches reported above will completely satisfy our aspirations. In Table 2.3 a comparison is made between some major localization systems considering their accuracy, advantages and disadvantages, networking technologies and localization methods.

Table 2.3: Localization Systems [5].

System	Network	Accuracy	Method	Overall Evaluation (A: Advantage; D: Disadvantage)
WhereNet [13]	RFID	2 m to 3 m	TDOA	A: Uniquely identify equipment and person. D: Need numerous infrastructure components
RADAR [14]	WLAN	2.26 mout of 312 m^2	Triangulation	A: Reuse the existing WLAN infrastructure. D: Low level accuracy, no consideration of privacy
Active Badge [9]	Infrared	Room level	RSS	A: Address privacy D: Low accuracy; long transmission period; influenced by fluorescent light and sunlight
Cricket [15]	Ultrasound, RF	10 cm	TOA and triangulation	A: Address privacy; low cost, decentralized administration. D: More energy consumption

Looking at this table we immediately realize that a better approach that fulfils all the requirements that we propose in section 1.3 is needed.

2.6 Sound Based Indoor Localization

In the research of an audible sound based indoor localization [16], the authors believe that once a fixed indoor environment has many spaces providing public address sound system, it is possible to design a localization system. In order to do that, the moving person should be carrying on a sound receiver with wireless transmitter and enough indoors signal coverage. To work as a sound receiver the person could use his/her own cell phone. However, there are inherent problems with using dedicated proprietary tools in this process:

1. Adaptation of a public address sound system to allow the simultaneous separated excitation of loudspeakers so that TOF (time of flight) and therefore distance may be estimated;

2. Simultaneous access and multiple users;

3. Data hiding in sound so that only reasonably small disturbance in the acoustical environment may occur;

4. How may a simple audio channel, like the one of a common cell phone be used as an acoustic receiver localizer;

Even these issues raise the sound based indoor localization system's problems, the authors were able to develop a similar approach. The system is named NAVMETRO and was already successfully tested in Trindade Metro Station (Porto, Portugal). The main goal of this system was to help visually impaired navigating inside the Metro Station.

This paper was focused on theoretical and practical aspects of a possible real implementation of an audible sound based indoor localization scheme using a standard audio channel receiver. Experimental results in a room with 6 m x 7 m x 3 m size using TDMA, FDMA and CDMA access schemes in a sound communication system, show that is possible to have a localization accuracy of a few centimetres in non-ideal conditions. To maximize the general performance of the system (e.g. accuracy, precision, ...) the authors conducted three experiments:

1. Latency analysis considering the Easera Gateway sound board with different API tools;

2. Comparison on three correlation techniques to perform Time Delay Estimation (TDE): cross-correlation, generalized cross-correlation phase transform and maximum likelihood;

3. Evaluation of the position estimation error and reliability with a sufficient SNR considering: TDMA (with unit pulses), FDMA (with chirps) and CDMA.

The three experiments showed promising results as we can see in table 2.4:

Table 2.4: Results from [16].

	TDMA	FDMA	CDMA
Average error in all area (cm)	5,3	5,4	4,5
Average error inside centre area (cm)	1,5	3,0	1,3
Average error outside centre area (cm)	6,4	6,0	5,4
Reliability inside centre area (%)	100,0	100,0	100,0
Reliability outside centre area (%)	98,3	98,9	97,8
Minimum SNR(dB)	24,7	11,4	7,2

2.7 Floor Sensing System

In this section we will explain with some detail several location systems based on floor sensing. For our final application we believe that it is essential to investigate methods for measuring footsteps. If we could achieve a detailed user's footprint measurement, we would not only have an accurate measure of the position and direction, but we would be able to solve the problem of matching those footprints to the right user 1.3. We believe that only with a complete floor sensing system we will be capable of correctly identify the user in a situation where multiple individuals are simultaneously using the system. Consequently, we believe that our system must provide, among others, the following elements:

1. Impact Time;
2. Time on Heel to Toe;
3. GRF - Ground Reaction Force;

There are already solutions commercially available. An example of those solutions is the electronic walkway under the brand name GAITRite [17]. The system consists in sensor pads, each containing 2304 sensors arranged in a 48 x 48 grids, typically covering a total active area of 1 m x 6 m. It is capable of measuring spatial and temporal gait parameters and it's portable. The GAITRite has been used by physiotherapists as part of clinical assessment, but its cost is substantial. This limits its usage to a laboratory tool rather than an affordable system potentially deployable in new homes or retro-fitted in existing dwellings.

In order to achieve a lower cost solution, a number of methods for footstep sensing have been proposed and developed including, for example: piezoelectric 2.7.1, force 2.7.2, resistive 2.7.3 and plastic optical fibre 2.7.4 sensing principles.

2.7.1 *The Magic Carpet*

This system employed a piezoelectric cable to produce a sensing floor of size 3 m x 8 m with a resolution of 10 cm [18]. The system had a scan rate of approximately 60 Hz.The authors created it in order to be used as an installation art piece where a person's motion controlled music.

One of the interactive environments conceived for this project involved creating a space where the position and pressure of a performer's feet would be measured together with upper-body and hand motion using a pair of Doppler radars. This data would be used to create a truly immersing environment, where any kind of body motion would be directly and immediately converted into expressive sound. However, although the system had a good performance in tracking an individual among the surface, it does not provide enough information for gait analysis.

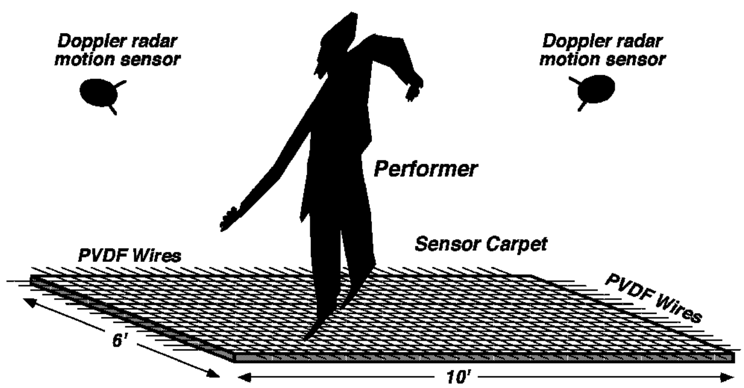


Figure 2.4: Arrangement of the prototype *Active Floor* [18].

Learning from this experience the same group later developed the *Ztile* which had hexagonal tiles (with 40 cm diameter) containing 20 sensors each. *Z-tiles* was a design for a fully scalable, self-organising, force sensitive surface that detects x and y locations as well as the force applied (z-axis). The main goal of this system was to develop a fully pixelated surface area that could detect both force and location in real-time situations.

2.7.2 *The ORL Active Floor*

The ORL Active Floor [19] is a square grid of conventional carpet tiles, each backed by 18 mm plywood and 3 mm steel plate, supported at the corners by cylindrical load cells which are instrumented to give us the total vertical force. In the data acquisition system described the authors have found that the load cells are able to resolve weight changes of about 50 grammes. The grid has a 50 cm spacing, and a sampling rate of around 250 Hz per load cell is employed. This approach uses hidden Markov model technique in order to attempt the classification of the footstep signature of a number of individuals. The main idea of the system is to allow the time varying spatial weight distribution of the active office environment to be captured.

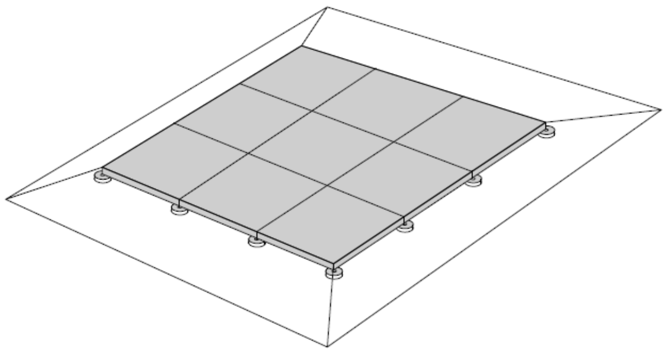


Figure 2.5: Sensor Arrangement for the Carpet System [19].

However, this system had some disadvantages with pressure sensitive technology (such as that from *Tekscan*):

1. Some weight is supported by the areas between the pressure sensitive pads, so the total weight on an area of floor cannot be accurately ascertained;

2. High point pressures will damage the small sensors, e.g. high heels. The available technology is not recommended for sustained use even with flat shoes;

3. The technology is expensive when applied to a whole office floor (more or less $3000 per tile).

For these reasons, the authors from *The ORL Active Floor* opted for the load cell approach. On the other hand, they lost the opportunity of processing image and concerning to that they weren't able to see, for example, which way feet were pointing.

2.7.3 *A floor sensing system for gait recognition*

This subsection will describe a prototype floor sensor as a gait recognition system [20]. In 2005 a novel approach emerged in order to measure gait accurately without the need of video analysis. The new sensor consists of 1536 individual sensors arranged in a 3 m by 0.5 m rectangular strip with an individual sensor area of 3 cm^2 and operates at a sample rate of 22 Hz.

This solution is inspired by computer keyboards design and is made from low cost, off the shelf materials (sensors themselves cost around $100). This application is based on the extraction of all the elements referred in 2.7 and the final results proved that the floor sensing system was able to achieve an 80% recognition rate. In the Figure 2.6 we can see the final prototype.

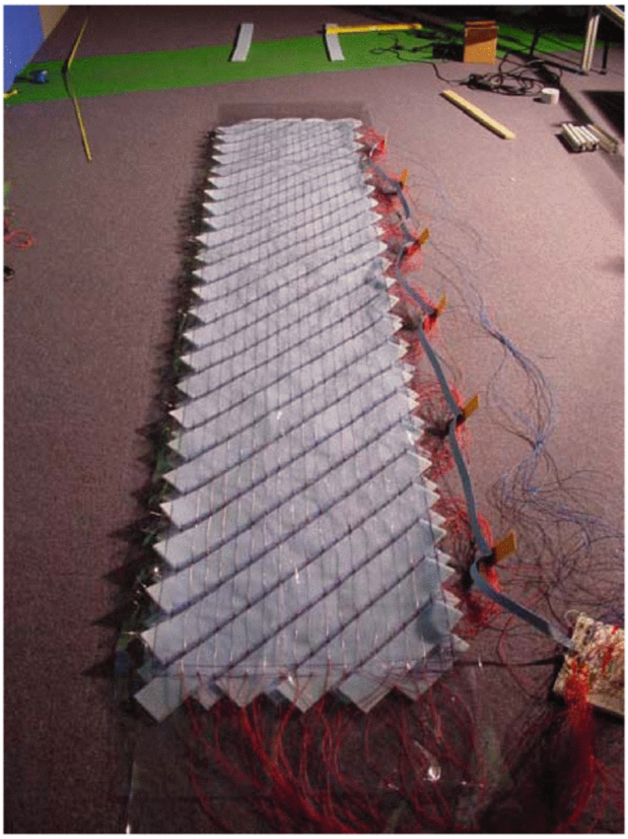


Figure 2.6: The final prototype sensor mat for gait recognition [20].

In this gait recognition system the authors found a problem known as ghosting. In ghosting a triangle of three points, when pressed simultaneously, will also illicit an erroneous fourth point. This occurs because the current can flow through multiple ways to the ground. Sometimes we solve this problem by placing diodes at each switch to stop current flowing backwards but since we have 1536 sensors we can't do that. In order to solve this issue the team placed an insulation along diagonal lines and overlap 4 grids. 2 layers of the sensor mat were also implemented so that the resolution could be increased.

In this paper the authors also explain the influence of sensor resolution on foot profile. Figure 2.7 describes this effect.

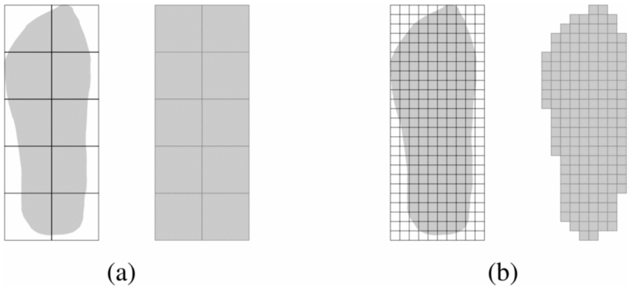


Figure 2.7: (a) low resolution - 1 cm^2; (b) high resolution - 5 cm^2 [20].

The analysis of the profile of all footsteps is needed in order to give a correct gait analysis:

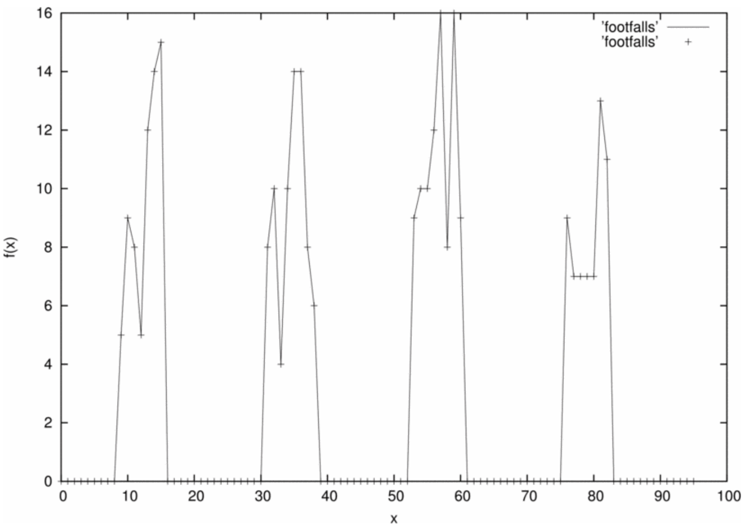


Figure 2.8: The profile of the 4 footsteps on the sensor mat [20].

2.7.4 Footstep imaging using Plastic Optical Fibre (POF)

The cost of sensors, reliability of the sensing signal, resolution of the sensing floor, sampling rate, and sensing area are all key factors that influence the performance of the intelligent environment. In order to get a better solution based on these requirements a new technique based on optical sensing has emerged.

The POF sensor utilizes light intensity transmission measurements and compared to other sensing methods offers the advantages of ruggedness, intrinsic safety and resistance to fluids. Typically, POF is available at the cost of less than 1 USD/m and can be combined with inexpensive, light and small optoelectronic light sources and detectors for the manufacture of energy efficient sensor elements. Furthermore, it is straightforward to integrate the POF sensor elements with a standard commercial underlay commonly used for carpeting, allowing the sensor head to remain inconspicuous under the normal carpet surface in the daily living space.

An example of footstep imaging using POF is the new *Intelligent Carpet System* [21] (2015). The authors of this novel approach manufactured an 1 m x 2 m sensor head by attaching 80 POF sensors on a standard commercial carpet underlay and it only cost 150 USD/m^2;

Since the POF sensitivity to bending is still weak in absolute values, several authors have demonstrated methods to enhance their sensitivity by embedding an imperfection in fiber, which modulates its wave-guiding properties as a function of the bending radius. As the authors of the *Intelligent Carpet System* illustrate, the simplest and least expensive way to enhance the POF sensitivity to bending is to manufacture grooves of suitable depth and period along the POF length. A few experiments were done in order to properly discover the dimensions of the grooves. We can found those experiments in the paper, however we prefer to show the final performance results.

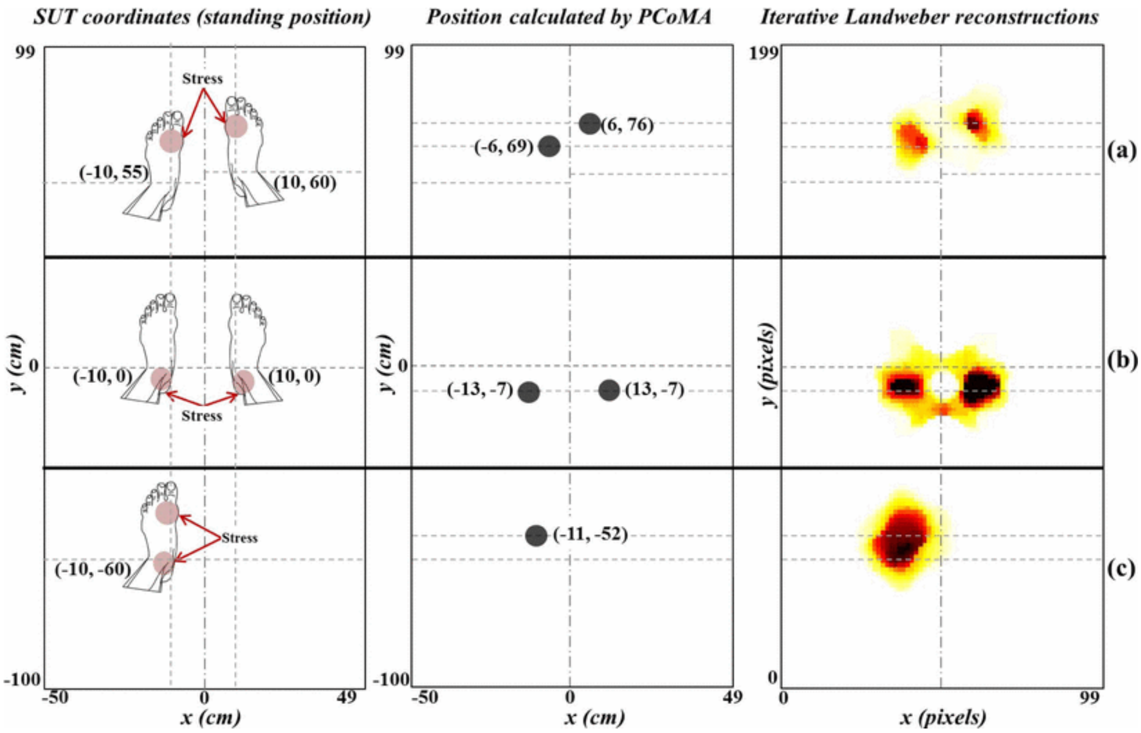


Figure 2.9: Footstep imaging from *Intelligent Carpet System* [21].

Figure 2.9 presents three different bare feet positions: (a) the weight is on the metatarsus bones; (b) the weight is on the calcaneus bones; and (c) the weight is balanced on the left foot.

> "Compared to the actual feet coordinates, the calculated CoM (center of mass) coordinates show for both feet a clear displacement of as much as 7 cm towards the back of the feet and about 3 cm outwards. In case (c), in order to keep the balance, stress is naturally distributed on both, metatarsal and calcaneus bones. Furthermore, the pressure is doubled, resulting in stronger deformation affecting areas spread further, as observed by ILM. The centre of deformation is shifted with about 8 cm towards the front of the foot, which corroborates that the pressure is more evenly distributed between the heel and the ball of the foot, compared to the other cases. There is generally good consistency in the PCoMA calculations and the ILM reconstructions."

So far this approach seems to be perfect for our problem, however we think we should discard this technique due two main reasons: first, the methods and algorithms that were used are quite complex, so the external expensive hardware needed to process all the information, raises substantially the cost of the system. Furthermore, in *Intelligent Carpet System*, a data scan of the complete frame takes 328 ms and considering the duration of the gait events, the time resolution or image frame rate of the system required has to be at least 10 Hz (100 ms imaging period). As we can see, in result of these two problems, a system based on POF is unable to reach requirement number 8.

2.8 Current Solutions

During this research, we found a few approaches using force sensing principles that partially solved the main issues presented on this dissertation.

Hyunseok Kim and Seongju Chang (2015) developed a touch floor system based on force sensitive resistors, capable of identifying user position and motion with high resolution. They applied a particle swarm optimization-based neural network (NN) initialized with the output of a Levenberg–Marquardt-based NN. This technique produced inaccuracy drawbacks of the trilateration method in position estimation due to sensor's non linearity to be reduced. In short, their system was capable of providing similar functionalities and almost the same resolution as a touch pad or pointing device such as a mouse [22].

A. Nunes et al. (2007) proposed an architecture of a pressure sensing floor divided in rigid tiles. The system was based on a network of flexible pad pressure sensors, used under all tile corners. The proposed architecture was applied in an interactive room with a 64 tiles floor (256 sensors), providing a network weight measuring system that allows detecting, recording and tracking the movement of objects or people over the sensitive area. Their system was organized as a wired network of modular acquisition and computational units that communicate wireless with a computer [23].

Gang Qian et al. (2010) presented an approach for people identification based on gait using floor pressure data. By using an high-resolution pressure sensing floor they were able to obtain both the 1D pressure profile and 2D position trajectories of the centers of pressure (COP) of both feet to form a 3D COP trajectories over a footstep. From the 3D COP trajectories of a pair of footsteps, a set of features were extracted and used together with other features such as the mean pressure and stride length for people identification. In the end, their method reached an average recognition rate of 92.3% [24].

Seongju Chang et al. (2010) implemented an interactive floor system named Ubi-floor which allows users to interact with a floor based smart environment. They focus on developing a modular smart floor which could be applied to the real built environment to provide various interactive multimedia services. The location detection algorithm is based on referencing and solving equations of relative voltage strengths measured through multiple pressure sensors. The application of this technique enabled the identification of user's position in a relatively large surface [25].

2.9 Summary

Along this chapter, research in the fields of biomechanics, wireless communication and floor sensing was revealed.

Solutions based on wireless communications such as Wi-Fi, Bluetooth, RFID, Ultrasonic and Infra-Red are mainly used as techniques for indoor navigation. However, we described in this chapter a few disadvantages regarding to these approaches. This type of solution typically isn't capable of detecting position with sub meter accuracy or, in some cases, they required additional devices beyond the smartphone. Figure 2.10 describes a survey of wireless indoor positioning techniques and systems.

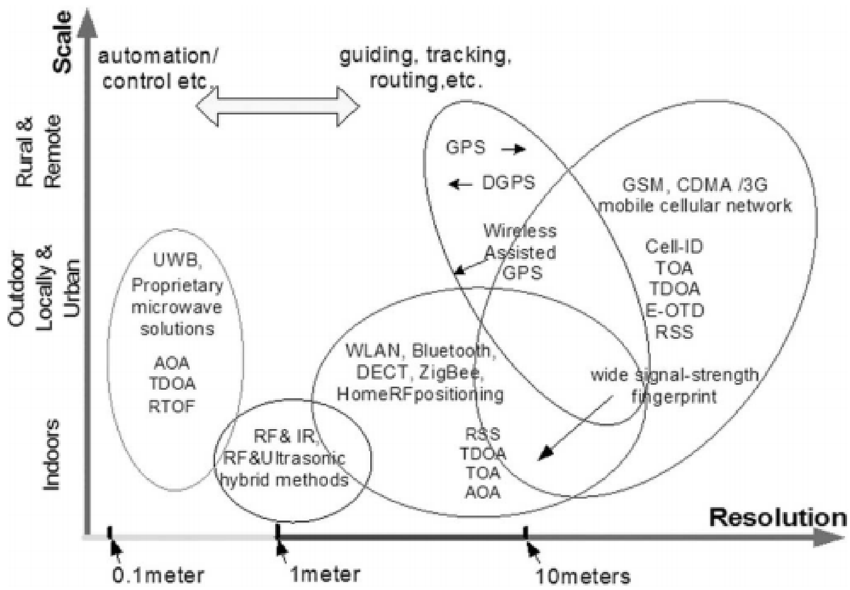


Figure 2.10: Current wireless-based positioning systems [26].

During the research of techniques for solving the requirements proposed on this dissertation, we realized that a solution based on floor sensing had more advantages than using wireless techniques.

Considering the fields directly associated with the study of the dynamics of human gait, it was important to research and study the models and parameters involved in gait analysis. Achieve knowledge in themes such as normal and pathological function of gait analysis was fundamental since our solution is based on recognition and identification of footsteps.

Chapter 3

System Development

In Chapter 1 we mentioned all the objectives of this dissertation, however we can resume our work in two main goals:

1. Determine the absolute user's **position** (less than 1 m precision);
2. Estimate the user's **orientation and direction**;

Taking into account the background and the state of the art described in the previous chapter, and considering the constraints in which this project was developed, the proposed solution follows the topics bellow:

- To determine the absolute user's position it was installed sensors on the smart floor which are activated when a person walks through the tile;
- The sensors used in our prototype are **force sensitive resistors**;
- The smart floor is size of 60 cm x 60 cm in order to match the standard dimensions of a raised floor tile;
- Our solution includes **4 sensors**, each one placed at the corners of the tile;
- The output of the sensors are **linearised** so we can get reliable data;
- The new **Raspberry Pi 2** is used to process all the data received from the sensors;
- To estimate the user's orientation and direction we implemented an approach based on **Fuzzy Logic**.

3.1 Hardware

Along this section we will explain and describe with some detail all the electric components used to build the prototype. A final budget will also be presented at the end. Since the beginning, one of the main goals was the final cost of the project which must be the lowest possible.

According to [21] the walk period of an average human is typically 1.2 s, which is double to duration of a single step full cycle. Assuming that the touchdown time of the foot is around 0.2 s , the time resolution of the system for simple gait recognition must be at least 10 Hz (100 ms imaging period). For this reason, the frame rate is an important factor that was considered before buying the material. Another requirements that deserve special attention are the power consumption and also the price (should be as cheap as possible).

3.1.1 Force Sensitive Resistors

The sensors network consists in **FSR 402** sensors which are a robust polymer thick film (PTF) devices that exhibit a decrease in resistance with increase in force applied to the surface of the sensor [27]. Unfortunately, the sensor has a non-linear behaviour when force is applied. In subsection 3.1.2 it can be found a brief explanation on how this problem was solved:

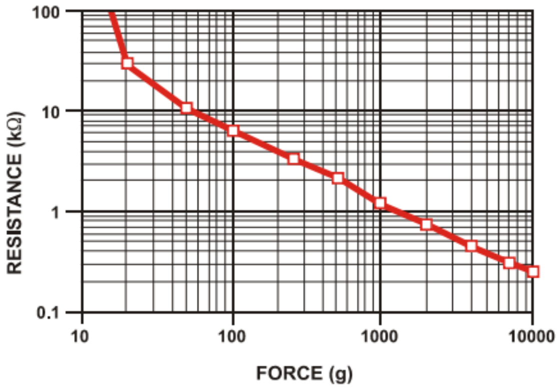


Figure 3.1: Force vs. Resistance [27]. Non-linearity is visible despite the logarithmic scale.

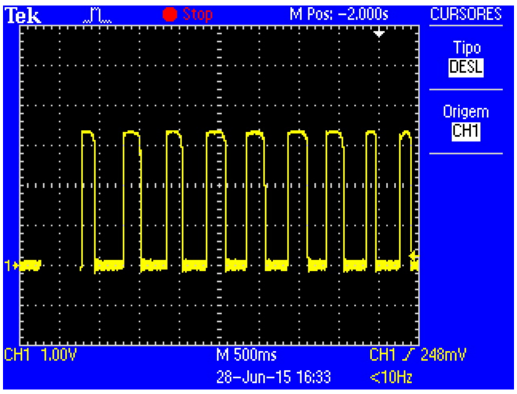


Figure 3.2: FSR output when slighty pressed. Without any type of signal conditioning, the output immediately saturates at Vcc voltage. It works as a switch (HIGH or LOW).

The FSR 402 is produced by *Interlink Electronics* and it's a round sensor 13 mm in diameter (active area) and its thickness range goes from 0.2 mm to 1.25 mm. As we can see, the small size of the sensor allows its installation under the floor without being detected. In Figure 3.3 we can see how minuscule a FSR402 sensor could be.

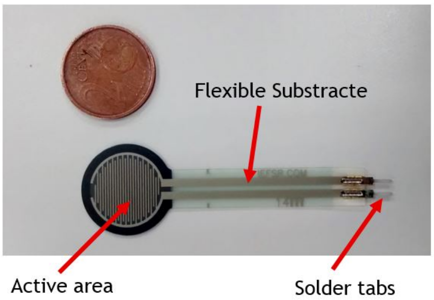


Figure 3.3: Comparing FSR 402 to a 0.02€coin.

The manufacturer ensures a lifetime of 10 Million actuations (1 kg, 4 Hz). The device also has a rise time less than 3 microseconds so it could perfectly be used in a real-time scenario as Yanbo Tao et al. prove, by applying the same sensors to a real-time intelligent shoe system for fall detection [28].

The unit price of each sensor is less than 7€but this value decreases when the sensors are bought in larger amounts (e.g for a package of 25 sensors the unit price is less than 5€) [29]. The FSR 402 sensor works in pressure sensitive ranges from $0.1\,\mathrm{kg/cm^2}$ to $10\,\mathrm{kg/cm^2}$ but the manufacture affirms that the force range can be increased to operating forces larger than 50 kg.

At the same time we were researching for a sensor that fits the requirements of our project, we found others force sensitive resistors that also could be applied to our prototype. Nowadays, most of these sensors are used for medical purposes, such as the ones that are used in GAITRite (section 2.7.3). Despite there are a lot of valuable sensors, only a few of those sensors were considered taking into account its price.

An example of an alternative sensor for our system is the FlexiForce A201 sensor made by *Tekscan* [30] which is physically very similar to FSR 402 sensor, as we can see in figure 3.4.

Figure 3.4: FlexiForce A201

Fabrizio Vecchi et al. have already made an experimental evaluation between these two commercial force sensitive resistors [31]. The objective of the evaluation was to make a comparative analysis and discover which one had better performance. Both sensors were submitted to several tests and at the end the authors conclude that the FlexiForce A201 sensors showed better performance in terms of repeatability, linearity, time drift (when mounted on a rigid substrate) and as well as in terms of dynamic accuracy. Table 3.1 describes the final results of this evaluation.

Table 3.1: Comparison between the errors measured for the FlexiForce A201 sensor and the FSR 402 sensor [31].

	Force range (N)	Flexiforce sensor	FSR402 sensor
Average error (N)	0-20	2.7	4.7
	0-25	3.6	4.9
	0-30	4.6	5.9
Maximum error (N)	0-20	7.4	12.4
	0-25	9.9	13
	0-30	14.2	19.6
Standard Deviation error (N)	0-20	1.8	3.1
	0-25	2.4	3.7
	0-30	3.6	4.4

Afterwards we analysed the table above, we easily conclude that FlexiForce A201 had a better performance. However, there is an important condition that we need to take into account: the unit price of FlexiForce A201. Since these are sensors mainly produced to be used in fields of biomechanics and kinematics, the unit price can reach almost 58€. According to this value, the final cost of our prototype would be 7 times more expensive. Considering that we proposed a scalable modular architecture which would allow the development of cost effective pressure sensing applications, we opted for a cheaper solution like FSR 402 sensor. Besides that, we have already proved in this section that FSR 402 is a reliable sensor that achieves all the demands of our project.

3.1.2 Signal Conditioning Circuit

The datasheet of FSR 402 suggests a simple force-to-voltage converter. To do that, the FSR device should be tied to a measuring resistor in a voltage divider configuration. The measuring resistor is used in order to maximize the desired force sensitivity range and to limit current. Additionally, the manufacturer recommends that the voltage divider should be followed by an op-amp.

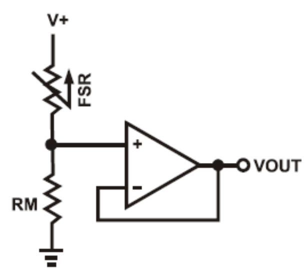


Figure 3.5: FSR Voltage Divider [27]

The force range can be extended by reducing the drive voltage, V+, or the resistance value of RM. The output of the circuit above is described by the equation:

$$Vout = V^{+} \times \frac{RM}{RM + RFSR} \tag{3.1}$$

In the previous section we mentioned that one of the problems of using the FSR 402 is its non-linear behaviour when pressure is applied. Nevertheless,the manufacturer point out a different circuit which should be used in situations where a linear response is desired. Instead of getting the voltage they considered to use a current-to-voltage circuit. As we can see in the figure 3.6, the FSR 402 has a more linear response when we are dealing with conductance.

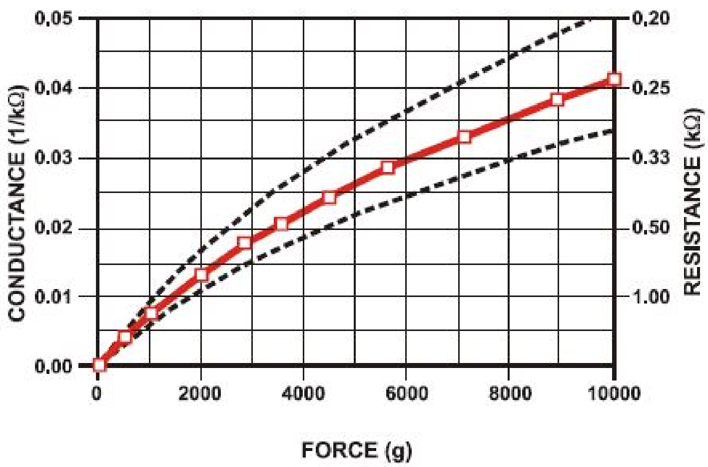


Figure 3.6: Conductance vs. Force (0 kg - 10 kg) [27].

At higher forces, the response eventually saturates to a point where increases in force yield little or no decrease in resistance. Looking at Figure 3.6 we find out that the saturation force is beyond 10 kg. To maintain the dynamic response, forces higher than the saturation force have to be measured by spreading the force over a greater area and therefore the overall pressure is kept below the saturation point. Further, we will see that this concept was successfully applied to our solution.

At this moment, and after some experimental evaluations with different approaches for signal conditioning (e.g Voltage follower, AC amplifier, Anti-log amplifier, etc.), we decided to follow the manufacturer suggestion and use a current to voltage converter. Our choice was based on the results obtained and at the end the current to voltage converter had the best performance in terms of linearity. Although, taking a closer look at Chapter 3 of "Sensor and Signal Conditioning" [32] we found that the author also recommends to use a current to voltage converter for non linearisation of force sensitive resistors. To conclude, the signal conditioning circuit, it was connected a Low Pass Filter to the output of the amplifier.

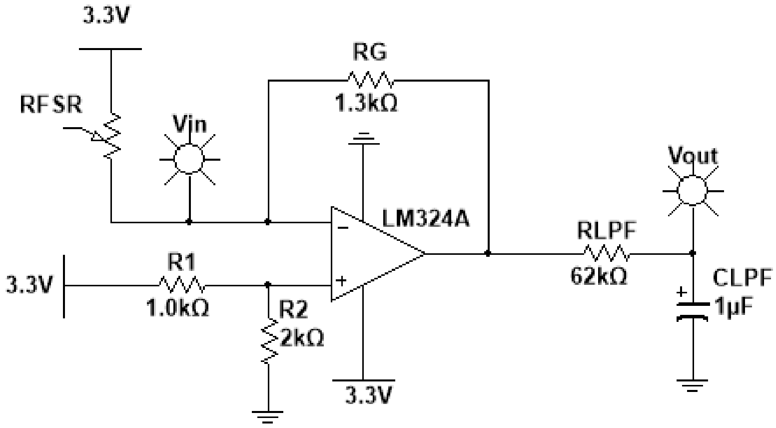


Figure 3.7: Signal Conditioning Circuit (designed with *Multisim 12.0*).

Once we chose to work with op-amp LM324 (low cost quad amplifier that accepts 3.3 V voltage supply) we needed to beware that its maximum output voltage is 1.8 V [33]. This restriction led us to an undesirable situation: when RG was greater than RFSR (higher pressure situations) the output (Vout) went into negative saturation. :

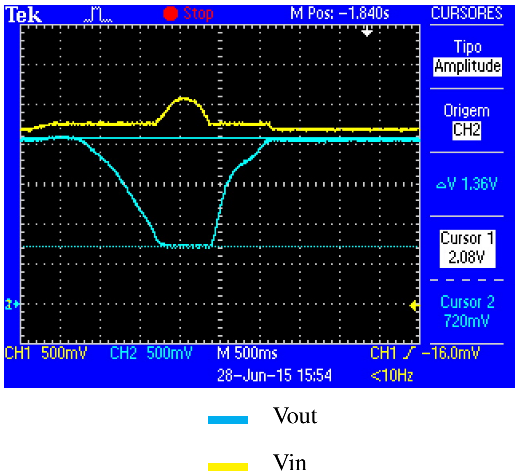
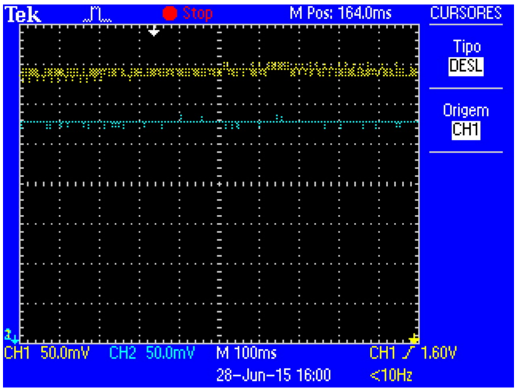


Figure 3.8: Negative saturation of the current-to-voltage converter.

In order to avoid this situation a few adjustments to the RG value were considered. We start by testing our prototype in the worst case scenario which is when someone presses the corner of the tile, right over the sensitive area of the FSR. We measured the resistor value of each sensor while a person was applying high forces into the sensitive area. The average value registered for a person with an average weight of 95 Kg was 1.5 KΩ. As a precaution, it was decided to use instead a 1.3 KΩ value for RG.

A Low Pass Filter (LPF) was implemented in order to reduce the noise of the output when no pressure was applied. As we can observe in the figure bellow, at initial conditions (Force = 0 N) the output was to noisy due the unstable value of the FSR resistor. Once the FSR resistor had a huge value when no pressure is detected and assuming that we want to get voltage values at the output, it’s expected to see the output signal stabilizing at 0 V (FSR works has opened circuit). Alternatively, with a current-to-voltage approach, the circuit is designed to read the current and therefore the output oscillates due the resistance’s instability (more than 10% according to the manufacturer). For that reason a LPF should be added to the final circuit. The values of RLPF resistor and CLPF capacitor were calculated in order to have a cut-off frequency of 2.5 Hz which was the one that revealed better results after a few experimental evaluations at several frequencies.

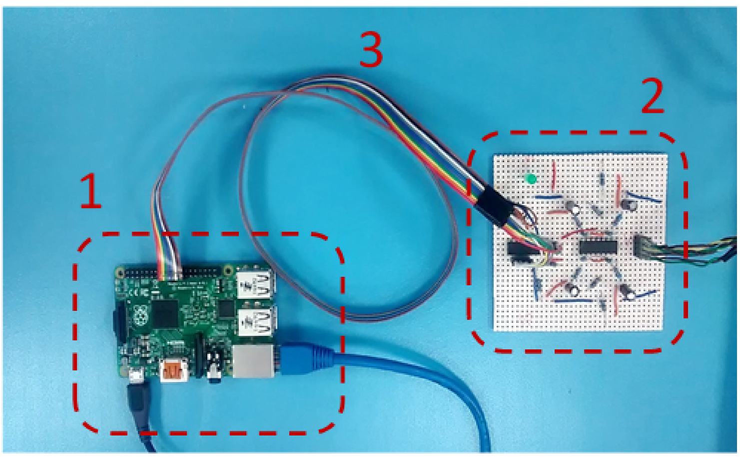


Vout

Vin

Figure 3.9: Results after applying the LPF. We were able to reduce 10% of the noise.

To conclude this section, in figure 3.10 we present the final physical appearance of our signal conditioning circuit.



1 - Microprocessor

2 - Signal Conditioning Circuit

3 - Communication Bus

Figure 3.10: Signal Conditioning Circuit (designed with *Multisim 12.0*).

3.1.3 Analogue to Digital Converter

At the beginning of this dissertation it was purposed a scalable solution that should allow an increase of the active area whenever it was needed. Regarding to this concern, it was chosen a microprocessor capable of accepting the higher number as possible of FSR sensors (section 3.1.4). Taking this into account, we implemented an ADC because it would allow using digital input rather than analogue input. Therefore, we didn't need to buy an expensive microprocessor with multiple analogue inputs. In other words, imagine if in the future we want to expand 6 times more the active area of our solution (to make an active area of 1.8 m x 1.2 m). In this situation we would use 24 analogue inputs which would be an expensive solution instead of using a few digital inputs. Considering our ADC, with a single digital input the system is able to read 8 sensors simultaneously.

The ADC elected was the MCP3008 from *Microship* [34]. It's a low cost solution (3€) and as we mentioned above it has 8 analogue input channels that can be configured for single ended and differential ADC conversions. The MCP3008 is a 10-bit ADC that can convert up to 200 kilo samples per second.

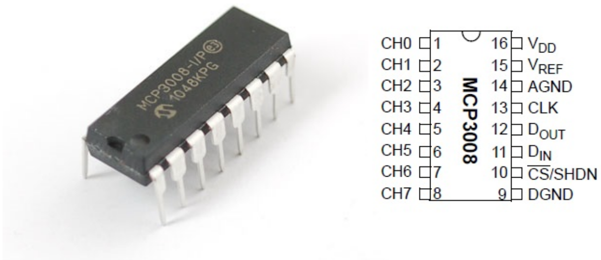


Figure 3.11: MCP3008.

In order to have the lowest power consumption possible we opted to use the 3.3 V supply from the microprocessor alternatively to use the 5 V. Besides that, with 3.3 V supply and considering that MCP3008 is a 10 bit ADC, we can achieve a lower resolution:

$$LSB = \frac{3.3\,\mathrm{V}}{1023} = 3.22\,\mathrm{mV} \tag{3.2}$$

3.1.4 Microprocessor

Figure 3.12: Raspberry Pi 2.

The Raspberry Pi is a microprocessor based single-board computer (SBC). One of the advantages about the Raspberry Pi is that it is running Linux. Thanks to Linux the Raspberry Pi benefits from a far more flexible and powerful development environment. We can program for it in C++, Java, Python or some other language. Basically a Raspberry Pi works as if a normal computer at a relatively low price (30€). In the past we worked with Raspberry Pi and so we are already familiarized with its development environment. Besides that the Raspberry Pi is also one of the best cost-effective microprocessors. One of the main reasons to use the Raspberry Pi was also that we needed a real-time solution. The new model is extremely fast once it has a 900 MHz quad-core ARM Cortex-A7 CPU [35] which is 6 times faster than the old Raspberry Pi model B+.

Before any decision was taken, others microprocessors like Intel Edison [36], Arduino Uno [37] or as well as Beagle Bone [38] were explored due to their potential and contribute for our solution. Table 3.2 shows some relevant specifications related to these microprocessors

Table 3.2: A brief outline of a few microprocessors.

	Raspberry Pi 2	**Beagle Bone Black**	**Intel Edison**	**Arduino Uno**
CPU	Cortex A7	Cortex A8	Atom + Quark	ATmega328
Cores	4	1	2 + 1	-
Clock Speed	900 MHz	1 GHz	500 MHz	16 MHz
RAM	1 GB	512 MB	1 GB	32 KB
GPIO	40 pin	2x46 pin	70 pin	14 pin
Power Supply	3.3 V or 5 V	5 V	3.3 V to 4.5 V	5 V
Price	30€	>40€	75€	20€

Afterwards analysing all the devices above we thought that the Raspberry Pi 2 was a reasonable choice to work as a microprocessor. Although, others microprocessors like the ones we mentioned could be perfectly used in our solution. The final algorithm was developed in C++ and it could easily be adjusted to run inside those devices.

For analogue reading, we used the MCP3008 connected to the Raspberry Pi 2 through SPI Interface. We opted for this bus protocol because it was easy to connect and didn't require any additional components. Furthermore, Raspberry Pi's GPIO header already supports this protocol. Fortunately, it already exists a useful library that allows accessing the GPIO pins of the Raspberry Pi - **wiringPi library**. With this library, we can easily read the analogue output of the sensors. Further we will explain, with some detail, on how it works.

3.2 Building the prototype

After a few weeks installing all the electric components and configuring the system, we were finally able to get some values from FSR 402 sensors. Subsequently, a platform that simulates the floor was demanded in order to create a real scenario situation. While we were searching for the perfect material to use in the prototype, we came to the conclusion that two types of material was needed: an unbending surface for spreading equally the applied forces, and a shock absorber surface for increasing the force range of the sensors.

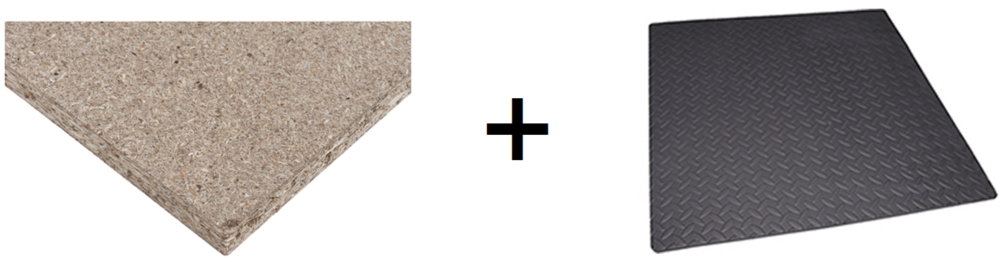


Figure 3.13: Chipboard surface (left) and Shock Absorber surface (right) from *Leroy Merlin*.

The smoother surface was installed over the sensors so we can absorb the applied forces and at the same time protect sensors sensitive area.

The unbending surface was used as a rigid layer over the shock absorber in order to ensure that the pressure is evenly distributed across active sensing area, avoiding local high pressures at the edges. Besides, this surface allows spreading the force over a greater area and therefore the overall pressure is kept below the saturation point (as we mentioned in section 3.1.1).

As a lower layer we added a chipboard surface because it is a cheaper and lighter solution and also is hard enough to support a footstep without bending. Additionally, this last layer is used to fix FSR 402 sensors at each corner of the smart floor.

Figure 3.14 illustrates the physical appearance of the prototype.

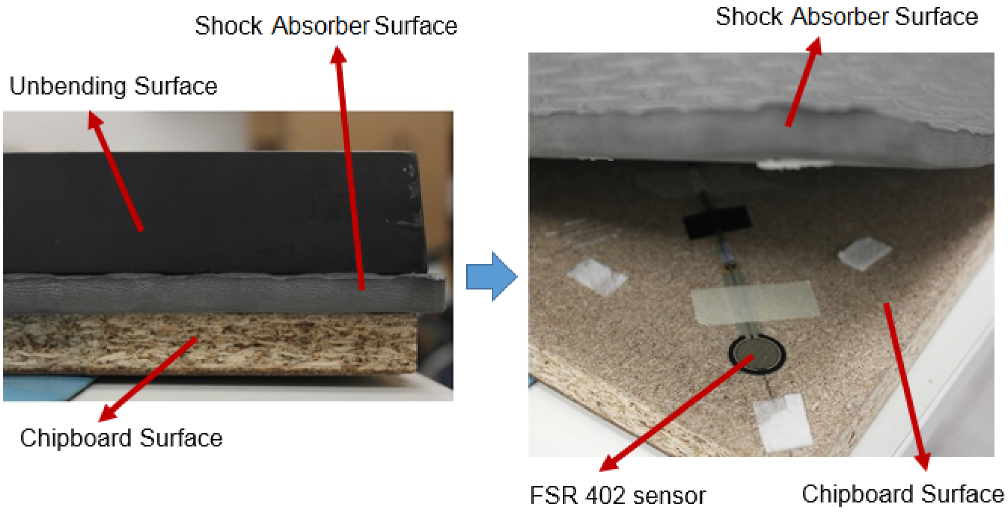


Figure 3.14: Side view of smart floor and its inside.

For fixing sensors at the corner of the tile we simply used tape. At this stage we think that's enough to ensure that sensors stay unmoved. The same technique is also used for sticking together both layers.

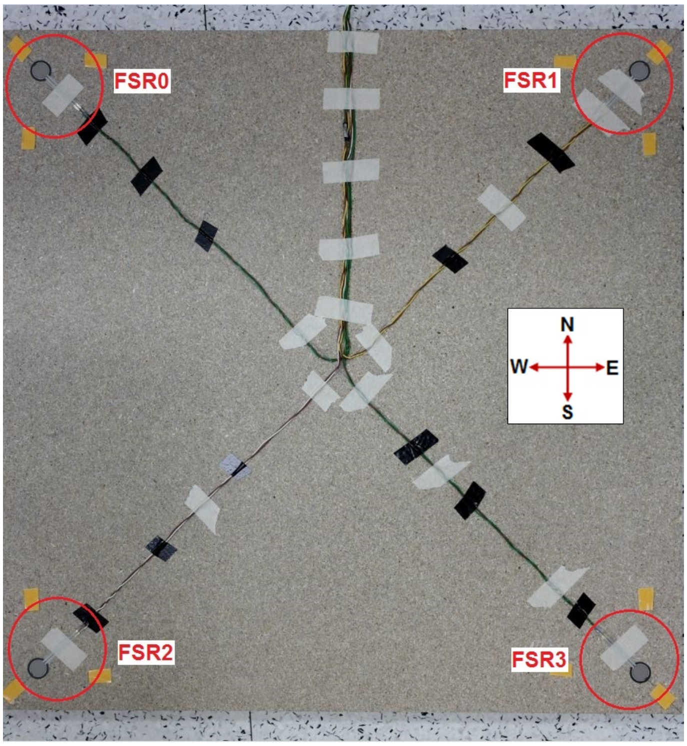


Figure 3.15: The bottom layer with FSR 402 sensors.

3.3 Final Budget

In Chapter 1 we purposed ourselves to come up with a low cost solution. Doing the maths, we can conclude that we maintain our goal of building a solution with less than 100€. Table 3.3 can confirm this achievement.

Table 3.3: Final budget.

Name	**Quantity**	**Seller**	**Unit Price**
FSR 402	4	DigiKey	6.72€
MCP3008	1	Farnell	2.18€
Raspberry Pi 2	1	Farnell	32.87€
5 V Power Supply	1	Farnell	4.19€
8 GB SD card	1	Farnell	8.89€
LM324	1	Farnell	0.413
Shock Absorber Surface 62x62 cm	1	Leroy Merlin	5.29€
Chipboard Surface 120x60 cm	1	Leroy Merlin	8.99€
		Total Price:	90€

A few aspects must be taken into account after analysing this table. More than a third of the total cost of the project is for the microprocessor. In case of we intend to use only 1 smart floor (60x60 cm total area) we can reduce considerably the total cost by substituting the Raspberry Pi 2 for a less expensive device. It's also important to refer that for larger sensing active areas the total cost per square meter will be reduced. Despite we would need to buy larger amounts of sensors, we would be able to work with a single Raspberry Pi (up to 24 sensors). At last, if could apply the solution to a raised floor we could use the existing panels to work as the unbending surface. We also could use the pedestals of the raised floor for fixing the FSR 402. Further we will present with more detail this idea (section 5.1).

In short, imagining that we would have 100 prototypes (6 mx6 m active area) taking advantage of a raised floor. In this case we would have a final price of less than 30€ (per tile).

3.4 Software

Along this section we will present the software used during the dissertation. The development of this project was mostly focused on 3 steps:

1. Configuring the Raspberry Pi;
2. Gathering data from sensors;
3. Modelling the algotihm using matlab

4. Implement the algorithm in C++ to run in real-time in the RPi.

In the next subsections it will be explored the 3 steps of the development of our solution. We will explain how to configure and calibrate the system and also present graphically data collected from the sensors in order to better understand the design of the Algorithm (which will be explained further). Along this section we will give special highlighting to MATLAB because it was a fundamental tool for modulation of the system.

We will see later that during the data processing, there are some important characteristics that we should extract. We believe that if we can acquire all the information that is presented bellow, we will be able to identify the person's walking:

- Person's weight. This measure will be accomplished by summing all four FSR values;
- Exact time when a footstep is detected (according to NTP server);
- Time on heel to time on toe;

Even if we can't get a precise measure of the person's weight we can at least do a qualitative classification of the weight (light, heavy,...) depending on the adopted thresholds. Considering the last item, if we can synchronize both clocks of the micro controller and the user's smart phone we will be able to save the exact time that the foot hit the smart floor. Since the smartphone is also capable of calculating the time that the foot hit the ground (using accelerometer), later we will be able to match these two information time and identify the user (impact time registered by the smart floor and impact time extracted using the accelerometer).

3.4.1 Raspbian and useful libraries

> "Raspbian is an unofficial port of Debian Wheezy armhf with compilation settings adjusted to produce optimized "hard float" code that will run on the Raspberry Pi. This provides significantly faster performance for applications that make heavy use of floating point arithmetic operations. All other applications will also gain some performance through the use of advanced instructions of the ARMv6 CPU in Raspberry Pi." [39]

Figure 3.16: Raspbian is the Debian Wheezy adapted to Raspberry Pi.

Raspbian can be directly downloaded from the Raspberry Pi webpage and it's a user's friendly operating system. It is very similar to Linux and for that reason we quickly configured the system. All the steps for configuration can be found at the Raspberry Pi webpage.

As we mentioned before, a useful library for our project was the wiringPi library [40]. WiringPi is a GPIO access library for the BCM2835 used in the Raspberry Pi. It's designed to be familiar and easy to use. It also has a lot of different functions that really help to acquire all the data through the Raspberry Pi. We now present the ones that were useful for our project:

- void pinMode (int pin, int mode);
- void digitalWrite (int pin, int value);
- wiringPiSetup (void);
- analogRead (int pin);

The first function sets the mode of a pin (for example to either INPUT or OUTPUT). The second function writes the value HIGH or LOW to the given pin. It's clear to notice that *digitalWrite* function depends on the previous one because the pin must be set as an output earlier. In our particular case, we used both functions to turn on a green LED whenever the main algorithm is running, in other words, while someone is stepping on the smart floor.

There are four ways to initialise wiringPi. We used the *wiringPiSetup* for assuming that the calling program is going to be using the wiringPi pin numbering scheme. Basically it's a simplified numbering scheme which provides mapping from virtual pin to the real underlying GPIO numbers. However, the user can choose another setup for the wiringPi (e.g. use *wiringPiSetupGpio* for using GPIO numbers directly with no re-mapping).

At last but not least, as the name indicates, the *analogRead* is used to get data from the ADC. This function does all the hard work and we just need to insert the channel pin number. Besides, the same author offers a library for the MCP3008. This library is implemented concerning the communication protocol of our ADC.

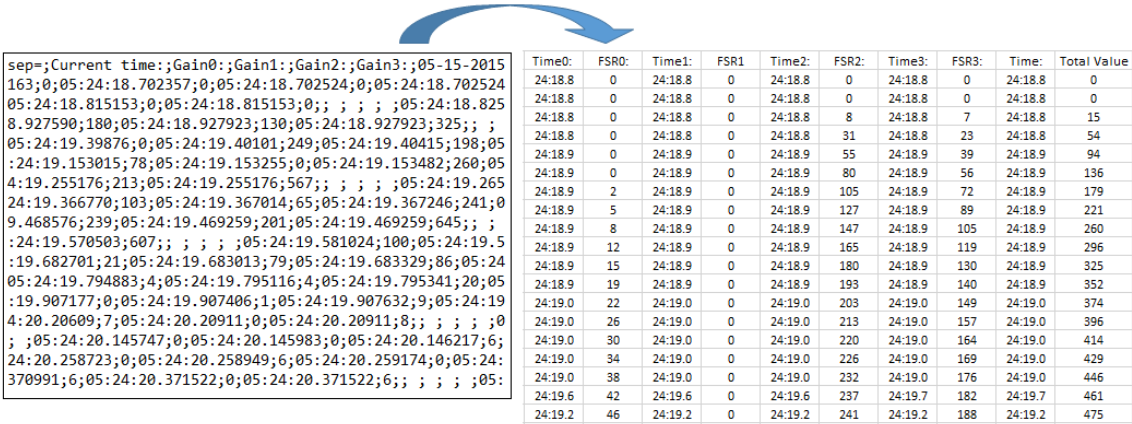
Allying the wiringPi library to the MCP3008 library, with a single instruction we are able to get the analogue values of FSR 402 sensors.

3.4.2 Development Tools

Considering that our solution was designed to be used as a checkpoint for position reference, it is appropriated to consider that in an average size building we could have dozens of replicas of our prototype. Concerning this mass production idea, we decided to develop the algorithm in one of the most used programming languages (C++) in order to allow the main code running on other devices. Thus, it should be no problem to apply our code to a lower price microprocessor.

In this section instead of explaining the algorithm (we will explore it in section 3.5), we will present some routines that we used to start reading values and also explain the dataset collection.

One of the first routines that we implemented is *analogread4.c*. The *analogread4.c* function was responsible for extracting to a *.txt* file the ADC values received by the MCP3008. It also saved the exact time that the values were read. Having a time perception of the events will be useful for designing the algorithm. This routine gathers the outpuit from the 4 FSR sensors and saves it taking into account *.csv*. Later, with this format, we were able to access the information using Excel or even Matlab. Some results are presented in figure 3.17.



```
sep=;Current time:;Gain0:;Gain1:;Gain2:;Gain3:;05-15-2015
163;0;05:24:18.702357;0;05:24:18.702524;0;05:24:18.702524
05:24:18.815153;0;05:24:18.815153;0;; ; ; ; ;05:24:18.825
8.927590;180;05:24:18.927923;130;05:24:18.927923;325;; ;
05:24:19.39876;0;05:24:19.40101;249;05:24:19.40415;198;05
:24:19.153015;78;05:24:19.153255;0;05:24:19.153482;260;05
4:19.255176;213;05:24:19.255176;567;; ; ; ; ;05:24:19.265
24:19.366770;103;05:24:19.367014;65;05:24:19.367246;241;0
9.468576;239;05:24:19.469259;201;05:24:19.469259;645;; ;
:24:19.570503;607;; ; ; ; ;05:24:19.581024;100;05:24:19.5
:19.682701;21;05:24:19.683013;79;05:24:19.683329;86;05:24
05:24:19.794883;4;05:24:19.795116;4;05:24:19.795341;20;05
:19.907177;0;05:24:19.907406;1;05:24:19.907632;9;05:24:19
4:20.20609;7;05:24:20.20911;0;05:24:20.20911;8;; ; ; ; ;0
; ;05:24:20.145747;0;05:24:20.145983;0;05:24:20.146217;6;
24:20.258723;0;05:24:20.258949;6;05:24:20.259174;0;05:24:
370991;6;05:24:20.371522;0;05:24:20.371522;6;; ; ; ; ;05:
```

Time0:	FSR0:	Time1:	FSR1	Time2:	FSR2:	Time3:	FSR3:	Time:	Total Value
24:18.8	0	24:18.8	0	24:18.8	0	24:18.8	0	24:18.8	0
24:18.8	0	24:18.8	0	24:18.8	0	24:18.8	0	24:18.8	0
24:18.8	0	24:18.8	0	24:18.8	8	24:18.8	7	24:18.8	15
24:18.8	0	24:18.8	0	24:18.8	31	24:18.8	23	24:18.8	54
24:18.9	0	24:18.9	0	24:18.9	55	24:18.9	39	24:18.9	94
24:18.9	0	24:18.9	0	24:18.9	80	24:18.9	56	24:18.9	136
24:18.9	2	24:18.9	0	24:18.9	105	24:18.9	72	24:18.9	179
24:18.9	5	24:18.9	0	24:18.9	127	24:18.9	89	24:18.9	221
24:18.9	8	24:18.9	0	24:18.9	147	24:18.9	105	24:18.9	260
24:18.9	12	24:18.9	0	24:18.9	165	24:18.9	119	24:18.9	296
24:18.9	15	24:18.9	0	24:18.9	180	24:18.9	130	24:18.9	325
24:18.9	19	24:18.9	0	24:18.9	193	24:18.9	140	24:18.9	352
24:19.0	22	24:19.0	0	24:19.0	203	24:19.0	149	24:19.0	374
24:19.0	26	24:19.0	0	24:19.0	213	24:19.0	157	24:19.0	396
24:19.0	30	24:19.0	0	24:19.0	220	24:19.0	164	24:19.0	414
24:19.0	34	24:19.0	0	24:19.0	226	24:19.0	169	24:19.0	429
24:19.0	38	24:19.0	0	24:19.0	232	24:19.0	176	24:19.0	446
24:19.6	42	24:19.6	0	24:19.6	237	24:19.7	182	24:19.7	461
24:19.2	46	24:19.2	0	24:19.2	241	24:19.2	188	24:19.2	475

Figure 3.17: Dataset Collection

The C++ code was developed in Sublime Text [41] and then compiled with Cygwin [42]. Therefore, we used Putty [43] to create a SSH connection to communicate with the Raspberry Pi. Through the SSH connection we also transferred files using the WinSCP tool [44]. In short, all the developed code was designed in a personal computer (modulation) and afterwards it was inserted into the Microprocessor (implementation). The figure 3.18 illustrates the role of each tool for analogue reading of FSR sensors. As we can see, in this case, pressure was being applied only to FSR0.

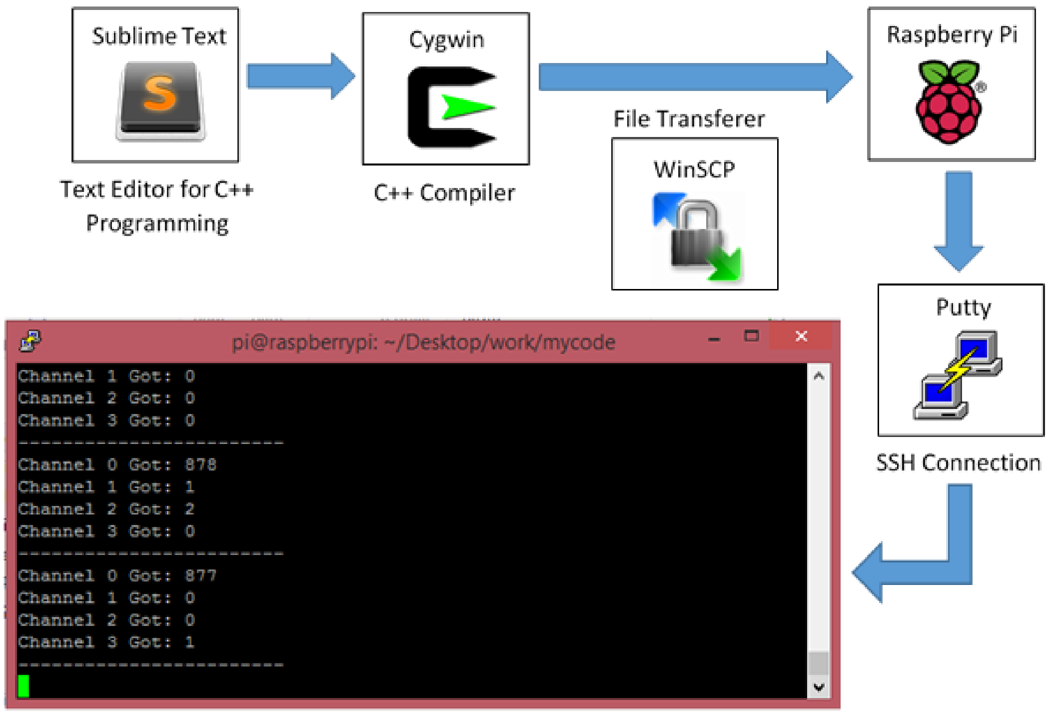


Figure 3.18: Analogue reading when pressure was being applied only to FSR0 sensor.

As soon as we were able to get pressure data from the sensors we decided to analyse this information graphically in order to better understand the system behaviour when a person walked over the smart floor. At first instance, we study the system performance using Excel:

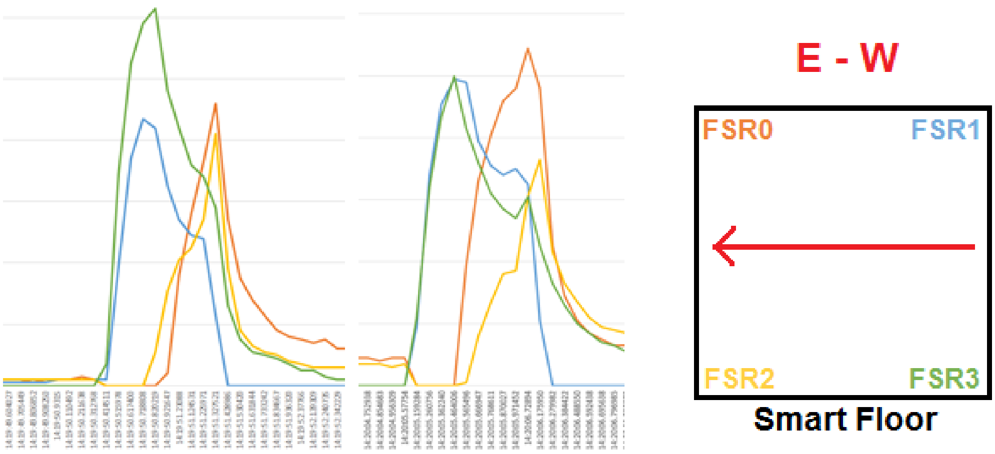


Figure 3.19: System performance when a person walks from right (East) to left (West).

Looking at the results presented above we can define two main events: firstly an abrupt ground reaction force due heel striking, and then another sharp ground reaction force associated to the toe push-off. Figure 3.20 explains these events.

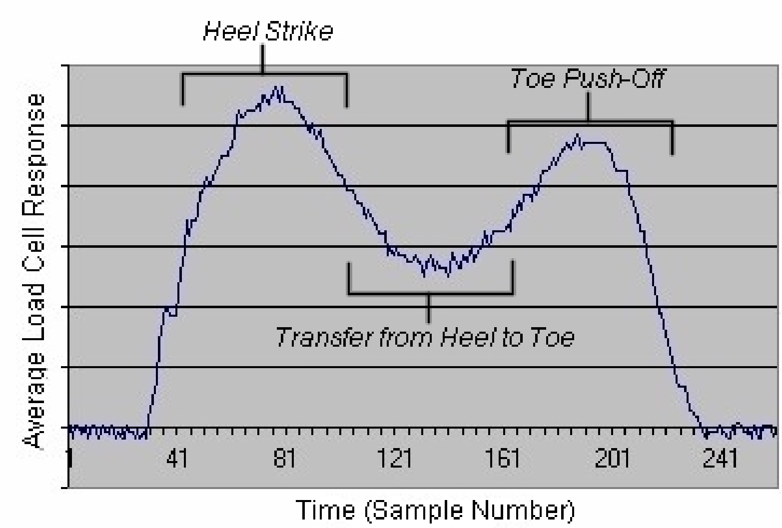


Figure 3.20: Sample ground reaction force (GRF) profile of a single Load Cell [45].

To improve data's analysis we switch from Excel to MATLAB. We used Matlab to analyse the collected data and model the algorithm. The following subsection will explain all the performed procedures.

3.4.3 MATLAB

> "MATLAB is a high-level language and interactive environment for numerical computation, visualization, and programming. Using MATLAB, you can analyse data, develop algorithms, and create models and applications. The language, tools, and built-in math functions enable you to explore multiple approaches and reach a solution faster than with spreadsheets or traditional programming languages, such as C/C++ or Java." [46]

Concerning the advantages mentioned above, on the context of this dissertation, MATLAB version R2014a was used for analysing the incoming data from the FSR sensors. Plotting these signals, it's possible to detect the two main phases associated to human gait. Similar to Figure 3.20, we can distinguish the heel striking from the toe-push off.

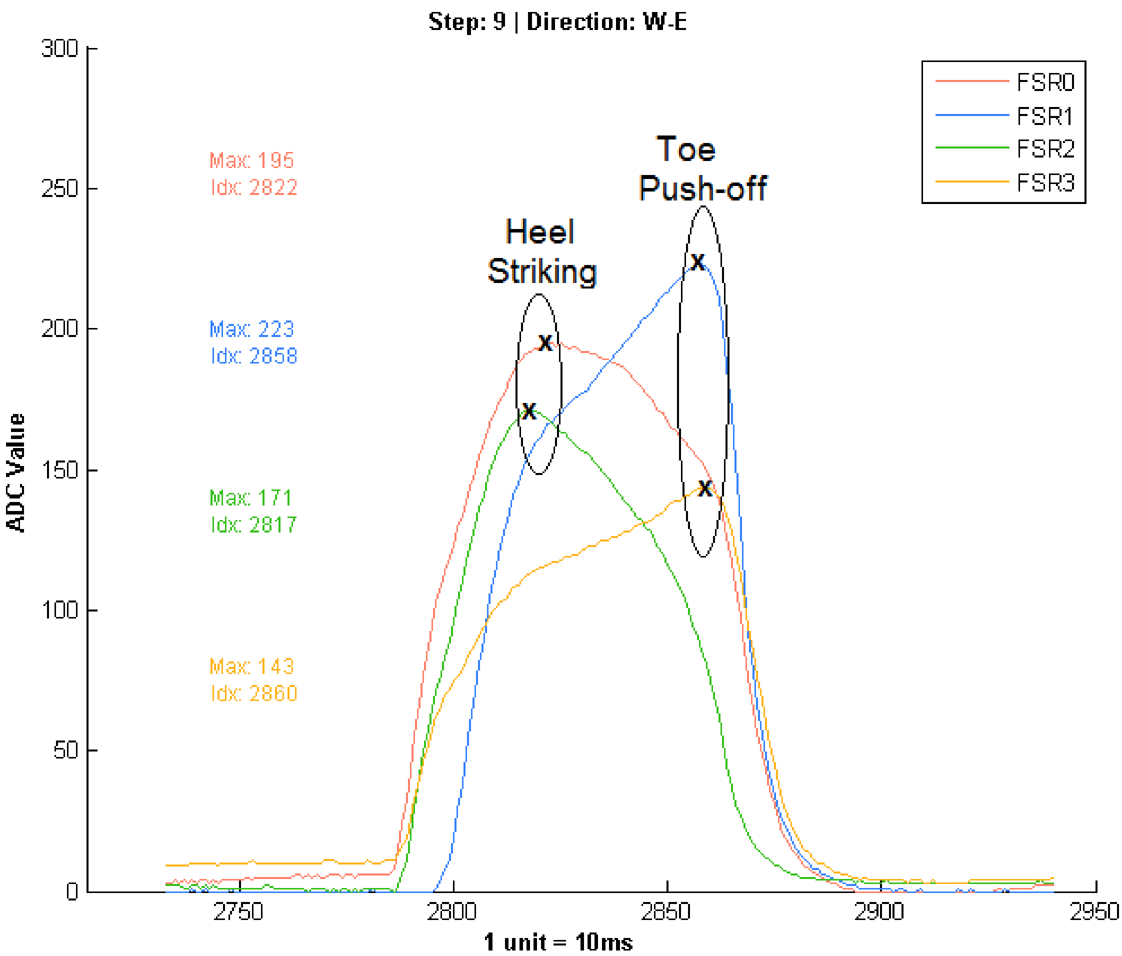


Figure 3.21: System performance when a person walks from left (West) to right (East)

After analysing a few samples from several tests with previously known directions, we start designing the algorithm for direction estimation. The MATLAB was also used to design and implement Fuzzy Logic. At the end MATLAB revealed to be a useful tool that helped us to reach a solution faster than expected. For example, using a few standard functions of MATLAB like *max* and *min* we conclude that depending on the direction, the footsteps were following a pattern. Based on finding maximum and minimum values, we start building the algorithm. Another advantage of using MATLAB is the availability of simulating fuzzy logic systems calling, for example, the *evalmf* function for designing memberships. In next section we will demonstrate the algorithm.

3.5 The Algorithm

As we said before, in a first stage, we opted to use MATLAB for the development of the algorithm. When we finished the algorithm, we tried to use the *MATLAB Coder™* to convert the algorithm to c++ language.

In MATLAB we were processing multiple steps at once, in other words, we start by acquiring all the data and afterwards we import and process all the information using MATLAB (sometimes more than 10 steps). We also used MATLAB for updating the algorithm until we reached 100% of correct directions. In other words, we start with a modulation stage (in MATLAB) and when the algorithm was completed we jumped to the implementation stage (on the Raspberry Pi).

However, our final solution was designed for running at real-time, which means that we just needed to process one step at time instead off collecting several footsteps. For that reason, our algorithm suffered a few changes. In short, we used MATLAB for designing the main code and then we manually were readjusting the algorithm until we had a functional c++ code running in Raspberry Pi. Along the design of the algorithm we develop the following routines:

1. **int calibration()**. This function is used for balancing the four FSR values. In other words, before any measurement, we set all values to zero. Using this function we avoid potential sloping floor that could compromise the results. Figure 3.22 illustrates the output from *calibration*. Looking at the average values collected we see that the floor is unbalanced;

```
------------------------------------------

06-16-2015  12:19:34.346813

Calibration? [Y/N]: y

We will now start Calibration!
Please don't touch the system.

3 seconds remaining...

2 seconds remaining...

1 seconds remaining...

0 seconds remaining...


Calibration complete!

The average values were:

FSR0 Got: 1013
FSR1 Got: 994
FSR2 Got: 968
FSR3 Got: 1013

------------------------------------------

System is ready!

------------------------------------------
```

Figure 3.22: Console output after Calibration.

2. **int nfsrinfo()**. When someone presses the tile and the sum of the four FSR values (*totalfsr*) exceeds the threshold value, we start gathering and saving the data. We stop the analogue reading as soon as the *totalfsr* value is back to zero (no pressure applied).

3. **int maxidxinfo()**. After we had the information about all FSR, we find the maximum values registered of each sensor and also the time when they occurred (in other words, the index). Ideally, at the end, we would have 2 pairs of similar values (max and index): one representing the heel striking and the toe push-off.

4. **int readjustidx()**. This function is used for non-ideal situations. Imagine that one of the sensors registered multiple max values. This can happen when the foot applies the same exact pressure for too long resulting on multiple samples with the same analogue value. In these cases, this function readjust the index in order to be as near as possible to the max value registered by the corresponding sensor.

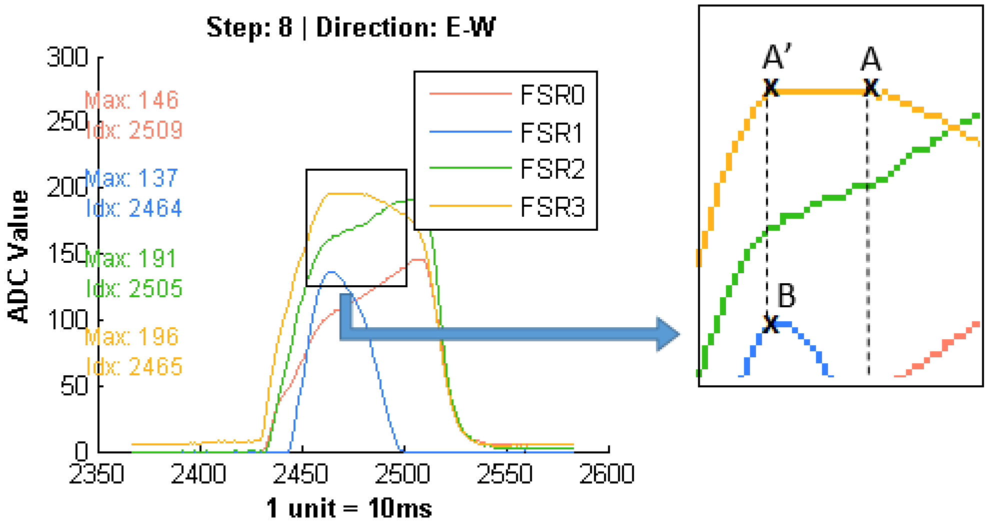


Figure 3.23: Example of index readjustment (A). Considering the final direction East to West, the algorithm checks all the multiple max values of FSR3 and chooses the one that is nearest of FSR1 index (which is A').

5. **int fuzzymf(char *mfname, float *value)**. This function creates the membership function *mfname* and returns the corresponding result according to the input *value*. In subsection 3.5.1 we will explore this topic.

6. **int fuzzyresult()**. We use this routine for labelling the acquired data. The function compares the information of one sensor among all the others, resulting in a 4 by 4 matrix. Next, it uses *fuzzymf* for labelling the results according to the fuzzy memberships.

7. **int setheeltoe()**. This routine solves the problem when there is a missing or false HEEL/TOE. We know that in a normal footstep we will have 2 FSR as HEEL and the others 2 FSR as TOE. Based on this information, if a FSR sensor doesn't detect any pressure, this function will set the missing FSR as HEEL or TOE. The function also checks for false HEEL/TOE. In other words, if we have more than 2 HEELs or TOEs, this routine will correct them.

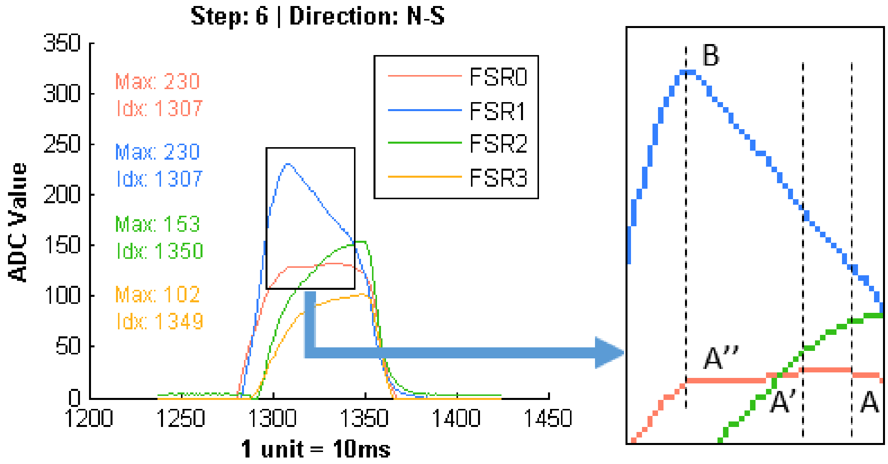


Figure 3.24: Example of a false TOE (A). Considering the final direction North to South, the algorithm checks all the multiple max values of FSR0 and chooses the one that is nearest of FSR1 index (which is A'). However, since A' is lightly above A" (which should be the correct index), the *readjustidx* function is insufficient. To avoid this situation, we implemented the *setheeltoe* function.

8. **int setthreshold()**. This function combines all the fuzzy results to a single value that later will be used for checking the side where the foot hit the smart floor. At this point, since all the missing or false HEEL/TOE are already correct, this routine also groups the FSR sensors as "neighbors" (if both indexes are too close that means they were pressed simultaneously);

9. **int fuzzyrules()**. For direction estimation, we use several fuzzy rules. In subsection 3.5.2 we will explore this topic as well as describe some examples of fuzzy rules;

```
******************************************
 Max ADC Value:          405
 Impact time (NTP):      12:19:44.29683
 Time on HEEL to TOE:    520 ms
 DIRECTION estimation:   W-E
******************************************
```

Figure 3.25: Console output after Direction estimator is complete.

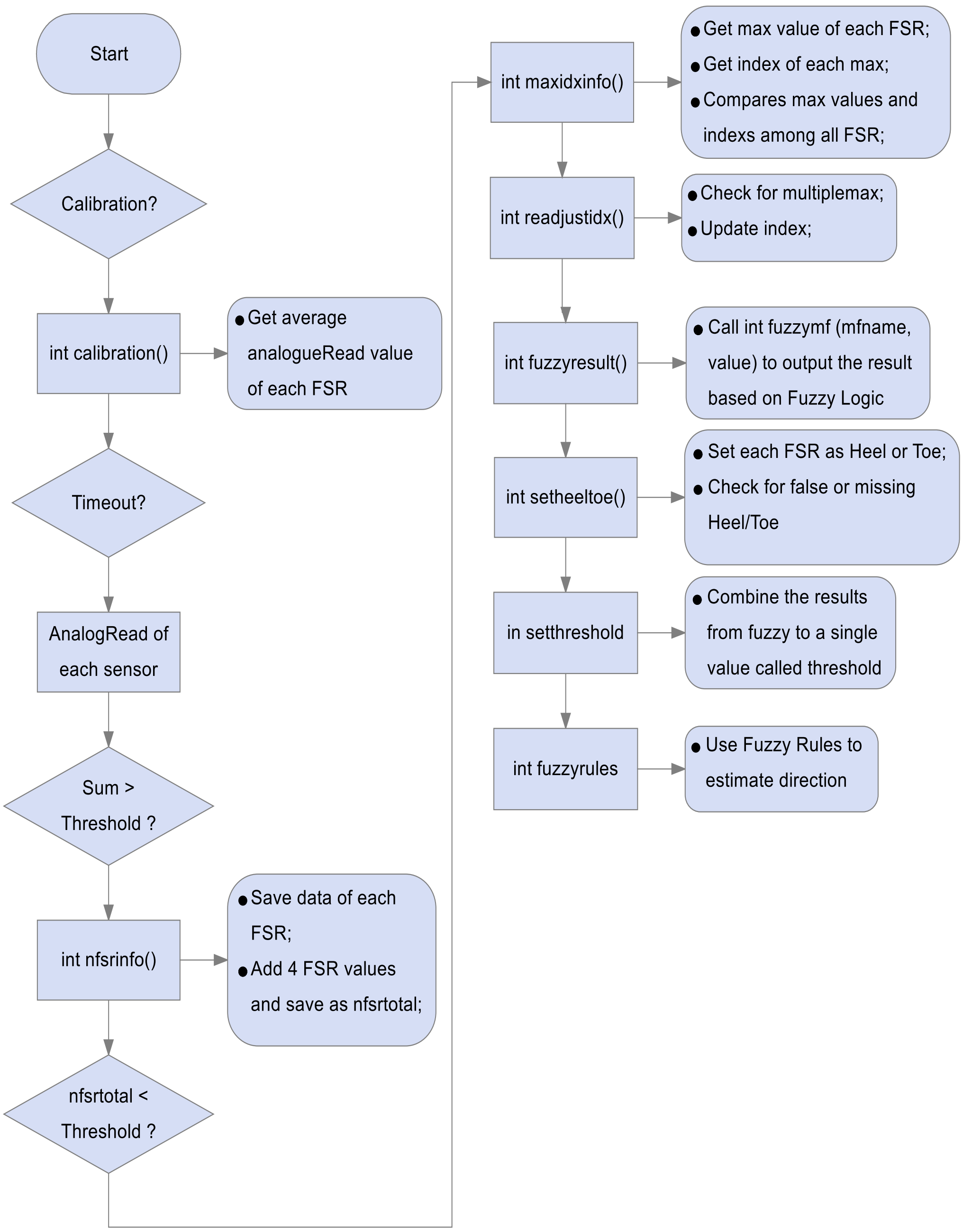


Figure 3.26: Flowchart representing the algorithm

3.5.1 Fuzzy Logic Memberships

In our Fuzzy Logic implementation, we decided to analyse the following characteristics:

1. Time interval between indexes. In other words, check if two FSR max values took place at the same time;

2. Difference between max values. By comparing max values of each FSR, we will know if it was applied the same force to both FSR;

3. Determine if a max value occurred at the beginning (HEEL) or at the end (TOE) of the footstep.

Afterwards we acquired the information associated to the FSR, we made a comparison between all the sensors, resulting in a four by four matrix. As soon as we got the results, we needed to classify them. We decided to use memberships function for classifying all the characteristics mentioned above. These functions are responsible for managing the degrees of partial truth associated to the fuzzy logic implemented. The memberships function were designed regarding the firsts performed tests. The figure 3.27 illustrates the final parameters used in our approach.

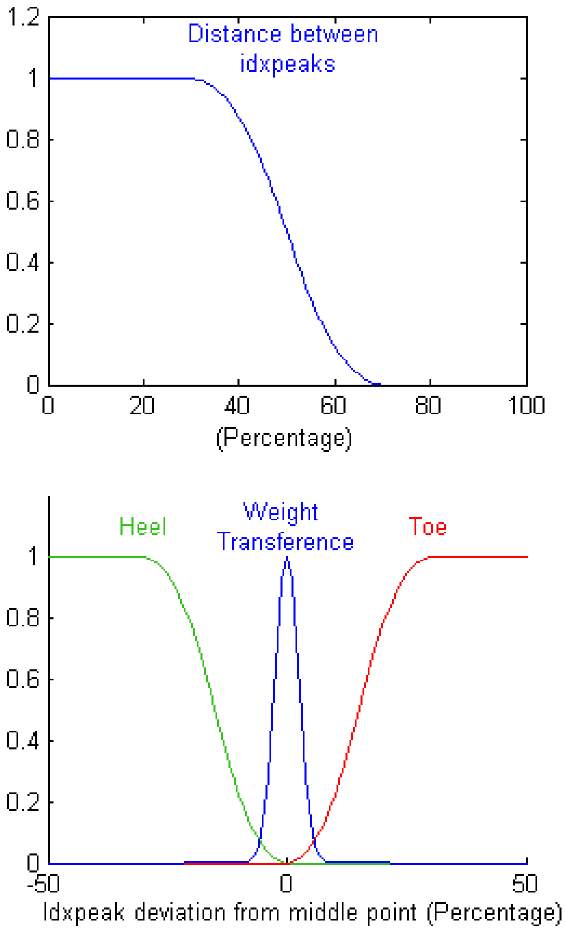


Figure 3.27: Membership functions used in the algorithm.

3.5.2 Fuzzy Logic Rules

At the end of our algorithm, we implemented a set of fuzzy rules. These rules were applied in order to get the direction estimation. Since we were using Fuzzy Logic, we were allowed to expand the algorithm for new non-ideal situations. For example, while we were doing the tests we notice a few events that weren't covered by the fuzzy rules such as when part of the foot hit outside the smart floor. Considering table 3.4, we conclude that most of the fuzzy rules applied are based on analysing the footstep position, in others words, knowing which sensors were activated first.

Table 3.4: Fuzzy rules examples.

Nr.	Fuzzy Rule
1	IF **FSR0** AND **FSR1** IS **HEEL** AND **FSR2** AND **FSR3** IS **TOE** THAN **DIRECTION** IS **N_S**
2	IF **FSR3** AND **FSR2** IS **HEEL** AND **FSR0** AND **FSR1** IS **TOE** THAN **DIRECTION** IS **S_N**
3	IF **FSR0** AND **FSR2** IS **HEEL** AND **FSR1** AND **FSR3** IS **TOE** THAN **DIRECTION** IS **W_E**
4	IF **FSR1** AND **FSR3** IS **HEEL** AND **FSR0** AND **FSR2** IS **TOE** THAN **DIRECTION** IS **E_W**

Table 3.5 shows the set of rules that we use for checking on each side the foot touch the smart floor. Afterwards we have a final direction estimation, we compared the max values of both FSR. If we had a "HEEL FSR" higher than its pair and subsequently the corresponding "TOE FSR" also higher than its pair it means that more pressure was applied to one side of the tile.

Table 3.5: Fuzzy rules for checking which side the foot hit the ground.

Nr.	Fuzzy Rule
5	IF **DIRECTION** IS **N_S** AND **FSR0** IS **HIGHER** THAN **FSR1** AND **FSR2** IS **HIGHER** THAN **FSR3** THAN **DIRECTION** IS **N_S (W)**
6	IF **DIRECTION** IS **N_S** AND **FSR1** IS **HIGHER** THAN **FSR0** AND **FSR3** IS **HIGHER** THAN **FSR2** THAN **DIRECTION** IS **N_S (E)**
7	IF **DIRECTION** IS **S_N** AND **FSR2** IS **HIGHER** THAN **FSR3** AND **FSR0** IS **HIGHER** THAN **FSR1** THAN **DIRECTION** IS **S_N (W)**
8	IF **DIRECTION** IS **S_N** AND **FSR3** IS **HIGHER** THAN **FSR2** AND **FSR1** IS **HIGHER** THAN **FSR0** THAN **DIRECTION** IS **S_N (E)**
9	IF **DIRECTION** IS **W_E** AND **FSR0** IS **HIGHER** THAN **FSR2** AND **FSR1** IS **HIGHER** THAN **FSR3** THAN **DIRECTION** IS **W_E (N)**
10	IF **DIRECTION** IS **W_E** AND **FSR2** IS **HIGHER** THAN **FSR0** AND **FSR3** IS **HIGHER** THAN **FSR1** THAN **DIRECTION** IS **W_E (S)**
11	IF **DIRECTION** IS **E_W** AND **FSR1** IS **HIGHER** THAN **FSR3** AND **FSR0** IS **HIGHER** THAN **FSR2** THAN **DIRECTION** IS **E_W (N)**
12	IF **DIRECTION** IS **E_W** AND **FSR3** IS **HIGHER** THAN **FSR1** AND **FSR0** IS **HIGHER** THAN **FSR2** THAN **DIRECTION** IS **E_W (S)**

Chapter 4

Tests and Results

Along this chapter it will be described the performed tests. 7 people with an average weight between 65 and 95 Kg tested the final system. The subjects heights were from 175 to 190 cm and their shoe size were from 39 to 44 (Euro Sizes), which conceded different sizes of footsteps.

At first stage, the individuals walked over the tile following previously defined directions. The data was collected and processed later using MATLAB (modulation). At the end, in order to better reproduce a real case scenario, we assembled a few tiles for simulating a raised floor. In this case, data was already processed in the Raspberry Pi at real-time, using C++ Language (implementation).

For the final tests, we also elevated the floor around the smart floor in order to avoid false positives results. In other words, since the smart floor has 3 layers, the subjects needed to climb to the top of the smart floor (3-5 cm), which could change the normal behaviour of human gait.

One of the first tests realized in order to prove the concept of our system is presented in figure 4.1.

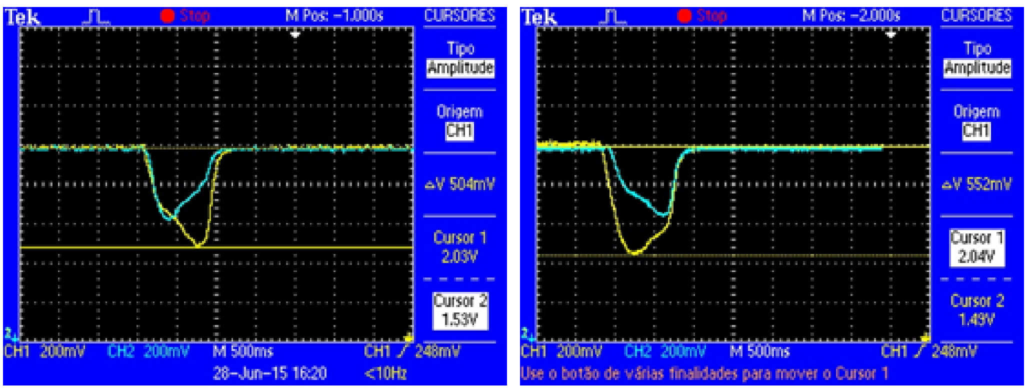


Figure 4.1: Oscilloscope output when someone is walking on the same direction but at opposite orientations (left and right). We can distinguish two max peaks which proves that we might be able to identify the heel striking (in blue on the left image and yellow on the right image) and toe push-off (in yellow on the left image and blue on the right image).

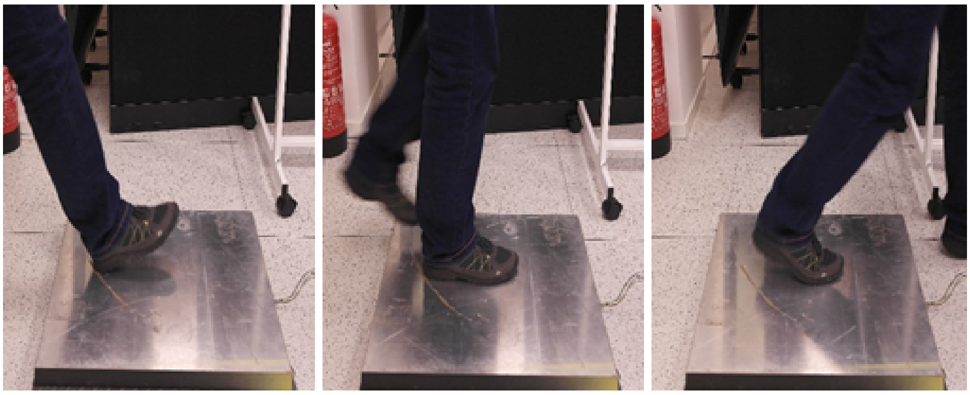


Figure 4.2: First performed tests following previously defined directions. The figure describes the three phases of human gait. Starting from left to right, first we have heel striking, than the weight transference and finally the toe push-off.

As we mentioned above, we start testing the system by gathering data and then processing it using MATLAB. All the performed tests were done under previously defined directions in order to later match the results and check if they were correct. Some results will be presented in subsection 4.1.

Taking into account that we were analysing the pressure applied to each sensor, it was fundamental to have a balanced surface for doing the tests. Sometimes the floor had small highs that unbalanced the smart floor and consequently the results were affected. In order to avoid this problem we tried to do all the tests on the flattest ground possible.

Figures 4.3 and 4.4 illustrated the final test bed used.

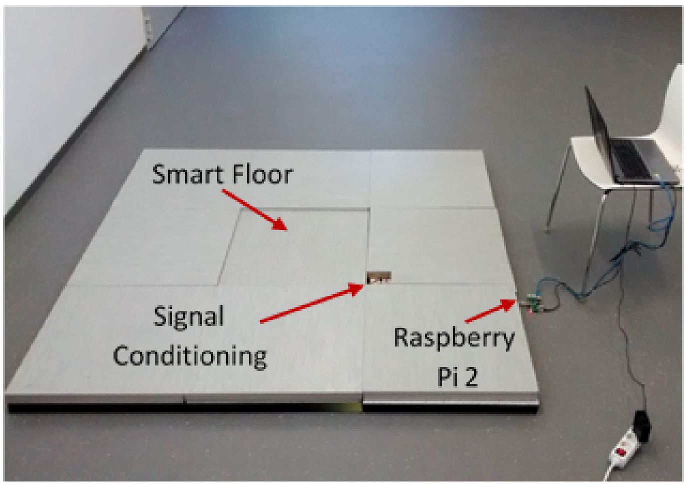


Figure 4.3: Test bed for the final tests.

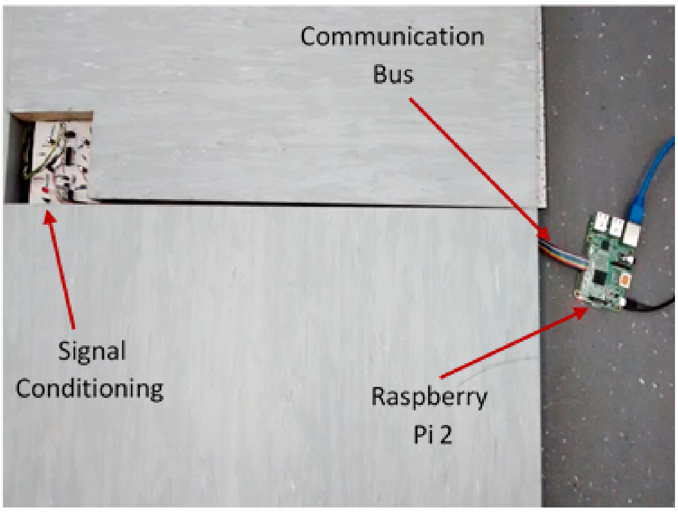


Figure 4.4: Zoom in microprocessor. All the cables and sensors are unnoticeable to the user.

The test bed described was used taking into account two main goals: simulating a real case scenario and proving the idea that all the system can stay undetectable.

4.1 Results

In this subsection we will present the obtained results. Regarding our solution, the final output is a string with the final direction estimation. There are 12 directions and orientations that the algorithm is able to identify:

1. N_S, N_S (W) and N_S (E);
2. S_N, S_N (E) and N_S (W);
3. W_E, W_E (S) and W_E (N);
4. E_W, E_W (N) and E_W (S);

The output mentioned above represents the final direction for when the foot hit the ground, respectively, on the centre, on right or on the left side of the smart floor.

Considering that the final result isn't something measurable, i.e, the final direction is not presented in angle degrees, we can't do a fair measurement of the precision of the final output of the system.

However, more than 100 steps from 7 different people were analysed and the algorithm were able to correctly identify all of them. It also differentiated the ones when the footstep hit the ground on the right/left side of the smart floor. An example of the differentiation mentioned is illustrated in figure 4.5.

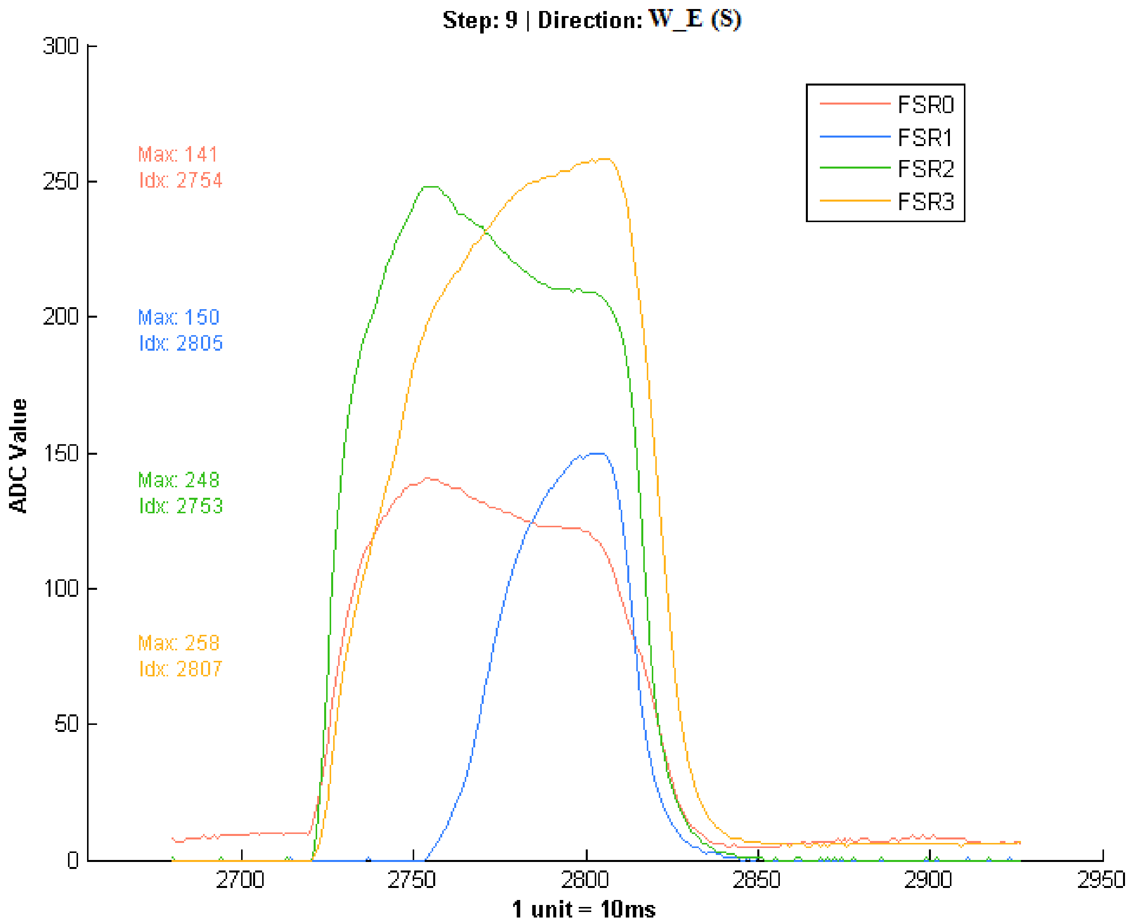
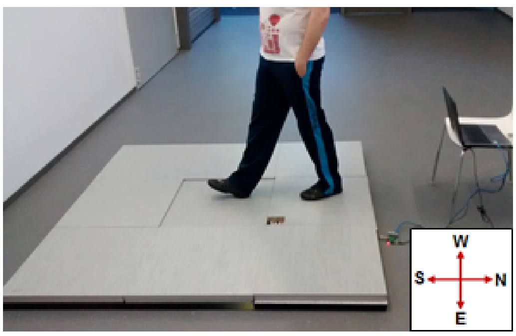


Figure 4.5: Example of a footstep hitting the right side of the smart floor.

The following figures illustrate some performed tests. The main four directions are exemplified as well as the plotting of the output signals.



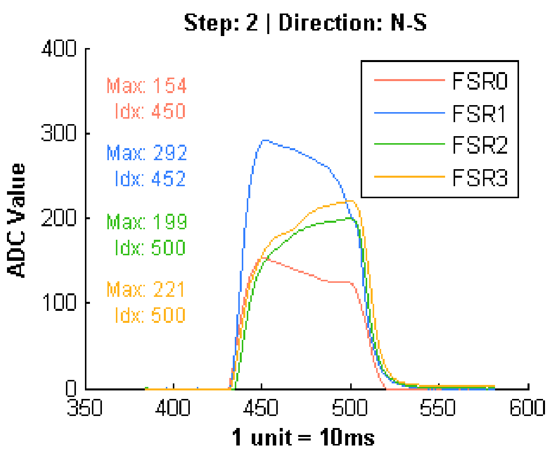


Figure 4.6: Test example of walking from North to South.

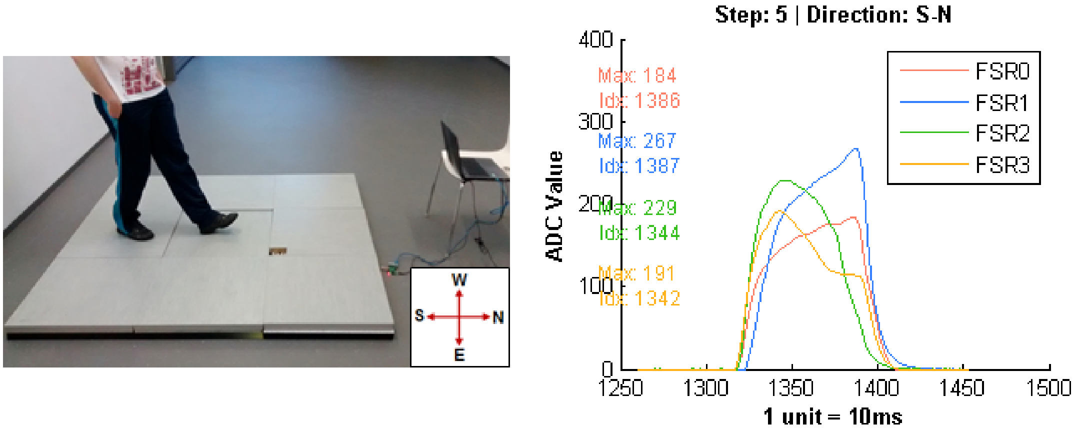


Figure 4.7: Test example of walking from South to North.

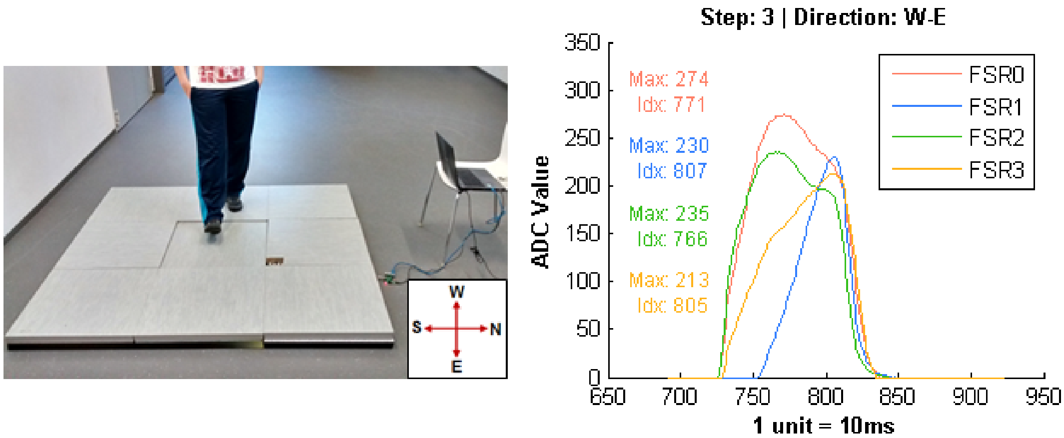


Figure 4.8: Test example of walking from West to East.

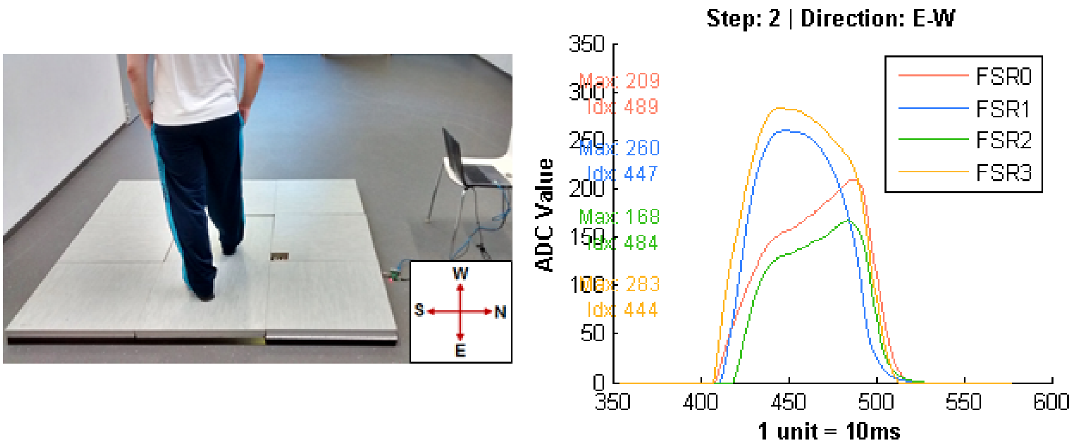


Figure 4.9: Test example of walking from East to West.

A second test was realized in order to evaluate the performance of the smart floor while a subject was walking randomly. The results are presented bellow.

Table 4.1: Results from walking randomly

Nr.	Direction	Output	Nr.	Direction	Output	Nr.	Direction	Output
1	E-W (N)	E-W	**17**	E-W (S)	E-W (S)	**33**	N-S	N-S (W)
2	W-E	W-E	**18**	S-N (E)	NONE	**34**	E-W	E-W
3	E-W (S)	E-W (S)	**19**	W-E	W-E	**35**	W-E (S)	W-E (S)
4	S-N	N-S	**20**	E-W (N)	NONE	**36**	E-W (S)	E-W (S)
5	N-S (W)	NONE	**21**	W-E (N)	NONE	**37**	S-N (E)	S-N (E)
6	S-N (W)	NONE	**22**	E-W (N)	E-W (N)	**38**	N-S (E)	NONE
7	S-N (E)	S-N (E)	**23**	S-N	S-N	**39**	S-N (W)	NONE
8	N-S	N-S	**24**	N-S	N-S	**40**	N-S (W)	N-S (W)
9	E-W (N)	E-W (N)	**25**	E-W	E-W	**41**	N-S (E)	N-S (E)
10	W-E (N)	W-E	**26**	W-E	W-E (N)	**42**	S-N (W)	NONE
11	S-N	S-N	**27**	E-W (S)	E-W (S)	**43**	W-E	W-E
12	N-S	N-S	**28**	W-E (S)	W-E (S)	**44**	E-W	E-W (N)
13	S-N (E)	S-N (E)	**29**	E-W (N)	E-W (N)	**45**	W-E	W-E
14	N-S (E)	NONE	**30**	W-E	W-E	**46**	W-E	W-E
15	E-W	E-W	**31**	N-S	N-S	**47**	S-N	S-N
16	W-E	W-E	**32**	S-N	S-N	**48**	N-S	N-S

Age	23
Sex:	Male
Weight:	95 Kg
Height:	190 cm

Average value read in Calibration (ADC)	
FSR0:	968
FSR1:	986
FSR2:	896
FSR3:	1004

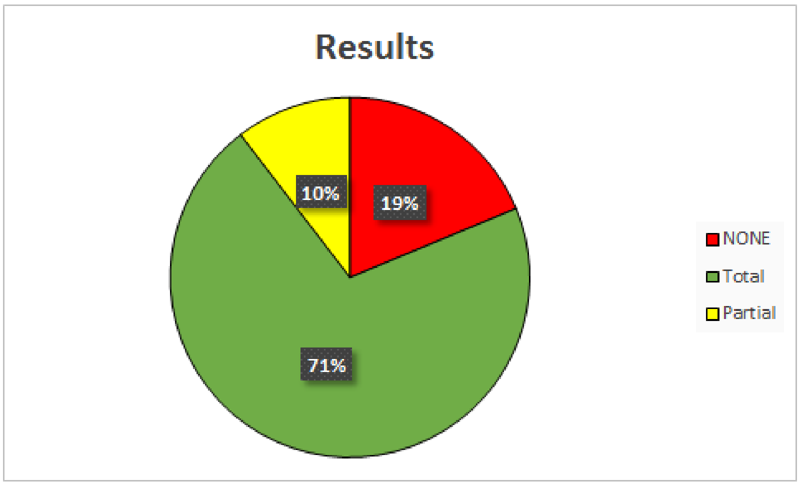


Figure 4.10: Results from table 4.1.

Looking at table 4.1 we can see that the results were satisfactory. Although the smart floor was not balanced (for example, FSR2 registered a ADC value 20% lower than FSR3), we were able of detecting correctly 81% of the tests. We believe that if the smart floor were correctly balanced (all 4 FSR values very similiar at the beginning) the results could improve. For example, if we had pedestals on each corner of the tile (just like a raised floor), the smart floor would be balanced and it also would help to spread the forces to the corners.

The 19% of the wrong results could have an explanation. Looking at table 4.1, we see that all the wrong results occurred while the subject was walking from N-S or S-N and the foot did not hit the centre of the surface (nr: 5, 6, 14, 18, 20, 21, 38, 39 and 42).

Considering the position of each sensor, we know that FSR2 and FSR3 were placed at the west and east side of the tile, respectively. Once we hadn't the FSR2 at the same level of the FSR3 (FSR2 was 20% lower than FSR3), it's comprehensive that, for example, when the subject hit the floor on the west side of the tile while walking from N-S, the output of the 4 sensors were influenced owing to the slope of the West side. Another interesting result is that we didn't have a single false positive result. In other words, when none of the fuzzy rules could calculate the direction, the output was "NONE". Finally, we believe that all the undetected directions could be calculated by adding new rules to the algorithm with, for example, higher values of threshold in order to include the tests number 5, 6, 14, 18, 20, 21, 38, 39 and 42.

In order to increase the number of tests, we asked for 3 volunteers to walk randomly over the smart floor. During the tests, we did our best to have a balanced floor and so the results could improve. Figure 4.11 illustrates the results from the 3 subjects. Table 4.2 also presents the ADC value registered in the calibration. At this point, we had better results comparing to the ones showcased above.

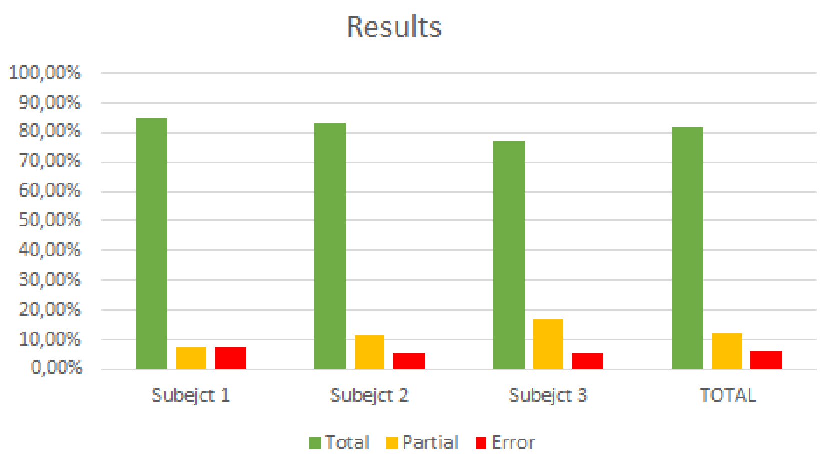


Figure 4.11: Results from walking randomly.

Table 4.2: Information related to the performed tests

	ADC value in calibration		
	Subject 1	**Subject 2**	**Subject 3**
sensor1	955	936	959
sensor2	981	939	950
sensor3	956	959	886
sensor4	998	1006	1006

weight	85 Kg	75 Kg	85 Kg
height	185 cm	186 cm	185 cm

		CLASSIFICATION RESULTS							
		W-E	W-E (S)	W-E (N)	E-W	E-W (S)	E-W (N)	NONE	Success Rate
GROUND TRUTH	W-E	86%	3%	3%	-	-	-	7%	93%
	W-E (S)	20%	80%	-	-	-	-	-	100%
	W-E (N)	25%	-	50%	-	-	-	25%	75%
	E-W	-	-	-	84%	8%	8%	-	100%
	E-W (S)	-	-	-	25%	75%	-	-	100%
	E-W (N)	-	-	-	14%	-	86%	-	100%
	S-N	-	-	-	-	-	-	-	-
	S-N (W)	-	-	-	-	-	-	-	-
	S-N (E)	-	-	-	-	-	-	-	-
	N-S	-	-	-	-	-	-	-	-
	N-S (W)	-	-	-	-	-	-	-	-
	N-S (E)	-	-	-	-	-	-	-	-

Table 4.3: Confusion Matrix - part I.

		CLASSIFICATION RESULTS							
		S-N	S-N (W)	S-N (E)	N-S	N-S (W)	N-S (E)	NONE	Success Rate
GROUND TRUTH	W-E	-	-	-	-	-	-	-	-
	W-E (S)	-	-	-	-	-	-	-	-
	W-E (N)	-	-	-	-	-	-	-	-
	E-W	-	-	-	-	-	-	-	-
	E-W (S)	-	-	-	-	-	-	-	-
	E-W (N)	-	-	-	-	-	-	-	-
	S-N	84%	6%	3%	-	-	-	6%	94%
	S-N (W)	17%	83%	-	-	-	-	-	100%
	S-N (E)	14%	7%	57%	-	-	-	21%	79%
	N-S	-	-	-	83%	9%	4%	4%	96%
	N-S (W)	-	-	-	-	75%	13%	13%	88%
	N-S (E)	-	-	-	-	-	100%	-	100%

Table 4.4: Confusion Matrix - part II.

Chapter 5

Conclusions and Future Work

This research proposes a low cost solution for improving the Indoor Location System accuracy. Our technique is based on a floor sensing system with four FSR sensors. The smart floor works as a position reference with direction estimator. At the moment the algorithm detects twelve different directions and orientations. We also developed a scalable solution that can easily be expanded to cover larger areas. In next section we summarize all the achievements of our project.

5.1 Achievements

Considering all the results of our solution we can affirm that, effectively, the smart floor can be used as a position reference and direction estimator. In the list bellow we enumerate the main achievements:

1. Low cost solution with a total price less than 100€;
2. Position reference width sub-one-meter accuracy;
3. Low energy consumption;
4. Real-time solution;

Associated to the algorithm we were able of:

1. Successfully detects more than 100 different footsteps from people with weights of 65 to 95 Kg and shoe size of 39 to 44 (Euro sizes);
2. When the subject was walking randomly, 81% of the results were correct, even with an unbalanced floor;
3. Distinguish on each side the foot hit the smart floor (centre, right or left);
4. Calculating the time on heel to time on toe;
5. Registering the exact time of the impact on the smart floor (according to NTP server);
6. Calculating the maximum force applied over the tile during the footstep (in ADC value);

5.2 Future Work

For future work, there are three improvements that could be done. First, we could expand the algorithm for detecting 8 possible directions. A potential solution is by generating more fuzzy rules. Secondly, we should develop a wireless communication, such as Bluetooth, for transferring the final outputs to the smartphone. Finally, we could install the final prototype into a raised floor tile. We would be able of testing the prototype in a real case scenario and also decrease the final budget since we would reuse the existing raised floor panel and pedestals. The figure bellow illustrates the idea.

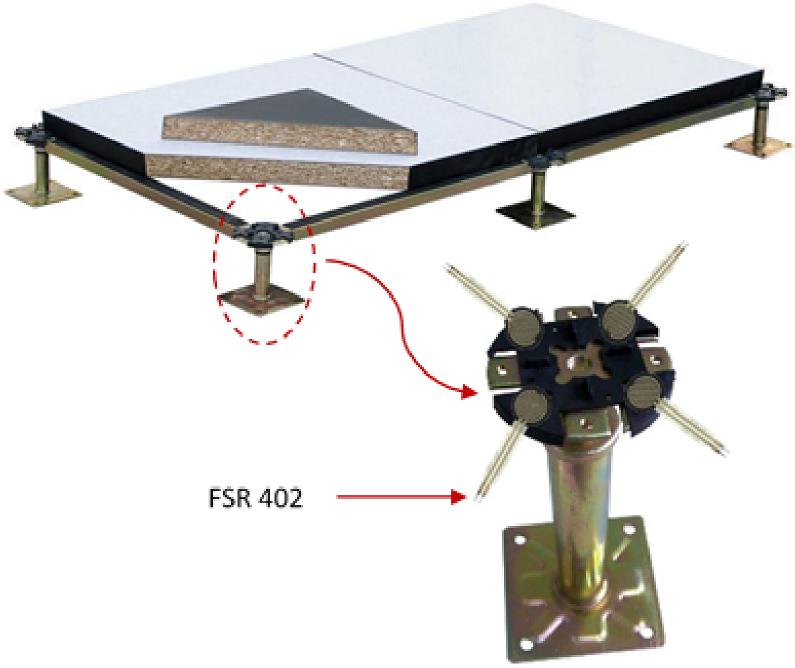


Figure 5.1: Concept of a raised floor with FSR 402 installed on the pedestals.

Bibliography

[1] U. E. P. Agency, “Report to congress on indoor air quality: volume 2.”, Washington, DC., Tech. Rep., 1989. [Online]. Available: http://nepis.epa.gov/Exe/ZyPDF.cgi/9100LMBU.PDF?Dockey=9100LMBU.pdf.

[2] Fraunhofer AICOS Portugal, *Portuguese researchers develop accurate indoor system*. [Online]. Available: http://www.fraunhofer.pt/en/fraunhofer%5C_aicos/news%5C_and%5C_events/news%5C_archive/portuguese-researchers-develop-accurate-indoor-system.html (visited on 02/14/2015).

[3] ——, *Fraunhofer portugal awarded at the microsoft indoor location competition*. [Online]. Available: http://www.fraunhofer.pt/en/fraunhofer%5C_aicos/news%5C_and%5C_events/news%5C_archive/fraunhofer-portugal-wins-3rd-place-at-microsoft-indoor-location-.html (visited on 06/15/2015).

[4] Microsoft, *Indoor localization competition - ipsn 2015 - microsoft research*. [Online]. Available: http://research.microsoft.com/en-us/events/indoorloccompetition2015/ (visited on 06/16/2015).

[5] D. Zhang, F. Xia, Z. Yang, L. Yao, and W. Zhao, “Localization technologies for indoor human tracking”, in *2010 5th International Conference on Future Information Technology*, IEEE, 2010, pp. 1–6, ISBN: 978-1-4244-6948-2. DOI: 10.1109/FUTURETECH.2010.5482731. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5482731.

[6] X. Li and K. Pahlavan, “Super-resolution toa estimation with diversity for indoor geolocation”, *IEEE Transactions on Wireless Communications*, vol. 3, no. 1, pp. 224–234, Jan. 2004, ISSN: 1536-1276. DOI: 10.1109/TWC.2003.819035. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1259415.

[7] C. Pearson, *Indoor location determination: taking a step back*. en, 2012. [Online]. Available: https://dspace.library.uvic.ca//handle/1828/4186.

[8] R. Wang, F. Zhao, H. Luo, B. Lu, and T. Lu, “Fusion of wi-fi and bluetooth for indoor localization”, in *Proceedings of the 1st international workshop on Mobile location-based service - MLBS ’11*, New York, New York, USA: ACM Press, Sep. 2011, p. 63, ISBN: 9781450309288. DOI: 10.1145/2025876.2025889. [Online]. Available: http://dl.acm.org/citation.cfm?id=2025876.2025889.

[9] R. Want, A. Hopper, V. Falcão, and J. Gibbons, "The active badge location system", *ACM Transactions on Information Systems*, vol. 10, no. 1, pp. 91–102, Jan. 1992, ISSN: 10468188. DOI: 10.1145/128756.128759. [Online]. Available: http://dl.acm.org/citation.cfm?id=128756.128759.

[10] C. Ciavarella and F. Paternó, "The design of a handheld, location-aware guide for indoor environments", *Personal and Ubiquitous Computing*, vol. 8, no. 2, pp. 82–91, May 2004, ISSN: 1617-4909. DOI: 10.1007/s00779-004-0265-z. [Online]. Available: http://dl.acm.org/citation.cfm?id=992039.992042.

[11] T. Shiraishi, N. Komuro, H. Ueda, H. Kasai, and T. Tsuboi, "Indoor location estimation technique using uhf band rfid", in *2008 International Conference on Information Networking*, IEEE, Jan. 2008, pp. 1–5, ISBN: 978-89-960761-1-7. DOI: 10.1109/ICOIN.2008.4472808. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4472808.

[12] MIT, *The cricket indoor location system: an nms project @ mit csail*, 2005. [Online]. Available: http://cricket.csail.mit.edu/ (visited on 02/14/2015).

[13] Zebra Technology Company, *Wherenet*. [Online]. Available: https://www.zebra.com/us/en/products/location-solutions/wherenet.html (visited on 06/28/2015).

[14] P. B. Padmanabhan and V. N., *Radar: an in-building rf based user location and tracking system*. [Online]. Available: http://research.microsoft.com/en-us/groups/sn-res/infocom2000.pdf (visited on 06/28/2015).

[15] Nissanka B. Priyantha, Anit Chakraborty and H. Balakrishnan, *The cricket location-support system*. [Online]. Available: http://www.eecs.harvard.edu/~mdw/course/cs263/fa04/papers/cricket-mobicom00.pdf (visited on 06/28/2015).

[16] D. F. João Moutinho, Rui Esteves Araújo, *Sound based indoor localization – practical implementation considerations*. [Online]. Available: http://ipin2013.sciencesconf.org/conference/ipin2013/eda%5C_en.pdf (visited on 06/29/2015).

[17] GAITRite Systems, *Portable gait analysis*. [Online]. Available: http://www.gaitrite.com/ (visited on 02/18/2015).

[18] Joseph Paradiso; Craig Abler; Kai-yuh Hsiao; Matthew Reynolds; *The magic carpet: physical sensing for immersive environments*, 1997. [Online]. Available: http://resenv.media.mit.edu/pubs/papers/97%5C_03%5C_CHI97%5C_Floor.pdf (visited on 02/16/2015).

[19] M. Addlesee, A. Jones, F. Livesey, and F. Samaria, "The orl active floor [sensor system]", *IEEE Personal Communications*, vol. 4, no. 5, pp. 35–41, 1997, ISSN: 10709916. DOI: 10.1109/98.626980. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=626980.

[20] L. Middleton, A. Buss, A. Bazin, and M. Nixon, "A floor sensor system for gait recognition", in *Fourth IEEE Workshop on Automatic Identification Advanced Technologies (AutoID'05)*, IEEE, 2005, pp. 171–176, ISBN: 0-7695-2475-3. DOI: 10.1109/AUTOID.2005.2. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1544421.

[21] J. A. Cantoral-Ceballos, N. Nurgiyatna, P. Wright, J. Vaughan, C. Brown-Wilson, P. J. Scully, and K. B. Ozanyan, "Intelligent carpet system, based on photonic guided-path tomography, for gait and balance monitoring in home environments", *IEEE Sensors Journal*, vol. 15, no. 1, pp. 279–289, Jan. 2015, ISSN: 1530-437X. DOI: 10.1109/JSEN.2014.2341455. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6861424.

[22] H. Kim and S. Chang, "High-resolution touch floor system using particle swarm optimization neural network", vol. 13, no. 6, pp. 2084–2093, 2013.

[23] R. N. A. Nunes, M. Piedade, "Cost effective immersive room with pressure sensing floor", *IMEKO 20th TC3, 3rd TC16 and 1st TC22 International Conference Cultivating metrological knowledge*, 2007. [Online]. Available: http://www.imeko.org/publications/tc16-2007/IMEKO-TC16-2007-071u.pdf.

[24] G. Qian, J. Zhang, and A. Kidane, "People identification using floor pressure sensing and analysis", *IEEE Sensors Journal*, vol. 10, no. 9, pp. 1447–1460, 2010, ISSN: 1530437X. DOI: 10.1109/JSEN.2010.2045158.

[25] S. Chang, S. Ham, S. Kim, D. Suh, and H. Kim, "Ubi-floor: design and pilot implementation of an interactive floor system", *Proceedings - 2010 2nd International Conference on Intelligent Human-Machine Systems and Cybernetics, IHMSC 2010*, vol. 2, pp. 290–293, 2010. DOI: 10.1109/IHMSC.2010.172.

[26] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems", *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 37, no. 6, pp. 1067–1080, 2007, ISSN: 10946977. DOI: 10.1109/TSMCC.2007.905750.

[27] InterlinkElectronics, "Fsr® integration guide & evaluation parts catalog with suggested electrical interfaces", 2010. [Online]. Available: http://resenv.media.mit.edu/classes/MAS836/Readings/fsrguide.pdf.

[28] Y. Tao, H. Qian, M. Chen, X. Shi, and Y. Xu, "A real-time intelligent shoe system for fall detection", *2011 IEEE International Conference on Robotics and Biomimetics, ROBIO 2011*, pp. 2253–2258, 2011. DOI: 10.1109/ROBIO.2011.6181633.

[29] DigiKey Electronics, *Fsr402 interlink electronics*. [Online]. Available: http://www.digikey.com/product-search/en/sensors-transducers/force-sensors/1966743?k=FSR%20402 (visited on 06/19/2015).

[30] Tekscan, "A201 flexiforce sensor", p. 4500,

[31] F. Vecchi, C. Freschi, S. Micera, A. M. Sabatini, P. Dario, and R. Sacchetti, "Experimental evaluation of two commercial force sensors for applications in biomechanics and motor control", *International Functional Electrical Stimulation Society IFESS*, pp. 4–7, 2000.

[32] R. Pallàs-Areny, *Sensor And Signal Conditioning - 2ed - Ramon Pallàs-Areny*. 2001. [Online]. Available: `http://pt.scribd.com/doc/49257926/Sensor-And-Signal-Conditioning-2ed-Ramon-Pallas-Areny%5C#scribd`.

[33] Texas Instruments, *Lmx24-n, lm2902-n low-power, quad-operational amplifiers*. [Online]. Available: `http://www.ti.com/lit/ds/symlink/lm124-n.pdf` (visited on 06/21/2015).

[34] Microchip, *Mcp3004/3008*. [Online]. Available: `http://ww1.microchip.com/downloads/en/DeviceDoc/21295d.pdf` (visited on 06/22/2015).

[35] Raspberry Pi Foundation, *Raspberry pi 2*. [Online]. Available: `https://www.raspberrypi.org/raspberry-pi-2-on-sale/` (visited on 06/22/2015).

[36] Intel, *Intel® edison*. [Online]. Available: `http://download.intel.com/support/edison/sb/edison%5C_pb%5C_331179001.pdf` (visited on 06/22/2015).

[37] Arduino, *Arduino uno*. [Online]. Available: `http://www.arduino.cc/en/Main/ArduinoBoardUno` (visited on 06/22/2015).

[38] Beagleboard, *Beagle bone black*. [Online]. Available: `http://beagleboard.org/BLACK` (visited on 06/22/2015).

[39] Raspbian, *Raspbianabout*. [Online]. Available: `https://www.raspbian.org/RaspbianAbout` (visited on 06/23/2015).

[40] Gordon Henderson, *Wiringpi*. [Online]. Available: `http://wiringpi.com/` (visited on 06/23/2015).

[41] Sublime Text, *Text editor*. [Online]. Available: `http://www.sublimetext.com/` (visited on 06/24/2015).

[42] Cygwin, *Functionalities similar to a linux distribution on windows*. [Online]. Available: `https://cygwin.com/` (visited on 06/24/2015).

[43] Putty, *Ssh and telnet client*. [Online]. Available: `http://www.putty.org/` (visited on 06/24/2015).

[44] WinSCP, *Free sftp and ftp client for windows*. [Online]. Available: `http://winscp.net/eng/index.php` (visited on 06/24/2015).

[45] R. J. Orr and G. D. Abowd, "The smart floor : a mechanism for natural user identification and tracking", *Conference on Human Factors in Computing Systems,*, pp. 275–276, 2000. DOI: `10.1145/633292.633453`.

[46] MathWorks, *Matlab*. [Online]. Available: `http://fr.mathworks.com/help/matlab/learn%5C_matlab/product-description.html` (visited on 06/25/2015).