University of Modena and Reggio Emilia
Department of Engineering “Enzo Ferrari”

PhD Dissertation for the XVIII Cycle of the Doctorate School in Information and Communication Technologies

“Human Environment Interaction with IoT Sensing Floor Devices”

Candidate:

Martino Lombardi

Tutor:

Prof. Rita Cucchiara

Director of the School:

Prof. Giorgio Matteo Vitetta

"Hell is paved with good intentions."

Samuel Johnson (1709 - 1784)

Abstract

In recent years, the research on sensing environments has received considerable interest, allowing a plethora of intelligent services to help people in their everyday life. Intelligent building automation, e-health, surveillance, entertainment, smart factories are only few examples of applications which can benefit from this technology.Different kinds of sensing floors have been proposed in the past, but they are conceived for specific applications and often very expensive and difficult to install.
In this thesis, we describe the design and implementation of the "Florim Age" technology: a new, versatile, scalable and cost-effective interface based on sensing floors compliant with the Internet of Things (IoT) approach.
All the components of the proposed framework are described and presented, from the hardware device, which captures the pressure field variations generated by moving people, to the high level applications. Algorithms of floor data processing, communication protocols, and system architecture details are also defined.
In particular, three different application types have been addressed: real-time surveillance, human-computer interaction, and zone statistics. In the first case, we proposed and tested algorithms for action recognition and human behavior understanding. In the second case, the floor has been used as input device for games and interactive applications. Finally, we exploited the sensing floor technology to develop people counting and zone statistics systems. All these applications have been deeply tested on both laboratory and public installations.

Acknowledgements

This work has been supported by Florim Ceramiche S.p.A. (Italy) and was done within the Regional Operational Programme POR FESR 2007-2013 of Softech-ICT. The foot painting and the edutainment applications have been developed in collaboration with the Italian Company Vision-e s.r.l. We finally thanks Dr. Federico Tasso (Accademia Belle Arti, Brera, Milan) for the prototype of the Street View application.

Contents

List of Figures

List of Tables

List of Abbreviations

CNC	**C**omputer **N**umerical **C**ontrol
COP	**C**enter **O**f **P**ressure
ESD	**E**lectro**S**tatic **D**ischarge
GPRS	**G**eneral **P**acket **R**adio **S**ervice
GSM	**G**lobal **S**ystem for **M**obile Communication
HCI	**H**uman-**C**omputer **I**nterfaces
IEs	**I**ntelligent **E**nvironment**s**
IoT	**I**nternet **o**f **T**hings
LTE	**L**ong **T**erm **E**volution
NFC	**N**ear **F**ield **C**ommunication
PET	**P**oly**E**thylene **T**erephthalate
PCB	**P**rinted **C**ircuit **B**oard
PIR	**P**assive **I**nfra**R**ed Sensor
PVDF	**P**oly**V**inyli**D**ene **F**luoride polymer
RFID	**R**adio **F**requency **ID**entification Tags
SFs	**S**ensing **F**loor**s**
3G	**3**rd **G**eneration
Ubicomp	**Ubi**quitous **comp**uting

This thesis is dedicated to my parents, my brothers, and Norma for their love, support and encouragement.
In loving memory of Luigina.

Introduction

According with the "Internet of Thing" (IoT) paradigm [3], the technological and cultural evolution of the last few years has led to an increasing presence around us of sensing devices. The environment where we live is increasingly scattered with sensors able to improve the human-environment interaction and to provide distributed surveillance and monitoring.
Among different types of sensors, in this thesis we will focus on the exploitation of sensing floors. Their applications are manifold in several fields, including both public and private environments. For example, smart buildings can include sensing floors to detect the presence of people and to automatically switch on/off the lighting or the heating systems.
In the e-health field, these devices can be used to detect dangerous situations such as an elder falling or getting out of his/her bed [42]. Furthermore, sensing floors can be used for people counting or to monitor crowd movements during public events, exhibitions, and so on. In comparison with other traditional technologies such as video cameras, sensing floors provide less information but

have two undoubted advantages. First, they are completely privacy compliant. In fact, it is not feasible to recognize and identify users from their data only. Thus, the installation of these sensors is also allowed in private places, such as toilets or bedrooms. Second, sensing floors data are not affected from occlusions, a typical issue of visual camera systems. We believe that smart floors could be a reliable solution, complementing or substituting other technologies (e.g., cameras or depth sensors).
In this work, we discuss about the technical requirements which we consider indispensable to obtain a reliable solution and which led us to the development of an innovative sensing floor. Hardware, architectural, and software issues have been handled during all development and testing steps of this research project, with a close cooperation between the Imagelab laboratory of the University of Modena and Reggio Emilia and Florim Ceramiche SpA, an Italian ceramic Company located in Fiorano Modenese (Modena, Italy).
This work has been partially funded by Florim Ceramiche SpA under the contract "FLORIMAGE: the floor is an image", started in September 2012, within the project funded by Emilia-Romagna Region in the "Bando Ricerca, Innovazione e Crescita" (2013-2015). After the end of the contract, the collaboration has been continued with a Research Agreement between Florim Ceramiche SpA and the Interdipartimental Center Softech-ICT for the establishment of the "Florim Lab" in the Technopole of Modena.

The thesis is structured as follows: in the next chapter we introduce the IoT paradigm and, after discussing its economic and strategic impact in modern society, we will describe the main characteristics of an IoT device. In the same chapter, we introduce the Sensing Floor devices and we will describe some related works of other experiences and prototypes. The chapter underlines that these proposals are very interesting research prototypes but none of them represents a good solution to be integrated in Intelligent Environment projects for Human-Environment Interaction.
In chapter 2, we provide an overview of an innovative sensing floor technology, compliant with the IoT paradigm and completely designed and developed in our laboratories, giving details on its hardware components and on physical processes exploited by the sensor. This result is obtained adding a sensing layer below ceramic tiles. The solution proposed is cheap enough to allow the coverage of wide areas with sensing elements which do not change the design or the appearance of the floor. In addition, the data collected from the sensors do not contain identifying elements, assuring an high respect of the user privacy. Following the IoT paradigm, the sensing elements communicate with the physical world by means of a on board electronic acquisition/communication unit.
In chapter 3, taking advantage of the technical characteristics of the proposed sensing floor technology, a hierarchical system architecture as well as an ad-hoc communication protocol were proposed

to obtain a scalable and reliable solution.
The low-level data processing system is described in chapter 4. This process is inspired by common computer vision and image processing algorithms and is applied on the data captured from the sensing floor. As a matter of fact, thanks to the regular distribution of the sensors below the floor, the processing unit generates a pressure image, in which each sensor corresponds to a pixel and the spatial neighborhood relations are preserved. Thus, the pressure value is an excellent feature to detect objects on the floor and to discriminate between noise, people and objects. Suitable algorithms for both people detection and tracking have been proposed and tested. Moreover, with the aim to recognize people actions and behaviors using floor data only, Machine learning algorithms have been applied. In particular, SVM and Random Forest classifiers have been exploited to recognize some people actions such as jumps and falls. Since the image processing algorithms above presented requires a grid distribution of the sensors, we also studied an alternative method based on center of pressures to extract high-level information, which we describe in the same chapter.
In chapter 5, we present some use cases where the system has been used as input device. In particular, three use cases are reported: two *Surveillance* applications in which the floor is used for automatic

detection of intruders and people falls, three *Human-Computer Interaction* applications in which the floor is used as gamepad controller for interactive applications, virtual reality and edutainment games, and a *Zone Statistics* application in which the floor data collection and analysis allows to characterize people movements and common flows, for example in commercial or cultural heritage expositions.

The ease of installation and configuration are essential features for the diffusion of sensing floor systems, specially when the number of installed modules grows. However, most of the solutions proposed in the past have two drawbacks. First, an individual calibration phase is required for each sensing element, because its response depends both on intrinsic characteristics and on installation issues. For example, the presence of ceramic tiles above the sensors, the morphology and the flatness of the layer under the sensors may affect the response level and range of each element. Secondly, a manual configuration of the setup should be provided to correctly specify the location of each module within the whole sensing area. More specifically, the spatial transformations which convert coordinates from the local coordinate systems of each module to a global and common coordinate system are needed. For these reasons, in chapter 6 we propose an automatic method to calibrate and discover the layout of sensing units solely based on acquired data.

Chapter 1

Internet of Things and Sensing Floors: an overview

1.1 What Is the Internet of Things (IoT)

The Internet of Things is becoming a hot topic in research and industry but it is not a new concept. It has its roots in the "*Ubiquitous Computing*" concept (also known as "*Ubicomp*") introduced by Mark Weiser in 1988 at the Computer Science Lab at Xerox Palo Alto Research Center.
This paradigm was introduced to describe a future in which a general purpose machine (the PC) will be replaced by a large number of specialized computers, which are embedded into everyday objects.
In one of his seminal articles Weiser [61] suggests that, due to the

trend of unobtrusive technology and more intrusive information, the Ubicomp technology will develop exponentially fast. He states that: *in the long run, the personal computer and the workstation will become practically obsolete because computing access will be everywhere: in the walls, on your wrist, and in "scrap computers" (like scrap paper) lying about to be used as needed. Hence, ubiquitous computing.*
At the time, this idea might seem visionary and it required major technological improvements. Today, since the size, cost and computing power of microcontrollers and microprocessors has dropped drastically, this vision became reality. There is an huge number of possible physical objects that might contain computing technologies: electrical appliances, digital measuring devices, smart homes, smart cars and so on.
However, in his work Weiser doesn't assume that there will be network connectivity but he only dealt with the question of how users would interact with an environment that is physical, but enriched with computing functionality. As he writes: *Ubiquitous Computing has as its goal the nonintrousive availability of computers throughout the physical environment, virtually, if not effectively, invisible to the user.*

Kevin Ashton, the co-founder of the Auto-ID Center of the Massachusetts Institute of Technology (MIT) is usually credited with coming up with the term IoT in 1998 [45].

At the time, Ashton was focused on the creation of a global standard system for Radio Frequency Identification (RFID) tags, when he had the idea of objects that could sense, communicate and share information, all interconnected over a public or private Internet Protocol (IP) networks.
Since then, the term IoT is used to refer to the general idea of things, especially everyday objects, that are readable, recognisable, locatable, addressable, and/or controllable via the internet, irrespective of communication means (*e.g.* via RFID, wireless LAN, wide-area networks or other means). Everyday objects includes not only the usual electronic devices we encounter or the products of higher technological development such as vehicles and equipment but things that we do not ordinarily think of as electronic at all - such as food and clothing.

As depicted in Figure 1.1, in 2012 the number of everyday physical objects and devices connected to the internet was around 8.7 billion. But Cisco's Internet Business Solution Group (IBSG) [17] predicts there will be over 50 billion connected devices by 2020 (about 6.58 connected devices per person). In this Scenario, the impact that IoT brings to our daily lives become more prevalent. New services and tools can emerge to address society and to help people in everyday life [21]. For example, online devices can be worn by people to track location, health, or fitness record,

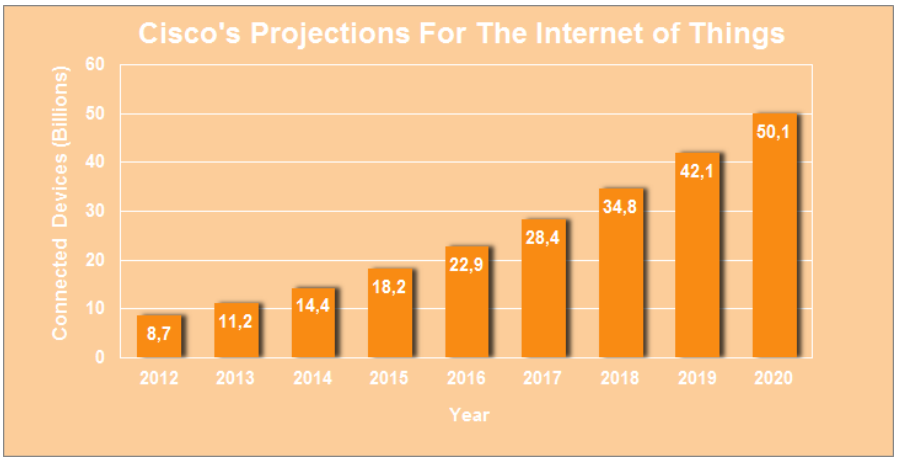


FIGURE 1.1: Number of connected objects expected to reach 50 billions in 2020.

similar to most wearable worn today. These devices can be used to monitor children, employees, or elderly people which may require special assistance. Furthermore with an advanced GPS management system with IoT integrated into smart vehicles, traffic and park services applications can be developed to help people to make the best routes to work or to the nearest free parking area. LCD monitors can be installed at bus stops to provide travel information from buses that are fitted with GPS capabilities alongside various other sensors which transmit data about the vehicle's location and current progress. Environmental IoT applications can be developed to use sensors to monitoring air or water quality, atmospheric or soil conditions and so on.

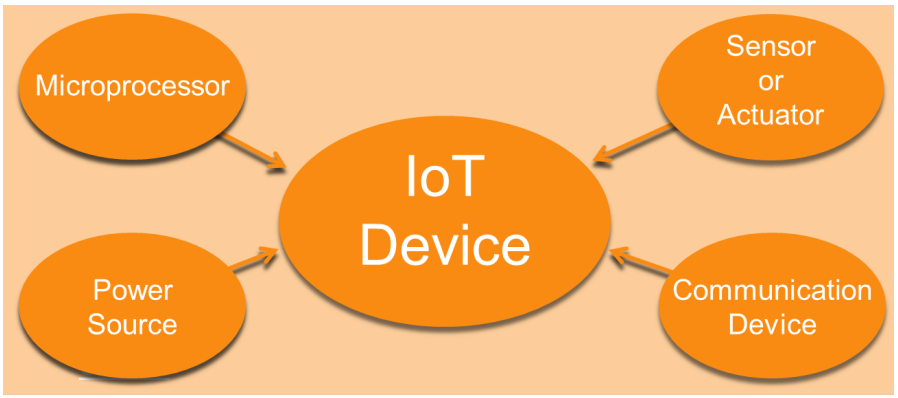


FIGURE 1.2: IoT device components: sensor or actuator, microprocessor, communication device, and power source.

The above examples are just a small sample of what you can do with the introduction of the IoT paradigm, but we can recognize some common characteristics that can be used to define an IoT device. These can be summarized as follow: an IoT device is a physical object equipped with a form of *sensor or actuator*, a *microprocessor*, a *communication device*, and a *power source* (see Figure 1.2) [56].

The sensor or actuator gives to the object the ability to interact with the physical world. An IoT device can possess one or more sensors to collect surrounding information, record it, forward it or react directly to it. Each sensor can monitor a specific condition such as location, vibration, motion or temperature. An IoT device can also

contain actuators to manipulate their environment (for example by converting electrical signals into mechanical movement). Such actuators can be used to remotely control real-world processes via the Internet.
The microprocessor (or microcontroller) plus a storage capacity enables the object to transform the data captured from the sensors, although at a limited speed and at limited complexity. These resources can be used, for example, to give products a "memory" of how they have been used.
The communication device enables the object to communicate its sensor readings to the outside world and receive input from other IoT devices. In the IoT paradigm, objects have the ability to network with Internet resources or even with each other, to make use of data and services and update their state. IoT devices can use various types of local area connections such as RFID, NFC, Wi-Fi, Bluetooth, and ZigBee. They can also have wide area connectivity such as GSM, GPRS and LTE. Within an IoT network, objects must uniquely identifiable. RFID or optically readable bar codes are examples of technologies with which even passive objects (with no built-in energy resources) can be identified (with the aid of a "mediator" such as an RFID reader, tablet or smartphone). Identification enables objects to be linked to information associated with the particular object and that can be retrieved from a server, provided the mediator is connected to the network.

The power source provides the electrical energy for the object to do its work. In addition to traditional power sources, IoT devices often make use of energy harvesting processes to capture and store energy from ambient sources - including light, vibration and temperature changes - and to convert this energy to electricity.

1.2 Intelligent Environment Systems and Sensing Floors

Advances in IoT and smart sensor technology have facilitated the rapid growth of the research activity on Intelligent Environments (IEs). Research in this area takes on different labels and terms such as intelligent environments, smart spaces, smart homes, and ambient intelligence are often used as synonyms; IEs are living (public or private) spaces with embedded sensors that sense and effectors that react to occupants. The occupant need not wear or carry a computing - capable device, although wearable devices may augmented and provide context awareness [30].
As proposed in [4], in the following we report a list of characteristics that every intelligent environment should aspire to have:

- to be intelligent to recognize a situation where it can help.
- to be sensible to recognize when it is allowed to offer help.

- to deliver help according to the needs and preferences of those which is helping.
- to achieve its goals without demanding from the user/s technical knowledge to benefit from its help.
- to preserve privacy of the user/s.
- to prioritize safety of the user/s at all times.
- to have autonomous behaviour.
- to be able to operate without forcing changes on the look and feel of the environment or on the normal routines of the environment inhabitants.
- to adhere to the principle that the user is in command and the computer obeys, and not viceversa.

These principles summarize the aims of this research area and many of these achievements can be obtained with the integration between an intelligent environment with a sensor devices IOT network. For this scope, a wide range of sensors can be adopted with varying capabilities, allowing, for example, the measurements of:

- position, direction, distance and motion,
- light, radiation and temperature,

- solids, liquids and gases,
- identification information, including biometric data,
- sound, and images

They provide a variety of different inputs, that can range from a simple on/off value, to values in numerical ranges (e.g., temperature, and weight of a person), to richer data like fingerprints, sound, pictures and video [13].
In Figure 1.3 a common working schema is depicted in which the state of each IE is provided by a variety of distributed sensors and the flow of the acquired data is channeled to the main system through a wired or wireless network. Then, all the flux of information is merged in real-time by a layer of software commonly referred as *middleware*.This layer presents the information in a way that can be useful to other higher decision-making modules of the system and manages eventual incomplete or corrupt information coming from malfunctioning sensors. The middleware of a IE system facilitates interoperability between different parts of the system (sensors,actuators, devices, network, etc.) created by different providers to understand each other and converge into a representation that is understood by software at upper levels of the system architecture [44].
In addition to those presented in Section 1.1, the IoT sensors

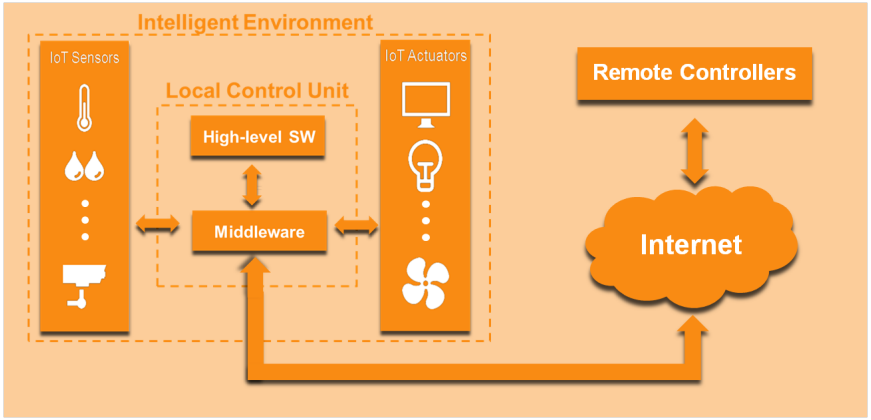


FIGURE 1.3: Work schema of a generic Intelligent Environment. Iot Sensors devices and actuators controlled via e middleware software service embedded into a local processing unit and remotely via internet.

devices used in this scenario must have other specific characteristics. For example, it is often required an high scalability for these devices, because the sensors should be integrated into a hierarchical network and they must also be invisible to the users and must preserve their privacy.

In addition to the most commonly used type of sensors in IEs projects, such as smart cameras [36], temperature sensors [37], pressure sensors [52] or a combination of heterogeneous sensors [5, 53, 10], Sensing Floors (SFs) are recently attracting the interest of the research community [47, 31, 18].

Simplifying its structure, a sensor floor can be seen as a pressure measuring system composed of a protective surface, made of (wood, plastic, ceramic, etc.) tiles, and a grid of pressure sensors below it (see Figure 1.4). Often the term sensing floor is also referred to common floors provided with proximity sensors such as Passive InfraRed sensors (PIR) or with sensors that exploit optical techniques. These can be used, for instance by illuminating translucent floorboards with IR from below and inferring range from detected intensity reflecting off the foot. While this can also measure the foot when it is above the floor, it requires calibration for variations in sole reflectance and floor transparency (which can change with time), and does not directly provide pressure signals.

In the past, many different types of measuring techniques were used to create SFs prototypes and in Section 1.3 we will introduce some of these setups. For example, SFs can use large-area force-sensitive resistors that respond to foot pressure. These can be fragile, however, and difficult to transport for mobile installations. Moreover electric fields can be used, either measuring the change in capacitance between two plates sandwiching an insulator that compresses with pressure, directly measuring the loading of a capacitive electrode by the body when a foot is nearby, or measuring the coupling of an external signal sent from the shoe into a receptor electrode on the floor. But each techniques has certain advantages and disadvantages and we will try to analyze them. However, none

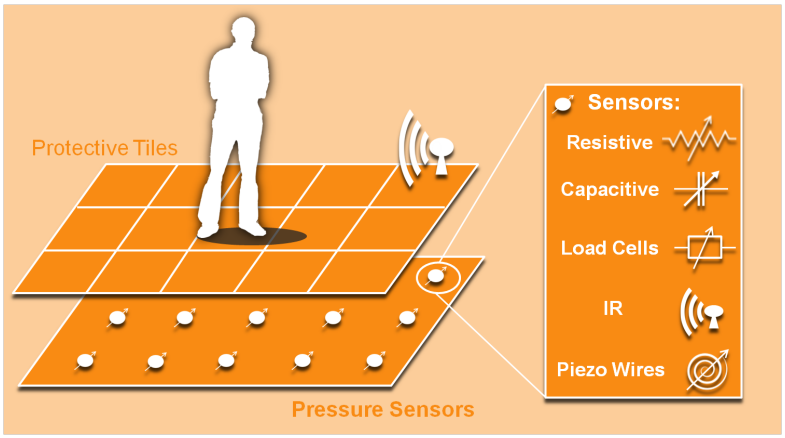


FIGURE 1.4: Work schema of a generic Sensing Floor. A SF can be seen as a pressure measuring system composed of a protective surface, with (wood, plastic, ceramic) tiles, and a grid of pressure sensors below it.

of them fulfills all the following requirements, which are mandatory to generate a reliable solution to an Intelligent Environment:

- *low cost*: the cost of the sensing elements should be comparable to traditional floors;
- *high scalability*: the sensors should be integrated into a hierarchical network, in order to allow the coverage of narrow rooms as well as wide areas;
- *transparent design*: for design issues, the sensing layer must be invisible to the users and the floor should appear like traditional floors;

- *high reliability and durability*: breakable and fragile elements should be avoided or limited to protected packages;
- *temporal and spatial resolutions*: they should be high enough to allow people detection and tracking, even in presence of multiple targets.

1.3 Related Works

Some prototypes of sensing floors for human-based action detection and identification have been designed in the past. They exploited different physical characteristics, and especially the pressure as measurable quantity and the proximity effect related with the electrical properties of human body. In the following, we provide a brief overview of some previously developed prototypes, underlying their differences and their capabilities in terms of some evaluation parameters: the cost, the scalability, the modularity, and the generality of the architecture. A summary is provided in the Table 1.1.

In 1997 J. Paradiso, from MIT, proposed the famous *Magic Carpet* [32] and it was the first example of a sensing floor. The system was designed to be used in an audio installation, where users could launch and modify complex musical sounds and sequences

by wandering about the carpet. A 16 x 32 grid of piezoelectric wires, placed below a 6 x 10 foot trapezoidal segment of carpet, running across the carpet at a roughly 4" inter-wire pitch, was used to sense foot pressure and position. Although this wire appeared as standard RG-174, 0.1" diameter, shielded, coaxial cable, the insulation between center and outer conductors was made from PVDF (polyvinylidene fluoride) polymer, which is a commonly available piezoelectric material, often used as a transducer. This cable will thus produce a voltage when pressed or flexed anywhere along its length. The signal from each PVDF wire under the carpet was buffered by a high-impedance operational amplifier, and a pressure profile was produced by a simple diode/capacitor envelope detector. This interactive environment was completed with the use of a pair of Doppler radar motion sensors to tracking movement of user's arms and upper body. Even if the system is characterized by a large sensing area and frame rate, it present a poor sensor densities and it is not modular.

The ORL active floor [1], designed by Oracle Research Lab, used a large 1.5 m x 1.5 m plate which was balanced on a 4 x 4 array load cells. Using a hidden Markov model they showed a degree of recognition of individuals. However, this project was primarily intended for tracking people, using the information on the distribution of the vertical ground reaction force measured

over the entire area of the floor. When only an object or a person is in contact with the floor, its center of pressure correspond with its position, but this is no longer true when two or more objects are in contact with the sensing surface. This drawback, together with a poor modularity and temporal and spatial resolution of the used sensors, make this system not very useful for the development of a modern intelligent environment.

Another FS system is the Z-Tile [41], designed by the University of Limerick Ireland and MIT Media Lab. The system uses modular nodes connected together to create a pressure sensitive area of varying size and shape, giving it the potential to be integrated into an IE. Each node, known as a Z-Tile, has twenty force sensitive resistors (FSR's) on its surface (0.5 sensors per square inch). The nodes themselves have an hexagonal shape, which allows them to interlock with each other in a regular pattern, and to be self-holding. Each hexagon on the tile surface, even those that interlock, has one FSR for reading pressure values, so there are no unsensed areas at tile joints. When the tiles are being used in a floorspace, the pressure sensors are protected by a 2 mm layer of plastic material that covers the surface of each tile. The system have the advantage of having a modular design, and, when joined together, the nodes form a flexible, pixellated, pressure sensing surface, with a high frame rate, but a low sensor density. The major drawback it's related with the

impossibility to hide the system to the users.

Lee Middleton [29] developed a floor sensor mat specifically conceived for gait recognition. The sensor is made of a set of binary switches and uses a simple design inspired by computer keyboards. Each switch is made by separating a pair of wires with a deformable foam material. Wires come into contact and the switch is closed when a pressure is applied. These switches are disposed on a grid and their state is read using the same approach of computer keyboards. The design is simple and easily scalable, even if the adopted switches are binary and they do not provide a response commensurate with the pressure.

A portable high-resolution sensing floor prototype was proposed by Prashant Srinivasan [50]. The sensor the developed measures the pressure field generated by people walking on it. The system consisted of several sensor mats, each one is composed of a 42x48 grid of pressure sensors with size of 48.8 cm x 42.7 cm. The system was entirely built employing off-the-shelf components. The sensor elements of the mat were made using a pressure sensitive polymer between conductive tracks on sheets of Mylar and they changed the resistance with the applied pressure. On the one hand the proposed architecture was suitable for research-oriented application but, on the other hand, it did not show the required scalability and it was not cheap enough to allow the coverage of

wide areas. A high number of expensive components were used for each mat,as well as a multi-sensor element (model-5315 by Tekscan [1]) and a Rabbit Ethernet-enabled controller.

Subsequently, Chen-Rong Yu [65] proposed a localization system to accurately estimate human's positions. It performs single person and multiple people tracking in a home environment. The Condensation algorithm is exploited to locate residents' position via multi-camera and sensory floor approaches. This work, based on a sensor floor and four video cameras, is almost completely focused on the proposed data processing techniques which are thoroughly explained. Conversely, no details concerning the nature of the forty pressure cells employed in the sensing floor are provided, neither on the architecture of the system used to collect the sensor data, nor on the size of the sensing area.

In their study, Domnic Savio [43] described three processing techniques to track the gait of human walk. They developed a smart carpet that can be laid on the floor. The sensor set forms a self-organizing sensor network. To identify the footstep, clustering algorithms based on Maximum Likelihood Estimate and Rank Regression have been applied. The proposed approach is scalable and commercially viable even if the binary nature of the embedded

[1] http://www.tekscan.com/5315-pressure-sensor

nodes does not provide a response commensurate with the applied weight. The authors developed a 240 cm by 200 cm smart carpet composed of 180 interconnected capacitive nodes forming the sensor network. Each node is equipped with a 16 bit micro-controller. This solution is very innovative and smart though, at the same time, very expensive. Besides, it does not seems to be not reliable and durable enough, due to its complexity and fragility and to the lack of a protective layer, being directly exposed to the human activities. As well the invisibility requirement is not fulfilled.

Yu-Lin Shen [46] developed a distributed sensing floor using an optical fiber sensor with a spatial resolution around 18 cm. The described method is based on Brillouin Optical Correlation Domain Analysis (BOCDA) technology, which detects the strain deviation along a fiber caused by pressure events. Main drawbacks of the proposed system are the complexity of the equipment used to detect the Brillouin frequency shift and the lack of spatial scalability.

A further step was made by the research of Miika Valtonen [54] in which an unobtrusive two-dimensional human positioning and tracking system based on a low-frequency electric field was described. The capacitance between multiple floor tiles and a receiving electrode was measured. Their method was based on the

fact that the human body is able to conduct a low-frequency signal. The authors did not handle, however, problems related to the large scale deployment, nor to the scalability of the hardware architecture. The proposed method, in fact, was based on the capacitive floor and did not provide any information related to the human weight. The lack of weight information represents a drawback to perform human action detection blocking any possibility to detect non-conductive objects, regardless of the weight.

Afterwards, Jan Anlauff [2] presented a prototype of a floor surface based on sensing elements made out of conductive black art paper and grouped into modules forming a grid of resistors able to measure quasi-static forces. The suggested approach was a low cost alternative for spatially resolved tactile sensing. However the employed signal conditioning system was based on the matrix arrangements of the sensing elements thus the proposed solution suffered from mutual interference between different sensors in the matrix. The authors don't provide a detailed description of the scalability of the communication infrastructure; moreover, the invisibility and the physical implementation requirements were not considered.

A probabilistic approach to the tracking and estimation of the lower body posture was presented in the work by Rishi

Rajalingham [38]. Their sensing floor had limited sizes, it was not easily scalable and commercially viable. It employed off the shelf resistive force sensors and consisted of a 66 array of rigid tiles, 30 cm on each side. Tiles were equipped with four resistive force sensors, which were located at the corners. An array of six small-form-factor computers was used for data processing. The studied solution was focused on data processing and the exploited sensing floor did not seem to be feasible for creating wide area implementations. Several reasons can be mentioned, *i.e.* the employed sensing elements are too expensive (several USD per piece), unfeasible for large scale applications; there is lack of details regarding modularity, scalability and physical implementation.

Finally, Ruben Vera-Rodriguez [57] presented a work focused on footstep signals as a biometric with a comparative analysis and fusion of spatio-temporal information of the signals for person recognition. The described experiments were carried out on a footstep database. The contribution of this work is the assessment of footsteps in time, in space, and in a combination of the two. The performance obtained for the two domains showed error rates in the range of 5-15% for each domain, and in the range of 2.5-10% for their fusion. The sensor arrangement used by the authors is based on two (45 x 30 cm) sensor mats. Each mat is equipped with 88 piezoelectric sensors and the sampling frequency for each sensor

is quite high, 1.6 kHz. Piezoelectric sensors need sophisticated conditioning hardware and they are expensive, and they need high sampling rates to exploit the dynamic nature of their electrical response. The work does not provide any details related with the physical architecture of the presented system and its scalability.

As shown Table 1.1, even many of these proposals are very interesting research prototypes which could be integrated in multi-sensing intelligent environment. However, none of them represents a good solution for covering large areas with a scalable and low cost technology.

TABLE 1.1: Sensing Floors

Authors	**Application Technology**	**Low Cost**	**Scalability**	**Invisibility**	**Reliability**	**Temporal and Spatial Resolution**
[32]	Piezoelectric Wires	n	n	n	y	y
[1]	Load Cells	n	n	n	n	n
[41]	FSR	y	y	n	n	n
[29]	Custom Switch Metrics	y	n	n	n	y
[50]	off-the-shelf FSR sensor based mat	n	n	n	y	y
[43]	custom capacitive textile based	n	y	n	n	y
[46]	Optical Fibers	n	n	n	n	n
[54]	custom capacitive	n	n	y	y	n
[2]	custom "Paper FSR"	y	n	n	y	y
[38]	off-the-shelf FSR	n	n	y	n	y
[57]	piezoelectric	n	n	n	y	y

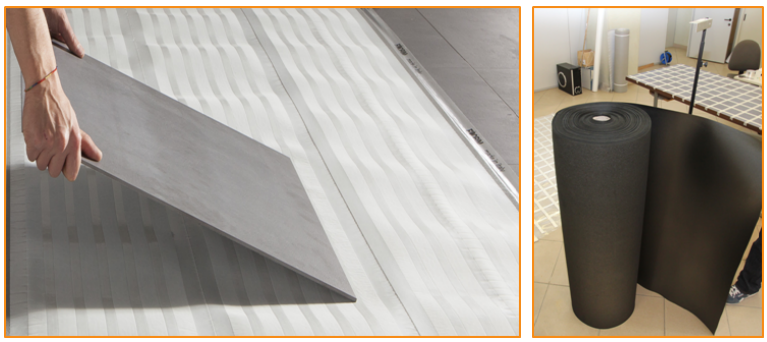
Chapter 2

Florim Age: an IoT Sensor for IE Applications

2.1 The Sensing Floor Device

Despite the development of several sensing floors as detailed in Chapter 1, none of the proposed solutions completely fulfills the mandatory requirements needed to integrate these smart sensors in intelligent environments. Therefore, a collaboration between the University of Modena and Reggio Emilia and Florim Ceramiche SpA [1] has been started aiming at developing a new solution, called Florim Age, which bridges the technological gap mentioned before. The basic idea behind this innovative sensing floor is to modify the already commercialized floating floors in order to obtain sensorised

[1] Florim Ceramiche SpA is an Italian company specialized in production of ceramic tiles; http://www.florim.it



(a) (b)

FIGURE 2.1: The Slim4/+ floating floor: (a) the tiles structure allows the products to be installed without adhesive. Tiles can be applied on pre-existing floors using a Velcro hooking membrane and a sealant for grouting the joints. (b) A roll of the conductive polymer used to replace the traditional hooking membrane in the Florim Age System.

versions retaining, as far as possible, features like ease of installation, practicality, functionality, and low cost of production.
A floating floor does not need to be nailed or glued to the sublayer and, thus, it might be constructed over a sub-floor or even over an existing floor. It consists of a polymeric layer holding up the tiles. Therefore, every tile can move perpendicularly to the floor plane and independently from its neighbors, so it can transmit pressure to the sublayer due to the presence of weight on it. The VELCRO® system is employed to keep the tiles connected to the floor and a sealant closes the joints between the ceramic slabs.

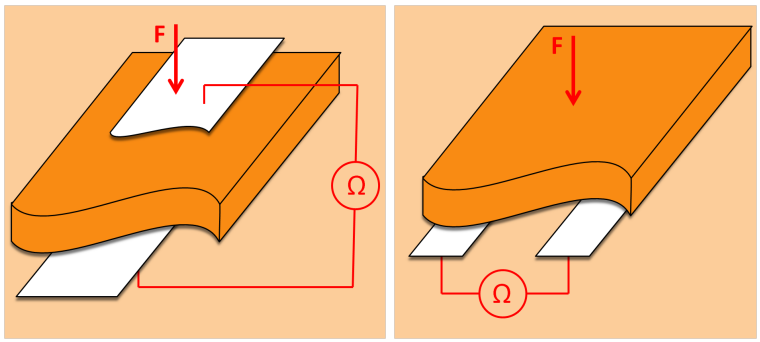
To fulfill the requirements detailed in the previous chapter, the ceramic covering follows the same process of the commercial product, while the polymeric coat has been replaced with a sensing layer (see Figure 2.1). In particular, superficially the module is covered using SLIM/4+ tiles produced by Florim. Three different versions of the Florim Age sensor module have been developed, to which here in after we refer as FA-v1, FA-v2 and FA-v3. Each sensor module has a rectangular shape and is composed of a sensing pad and a capture board. In particular, the three versions differ for the sensing pad part. Although all of them exploits the conductive layer described in 2.1.1, they differ for the particular technique utilized to physically connect the layer to the capturing board.

2.1.1 The Sensing Layer

Resistive tactile sensor systems exploit the changes of the interface resistivity between two surfaces due to the applied load. This effect was discovered by Theodore du Moncel in the late 19th century. He discovered, that an electrical current flowing between a sooted metal plate and a nail is modulated by acoustic waves. Based on this conclusion, he invented the carbon microphone which revolutionized telephony [15].
Today, in resistive tactile sensors the sooted metal plate can be replaced with conductive polymers. The load-dependent change in resistivity can be acquired using an electrode matrix and the

electronics is conditioned by a simpler signal compared to that of capacitive sensors. Plotting the measured electrical resistance as a function of the applied load on a resistive tactile sensor system, a hyperbolic trend can be observed. This nonlinear behavior can be used in collision detection, since for lightweight contacts to its surface the sensor is more sensitive than at high loads. Moreover, resistive tactile sensors are generally very robust on overpressure, shocks and vibrations.

The electrodes for measuring the resistance can be positioned in two different arrangements: on opposite sides of the conductive layer (Figure 2.2(a)) or on the same side (Figure 2.2(b)). In both cases, a compression of the electrodes on the conductive layer changes the resistance between them.
The first arrangement was used for the FA-v1 version of the Florim Age sensors, since it enables an easier construction of the sensor matrices. A grid of force sensitive resistors can be obtained by placing horizontal and vertical (following a matrix layout) electrode (alluminium) stripes directly on the sensor material. In this arrangement, the load has to be applied over the upper electrode. This is a drawback, since the sensor material is usually flexible, whereby the upper electrode is exposed to a bending stress, which reduce the life time of the sensor.
The second arrangement, adopted in the Fa-v2 and Fa-v3 versions,



(a) (b)

FIGURE 2.2: Two different arrangements for resistive tactile sensors: (a) double sided contacting of the sensor material, (b) single sided contacting of the sensor material.

solves this problem, since in this case the electrodes are not bended due to the material compression.

As before mentioned, the sensorised solution includes a layer of conductive polymer, obtained as a foam of Polyethylene and Carbon, which substitutes the original electrically insulating material of the commercial floating floor. The layer is an Electrostatic discharge (ESD) polyethylene extrusion with anti-static properties, medium density (32 Kg/m^3) and closed-cell foam. The obtained element is durable, lightweight, flexible, and solid. It meets the EIA 541 requirements [16] for static decay and surface resistance. Many types of conductive polymeric material with different resistivity, elasticity and temperature range are commercially available. The

mechanical and electrical characteristics of the polymer used in our application are specified in Table 2.1. Among others, the ESD roll shown in Figure 2.1(b) can be easily supplied at affordable costs. The anti-static characteristic is achieved through conductive chemical additives. These additives are usually incorporated into the foam during the manufacturing process. Such a technique guarantees an even distribution of the conductive elements throughout the block, which is mandatory to have a uniform electrical resistance property. As shown in Figure 2.3, regardless the used arrangement, when a pressure is applied on the top of the tiles, the rough surface of the conductive polymer is compressed onto the electrodes surface [62]. The electric resistance between the two stripes is related to their physical compression rate on the intermediate polymeric layer. In particular, the contacting area between polymer and electrodes is increased whilst the resistance between them is proportionally reduced. In other words, the sensor converts the applied pressure to an electric resistance, which is measured with the capture board described in Section 2.1.2.

2.1.2 The Capture Board

Each sensor module is equipped with a custom capture board (Figure 2.4(c)), which measures the electric resistance of each sensing element $S_{r,c}$ (cross between two electrode stripes). In addition, the

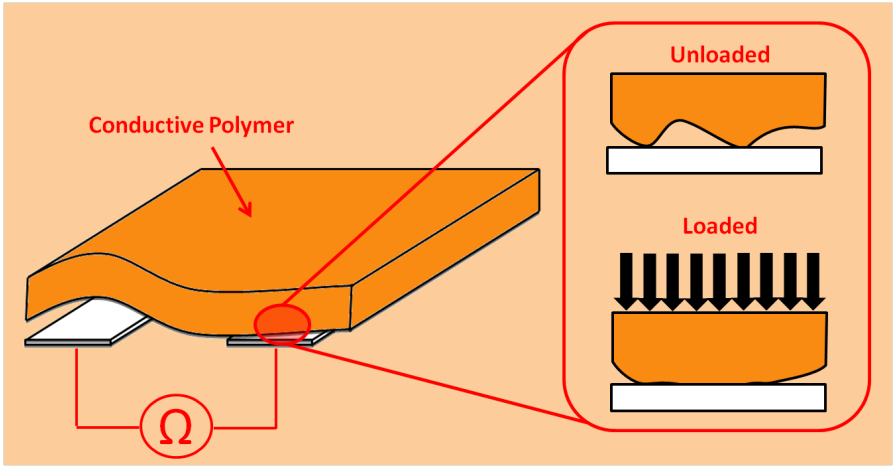


FIGURE 2.3: Working principle of a resistive tactile sensor cell.

board digitizes and transmits these values $\psi_{r,c}$ to the processing unit.

The board is equipped with a 32-bit ARM micro controller (STM 32 F 103 RB [2]). The signal conditioning is done using a custom circuit composed of an array of trans-resistance amplifiers. The amplifier outputs a voltage response V_o proportional to the conductance G of the corresponding sensing element. A base configuration of each cell is reported in Figure 2.4(a).

The sensing element is depicted as a resistance R_S. Due to the inverting configuration, a constant voltage of $V_{cc}/2$ is applied to the

[2] http://www.st.com

TABLE 2.1: Mechanical/electrical characteristics of the resistive polymer.

Characteristic	Min	Max	Unit
Thickness	2.5	3.0	mm
Compressive strength at 10 % of relative deformation	15	25	kPa
Compressive deformation at 100 kg for 60x60 cm tile	6	10	%
Temperature range	- 20	+80	^{o}C
Surface Resistance	1.0×10^5	1.0×10^{10}	Ω/sq
Volume Resistivity	1.0×10^4	1.0×10^{10}	$\Omega \times cm$

sensing element, inducing a current I_s through it:

$$I_s = \frac{V_{cc}/2}{R_S} \tag{2.1}$$

Since the electric currents I_f and I_s must have the same intensity, the output voltage Vo becomes:

$$Vo = V_{cc}/2 \cdot R_f \cdot G, \tag{2.2}$$

with Vo ranging from $V_{cc}/2$ (when $R_S >> R_f$) to V_{cc} (when $R_S \leq R_f$). Reference resistances R_f in the range of $[20 \dots 100]$ $K\Omega$ have been exploited in our experiments.

Thanks to the grid distribution of the sensing elements, the overall configuration of the amplifier array follows a matrix layout, as shown in Figure 2.4(b). An inverting amplifier has been connected to each column. Selecting one row at a time, the amplifier senses the electrical conductance of the cell placed in the cross

between the selected row and column.

The row selection is provided by an array of digital switches. If the row is connected to ground, the corresponding cell is exposed to a positive voltage and its behavior is the same of the schema in Figure 2.4(a). On the other hand, the other rows are connected to $V_{cc}/2$, and thus all the corresponding cells are subjected to a zero voltage. This solution has a good reliability against crosstalk effects [48], reduces the number of components and requires only a single supply.

The output pins of the amplifiers are connected to the Analog-to-Digital Converters (ADC) of the micro controller, which also triggers the analog switches of the row selection. The micro controller has been programmed with the acquisition process reported in Algorithm 1.

Each captured value is encoded using 16 bit unsigned integers. The device is able to sample each value in about 10μs, allowing a scan of the whole sensor matrix and the corresponding data encoding and transmission at up to 20 samples per second. Two operational modes have been implemented, *i.e.* the single scan and the streaming mode. In the first case, a single capture is done and sent for each request, whereas in the second one the sensor data are continuously captured and sent as a stream, until a stop command is

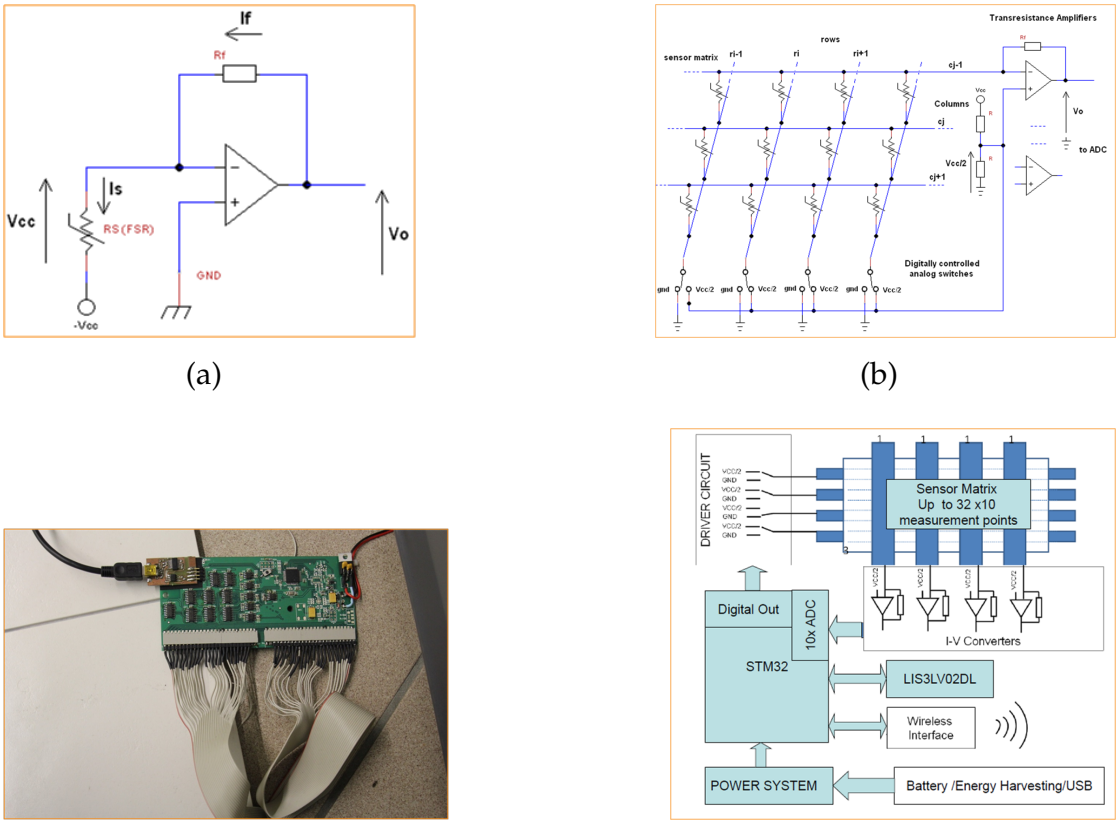
ALGORITHM 1: The acquisition process

while *stream enable* **do**
    $sensordata \leftarrow zeros(N, M)$;
    for $r := 1$ *to* N **do**
        Apply $0V$ voltage to row r;
        Apply $V_{CC}/2$ voltage to the other rows;
        for $c := 1$ *to* M **do**
            $V = Digitize(V_c)$;
            $sensordata[r, c] \leftarrow V$;
        end
    end
    Send $sensordata$
end

received. A schema of the hardware components of the signal conditioning circuit is reported in Figure 2.4(d).

2.1.3 The Ceramic Tiles

The sensitive area of the described stack does not cover the overall surface, but is limited to the regions of intersection between vertical and horizontal aluminum stripes. Therefore, only objects or people located within these regions can be detected. This drawback is solved thanks to the ceramic tiles, which are placed on the top of the sensors (see Figure 2.5). First, the tiles have been included in the system to obtain a complete and stylish floor solution, which should fulfill the requirements described in the previous chapter.



(a) (b)

(c) (d)

FIGURE 2.4: Signal conditioning circuit : (a) base and (b) matrix configuration. The capturing board: (c) a picture of the board and (d) a schema of the hardware components.

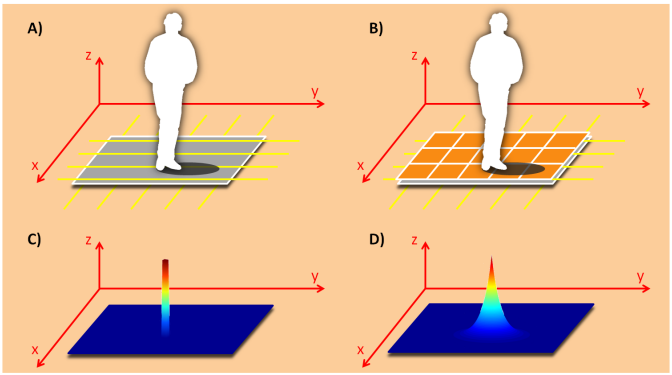


FIGURE 2.5: Measuring the pressure field. (a) the pressure field $f(x, y)$ generated by a person on the floor is captured by the sensor only when placed on the top of it. (b) Using the ceramic tiles, the pressure is diffused over the whole tile extent. (c) and (d) shown the weighting functions without and with the tiles respectively.

In addition to aesthetic and design reasons, the ceramic tiles play an important role in the sensor device, since they act as a blurring filter. Ideally, the pressure applied on a specific point is distributed on the entire floor, with an effect inversely proportional to the distance. As a consequence, the value $\psi(r, c)$ acquired by the sensor $S_{r,c}$ is related to all the pressure stimuli applied to the floor. The relation between the pressure field applied on the floor and the sensor response can be modeled with a kernel convolution:

$$\psi(r, c) = K_{r,c} \cdot \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x_0 + x, y_0 + y) \cdot w_1(x, y) \; dxdy \quad (2.3)$$

where (x_0, y_0) is the spatial location of the sensor $S_{r,c}$ on the floor, $f(\cdot,\cdot)$ is the pressure field orthogonal with the floor and $w_1(\cdot,\cdot)$ is a weighting function.

The kernel term $w_1(x, y)$ depends on the distance between the sensor and the object/person above the floor. The normalization constant K is related to several building factors and, thus, its value should be calibrated for each specific sensor $S_{r,c}$.

The response of the sensing layer (without the ceramic tiles) can be obtained using the weighting kernel $w_1(x, y)$ of Equation 2.4:

$$w_1(x,y) = \begin{cases} 1 & |x| < l \wedge |y| < l \\ 0 & otherwise \end{cases} \tag{2.4}$$

where l is the width of the aluminum stripes.

The sensing layer without the ceramic layer can be effectively used only with a very high spatial frequency of the stripes. This principle is similar to the one used in several previously developed carpets, where discrete arrays of FSR sensors are placed for detecting pressure. Since they are sensible only in the exact position where the pressure is held, many sensors must be used to avoid insensitive zones. As a result, the system becomes too expensive for covering large areas.

Pairing the sensing layer with very slim ceramic tiles, the response of the sensor with respect to the object distance follows

a Gaussian law. The corresponding kernel $w_2(x, y)$ is reported in Equation 2.5:

$$w_2(x, y) = \exp\left[-\left(\frac{(x-u)^2}{\sigma_x^2} + \frac{(y-v)^2}{\sigma_y^2}\right)\right] \tag{2.5}$$

2.2 Florim Age Module Versions

During the research activity, three different versions of the Florim Age technology have been designed and tested. Each version consists of rectangular modules (2x1 meters large) with a layered structure that can be arranged to cover areas with different sizes and shapes. Each module is composed of a *tile coverage*, a *conductive layer*, and a *capturing board* as previously described. Nevertheless, each version differs for the electrode stripes arrangement and technology. In particular, in the FA-v1 version the electrodes and the wires toward the capturing board are obtained with a sequence of aluminium stripes unrolled on the two sides of the conductive polymer, lengthwise on the top and crosswise on the bottom of the layer. The first arrangement described in Section 2.1.1 is exploited. On the contrary, Fa-v2 and FA-v3 versions adopt the second arrangement of the electrode stripes and exploit a rigid and a flexible Printed Circuit Board (PCB) to connect the conductive layer to the

capturing board, respectively. Advantages and drawbacks of each version are analysed in the following.

2.2.1 FA-v1

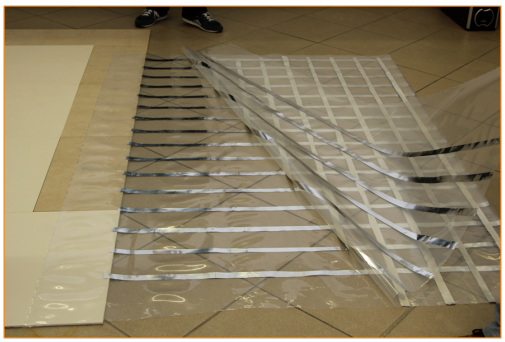
In FA-v1 version the conductive polymeric layer is put between aluminium stripe electrodes, lengthwise on the top and crosswise on the bottom, thus a *sensing element* $S_{r,c}$ is created at each intersection between a lengthwise and a crosswise stripe. The stripes cannot be glued to the conductive element, otherwise the contact resistance will not change during the working conditions.
Therefore, two thin polyethylene sheets have been included in the sensor stack as a support for the adhesive aluminum stripes (Figure 2.6(a)). The whole sensing module turns out to be composed of four layers: a polyethylene sheet with the horizontal conductive stripes on the bottom, the conductive polymer, another polyethylene sheet with the vertical conductive stripes from one side and the Velcro on the other side (Figure 2.6(b)), and ceramic tiles on the top (Figure 2.6(c)).
The spatial resolution of the sensor is defined by the distance between two consecutive stripes. A step of 125 mm has been adopted in the developed prototype, so that each rectangular module contains 8x16 sensing elements. This size is a reasonable tradeoff between a sufficient spatial resolution (to detect each foot of a walking person) and the sensor cost.

The connections of the stripes to the capture board have been provided by ribbon cables manually soldered to the stripe endings (see Figure 2.6(d)).

The FA-v1 version of the Florim Age technology has the advantage to be really not expensive and easy to reproduce for experimental environments, but, as mentioned in 2.1.1, in this configuration the upper electrode stripes are exposed to a bending stress, reducing the life time of the sensor module. Moreover, the external wires manually soldered to the stripes ending have proved to be not simple to reproduce in an industrial production process, making really difficult the coverage of wide areas.

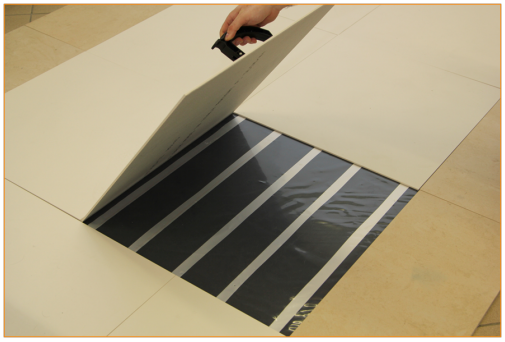
2.2.2 FA-v2

In the FA-v2 version, below the ceramic tiles coverage, the conductive polymeric coat is laid on a rectangular module obtained by joining together eight square sub-modules (50x50 centimeters large) as those proposed in Figure 2.7(a) and in Figure 2.7(b). Each sub-module is a rigid PCB in which the electrode stripes connections are directly etched from copper sheets laminated onto a fiberglass non-conductive substrate, obtaining the arrangement shown in Figure 2.7(c). The electrical connection between the upper copper layer and the lower one is obtained through the metallization

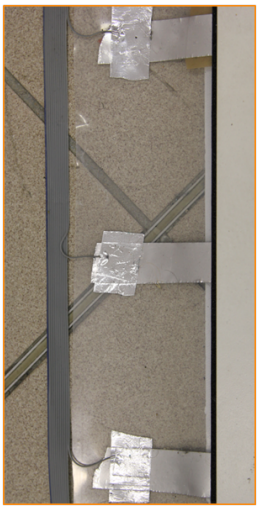


(a)

(b)

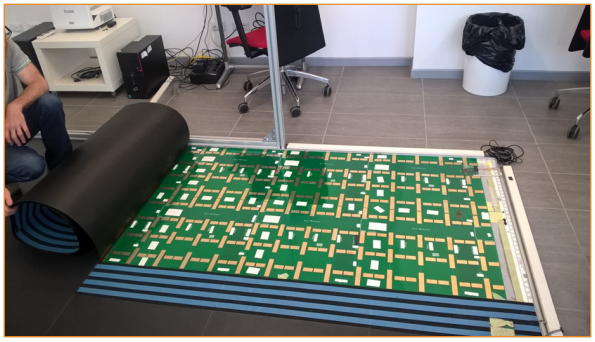


(c)

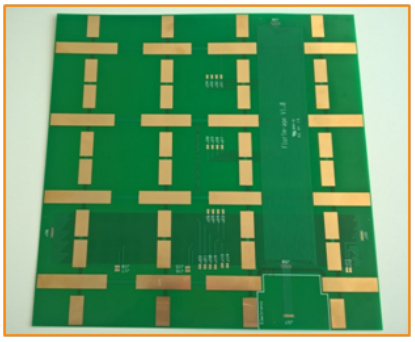


(d)

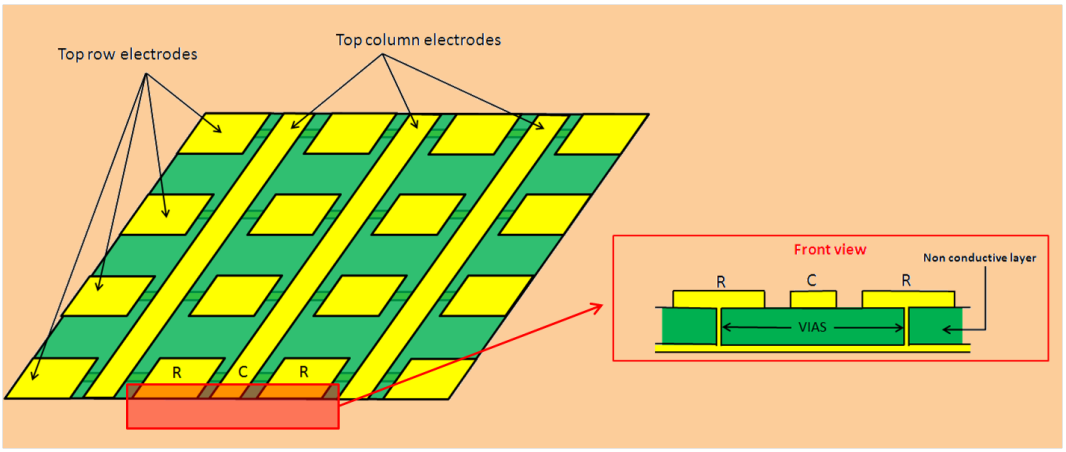
FIGURE 2.6: Hardware components of the FA-v1 sensing floor version: (a) the aluminum stripes glued on polyethylene sheets, (b) the conductive layer overlapping, (c) the ceramic floating tiles placed on the top of the sensors, (d) the external wires from the sensor to the capturing board.



(a)



(b)



(c)

FIGURE 2.7: Hardware components of the FA-v2 sensing floor version: a) the 2x1 meters large module below the ceramic tiles coverage and the conductive polymeric coat, b) the 50x50 centimeters sub module and c) a schema of its basic structure.

of all holes (called "vias"') previously punched via a Computer Numerical Control (CNC) milling machine. The overall module generating a 8 x 16 grid of sensors through the lengthwise and crosswise copper electrode stripes parallel connected with a single side of the conductive polymeric layer. The electronic capturing board is directly mounted on one of the eight sub-modules and connections between sub-modules are obtained via custom flexible connectors. This sensing floor version allows to create low cost setups and it is characterized by a high reliability and scalability features, that make the proposed solution very promising also from the commercial point of view, allowing wide installments in both private and public spaces.

2.2.3 FA-v3

The Fa-v3 Florim Age system can be seen as the natural technical evolution of the Fa-v2 version, but actually it represent just a proof of concept. The rigid FA-v2 module can be substitute with a flexible one in which the 8 x 16 grid of electrode stripes used to connect the conductive layer to the capturing board can be printed directly on a Polyethylene terephthalate (PET) sheet using the *ink jet material deposition* [55] and [63]. This technique is used to deposit materials of different physical characteristics on a substrates and aims to eliminate fixed costs of production, reducing the amount of materials

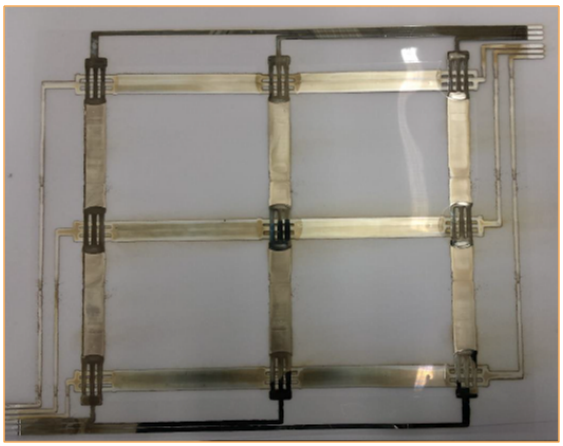
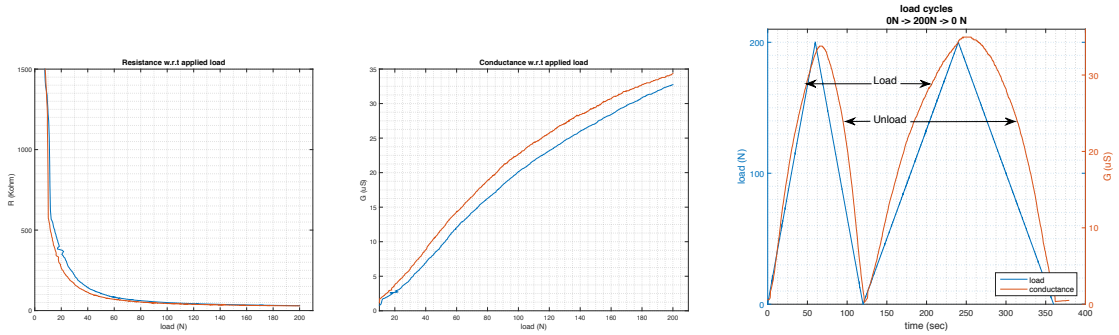


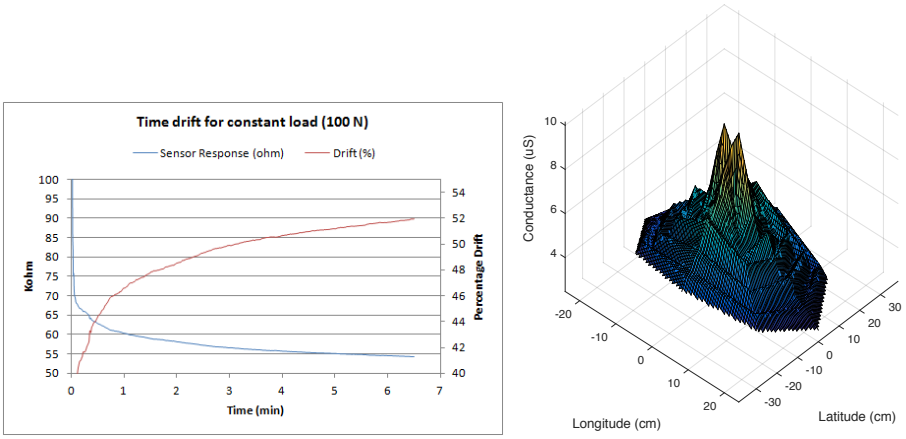
FIGURE 2.8: Prototype of a FA-v3 module.

used to realize a PCB. Moreover, using appropriate printing machines, this technique allows the production of larger modules respect to common rigid PCB production techniques. The alternating layers of conductive and non-conductive materials to be deposited on the PET sheet to realize the (2 x 1 meters large) FA-v3 module are shown in Figure 2.8. Preliminary laboratory experiments on prototypes have demonstrated the feasibility of the production process, but these prototype are still far from a reliable and definitive version.



(a) (b) (c)

FIGURE 2.9: (a) Resistance w.r.t. applied load, (b) conductance (G) w.r.t applied load, (c) load cycles 0N-200N-0N: applied load and conductance w.r.t time.



(a) (b)

FIGURE 2.10: (a) Sensor time drift for constant load (100N applied load).(b) Sensor element 200N side effect behavior.

2.3 Sensor Characterization

Aiming at characterizing the single sensor element behavior, a test bench has been designed to evaluate changes in the electrical resistance and the mechanical displacement due to the applied strain on a sensing module. The test bench consists of:

- INSTRON 5567 tension/compression testing machine;
- TEKTRONICS digital multimeter model DMM4040 6-1/2 digit precision;
- a Fa-v2 sub-module;

The device under test is a single sensor element, a three layers stack composed by a 15 x 15 (or 60 x 60) cm SLIM/4+ ceramic tile placed above a patch of the same size of the conductive polymer laid over a rigid PBC FA-v2 sub-module. The compression testing machine can exert to the sample a maximum force of 6750 lbs (30 kN). The machine can be equipped with different shape and size crosshead. We used a circular shape 6 cm diameter crosshead with a speed range from 3.33×10^{-5} in/s to 0.33 in/s (8.33×10^{5} mm/s to 8.33 mm/s). Force is measured with a load cell installed on the crosshead of the machine. An LVDT (linear variable displacement transformer) provides measurement of the cross-head position, and an extensometer is used to determine the strain applied to a sample. The machine is fully computer controlled, being able to

use any of the aforementioned transducers as the feedback signal. Data collection from the transducer signals is also computer based. Finally, to evaluate the sensor electrical response we used the digital multimeter.

Using the above mentioned experimental setup, we performed three different kind of measurements:

1. Sensor time drift for constant load: evaluation of the sensor time drift for constant load applied. The resistance drift of the single sensor element is an important parameter for FSR because their resistance vary over time when a constant load is applied and gives the force-sensing-resistors not suitable to perform quantitative measurements of force. The evaluation has been performed by applying a constant load of 100N (by placing a calibration mass) to a fixed tile position, capturing 1000 samples of the electrical resistance by DMM4040 multimeter for a test duration of 6 min. The results are shown in Figure 2.10(a). The trial has shown a drift percentage of 52% respect to the initial value over the whole sixty minutes test duration. The most consistent drift, more than 50%, occurs in the early stage of the test, during the first two minutes following the application of the constant load. Then, starting from the second minute up to the end of the test, the drift does not

exceed 2%. This can be explained by the viscoelastic properties related with the creep behavior that semiconductive polymer composites display (see [24] and [66]). When a constant force is applied, the resistance decrease with time because of the creep of the polymer.

2. Electrical resistance, hysteresis and mechanical displacement versus force characteristics: The evaluation of the electrical resistance, conductance, hysteresis and mechanical displacement versus force characteristics has been performed by INSTRON 5567 testing machine to apply a dynamic normal force to the sensor plane on a fixed tile position. Two ramps from zero to 200N and back to zero at the different slopes of 200 N/min and 100 N/min have been applied. The electrical resistance values have been measured on the digital multimeter by collecting 1000 samples (test duration of 380 sec). The results are reported in Figure 2.9. Figure 2.9(a) shows the measured resistance versus force characteristic, while Figure 2.9(b) shows the conductance versus applied force. The conductance has been calculated using the measured resistance values. The graphs report the resistance values collected during the application of the two load ramps labelled "Load" in the Figure 2.9(c). The force in the two ramps is from 0 N to 200 N, but it is applied at different slope, 200 N/min and 100 N/min respectively. The trial shows the sensor response is

related at the speed of the applied load slope, the gap is less than 0.9 N/min.

3. Ceramic tiles blurring filter effect: The blurring filter effect due to the role played by ceramic tiles. The ceramic tile is located at the top of the three layers stack structure of the sensor element. The pressure due to the force exerted by an object on the tile is spread on a wide area, as consequence, each sensor captures the weights applied on a neighborhood of its position. This effect has been evaluated using a 60x60 cm wide tile with only one golden copper stripe electrodes located (buried) in the middle. Using a 6 cm diameter crosshead, a 10kg reference mass has been placed on the tile at different positions following a 2x2 cm mesh grid. The Figure 2.10(b) shows the blurring filter effect plotting the sensor conductance as a function of the sensor-to-reference mass distance.

Chapter 3

Florim Age: Network Architecture and Communication Protocols

3.1 Florim Age Network Architecture

The hierarchical system architecture we propose is consisting of three different levels of deployment: sensing floor modules, local processing units, and a desktop class host/server machine. A high level view of this three-tiered architecture can be seen in the Figure 3.1.

At the lowest tier more sensing floor modules can be equipped in parallel and can be synchronized to cover large areas, adopting the same methodology used for creating large camera networks.

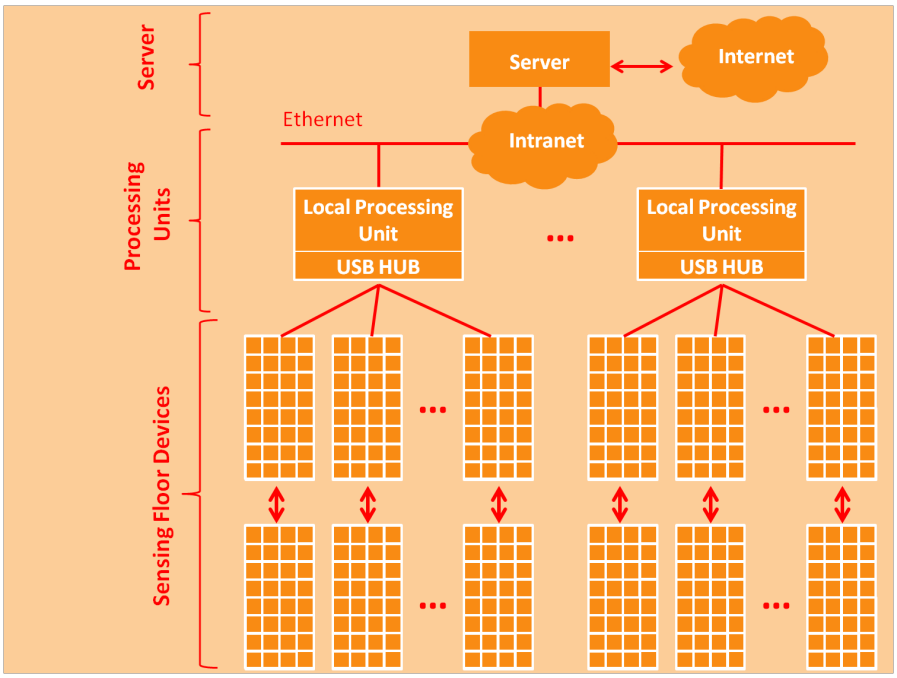


FIGURE 3.1: Architecture of the Florim Age System.

Each device has a unique identifier that the low-level processing uses to address and manage multiple sensors. It is connected to a local processing unit through a USB link, exploiting compliant hubs if needed.

The third tier is composed by a dedicated server connected to the local processing units through an Ethernet network. The server also hosts a database containing information about the different sensor modules and the local processing units available. Finally, the server makes available different data interfaces for the end-users.

The entire system architecture is based on cheap, off-the-shelf hardware and exploits open-source software at every tier. In addition, other sensor devices located in the same environment can be included and the corresponding data can be sent to the server by means of the local processing units.

3.2 Local Communication Networks

3.2.1 Serial over USB Network

The capture board described in Section 2.1.2 is equipped with two serial ports for data communication, namely a main port and an auxiliary port. RS422 or RS232 serial standards are supported by both of them. The role of the main port is to provide a direct communication channel to the low-level processing unit, whereas the auxiliary port can be used to establish a chain of neighbor capture boards. In this case, each board receives data packets from the auxiliary port and forwards them to the next board of the chain or to the processing unit through the main port. The data packet contains a header field, which embeds the unique identifier of the board, a body with the sensor data, and a check-sum byte at the end. Sensor values are arranged by rows and added to a data packet whose format is depicted in Figure 3.2. $S_H(i,j)$ and $S_L(i,j)$ are the most and the least significant byte from the sensor $S_{i,j}$, respectively.

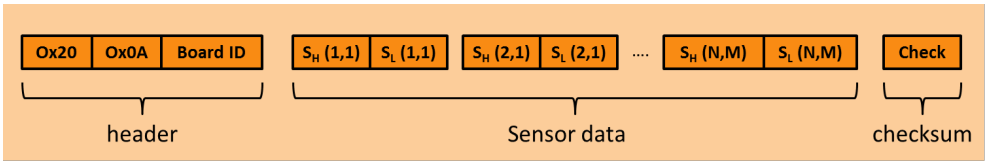


FIGURE 3.2: Data packet format.

The main port is also equipped with a serial to USB adapter to provide an easier connection with the low level processing unit. A serial to Bluetooth adapter is also available if a wireless connection is required. The wired connection is preferable since it is more stable and, together with data cables, power lines can be installed toward the board as well. At the same time, a wireless connection drastically reduces the network and installation costs, but a battery power supply should be used.

The adopted serial protocol does not exploit compression algorithms to reduce the computational load on the capture board. The following analysis of *bandwidth requirements* confirms the scalability of the system even without data compression. As reported in Section 2.1.1, a sensor module contains 128 sensors and covers a region of two square meters. The corresponding data packet has a fixed size of 260 bytes. We have estimated the required bandwidth in three different hypothetical setups, *i.e.* a single module, a sensing floor for a typical room of 16 square meters, and a very large area of 10.000 square meters. Three different frame rates have been also

TABLE 3.1: Bandwidth estimation in three different hypothetical setups

	Single module	*Typical room*	*Large area*
Area $[m^2]$	2	16	10.000
N. modules	1	8	5.000
Sensors	128	1.024	640.000
Bytes for frame	260	2.080	1.300.000
Bandwidth @ 5fps [B/s]	1.300	10.400	$6.5 \cdot 10^6$
Bandwidth @ 10fps [B/s]	2.600	20.800	$13 \cdot 10^6$
Bandwidth @ 20fps [B/s]	5.200	41.600	$26 \cdot 10^6$

considered. Table 3.1 summarizes the results. Even in the worst case, the bandwidth is lower than 30 MB/s, which is more or less the real limit of a single USB 2.0 link. Therefore, the implementation of compression algorithms is not required. In addition, the total number of sensors in the very large room is comparable with the number of pixels of a video camera. The further processing algorithms should fulfill similar constraints imposed on real time surveillance systems. As a consequence, it is reasonable to assume that a single processing unit supports a whole sensing floor in most of the real setups.

The adoption of the RS232 to USB converter leads to two main benefits. From one side, it simplify the connection to the low level processing unit, reducing any problem of compatibility and port

availability. Moreover, USB hubs can be added to the system multiplexing some links and eliminating the requirement of a one-to-one connection between a sensing floor module and the corresponding local processing unit. This enables significant cost savings without compromising any of the system functionality.

Even more, the USB hubs give to the system one of its most important capability: the binary power-control over the sensing modules in the system. The USB Hub Specification 2.0 requires that self-powered hubs support port power switching. By sending a suitable USB control message, the software can control the power state of a given port on the hub, effectively enabling/disabling the power supply for any attached device(s).This means that we are able to individually control the power supply of any sensor node in the system architecture by simply issuing a suitable USB control message to that particular hub to which the corresponding sensing module is connected. Finally, additional sensors (e.g., Microsoft Kinects, cameras, or mems) can be connected to the low-level processing units through this USB hierarchy.

3.2.2 Serial Communication Protocol

The low level processing unit communicates with each board using an ad-hoc communication protocol. In particular, commands can be sent using a single 8-bit data. The four types of commands are reported in Table 3.2. The capture board replies with a information

or data packet (or a stream of them, if required). Details on the format of the info and data packets are reported in Tables 3.3 and 3.4, respectively.

TABLE 3.2: Serial commands

Command	**Description**	**Expected answer**
's'	data stream start	stream of data packets
'e'	data stream end	nothing
'1','2','3','4'	set capture frame rate (5, 10, 15, 20 Hz)	nothing
'i'	get information	info data packet

TABLE 3.3: Info Packet format

Field	**Size [Bytes]**	**Description**
Board Id	2	Id of the board (LSB/MSB)
Firmware version	2	Version of the capture board firmware
Framerate	1	Capture speed [Hertz]
NR	1	Number of rows (default 16)
NC	1	Number of columns (default 8)
BPP	1	bit per pixel (default 16)

3.2.3 The Ethernet Network

The connection among the low-level processing units and the server is provided by a LAN network, which guarantees a better scalability, remotization and distribution of the system.

The USB infrastructure is limited to 127 USB devices with a maximum distance of 30 m between the low level processing unit

TABLE 3.4: Data Packet format

Field	Size [Bytes]	Description
header	2	0xFD 0xFD
N	2	frame number (auto-increment)
Data	NR x NC x 2	sequence of sensor values (by rows)
Check-sum	1	bitwise Xor (header included, check-sum excluded).
Footer	1	0xFE

TABLE 3.5: Default Serial link parameters

Parameter	Value
Baud rate	460800
Databits	8
Bit Stop	1
Flow control	None
Parity	None

and the sensor modules (achieved by daisy-chaining of up to 5 USB hubs) [22]. While suitable for small to medium size architectures, these limitations are not compatible with our goals for scalability of the architecture and support for geographically extensive deployments.

Thus, the local processing units also act as bridge between the USB infrastructure and the Ethernet one. In this case, the serial data packets described above are embedded in TCP or UDP data packets, if needed.

Finally, the Ethernet Network supports the transfer of processed

data as described in the following chapters. Several application-specific and generic protocols have been defined and tested. Some examples are reported in Chapter 5.

Chapter 4

Data Processing

This chapter describes the software algorithms and architectures specifically conceived to process floor data. As reported in Chapter 3, sensor values are sent to a processing unit which aim at extracting high level information, such as presence and positions of people and objects. Two main approaches can be adopted. The first one takes inspiration from the video surveillance field. Sensors are seen as pixels of a video camera. Thus, each frame (called floorimage, hereinafter) contains a representation of the sensor data, where the intensity of each point is proportional to the sensor value. A grid distribution of the sensors is mandatory. Then, similarly to video surveillance systems on static cameras, moving objects are detected with the help of background subtraction steps and tracked using discriminative or generative approaches. Finally, action and behavior classification algorithms are adopted as last step to extract high level information for each tracked person (see Fig. 4.1).

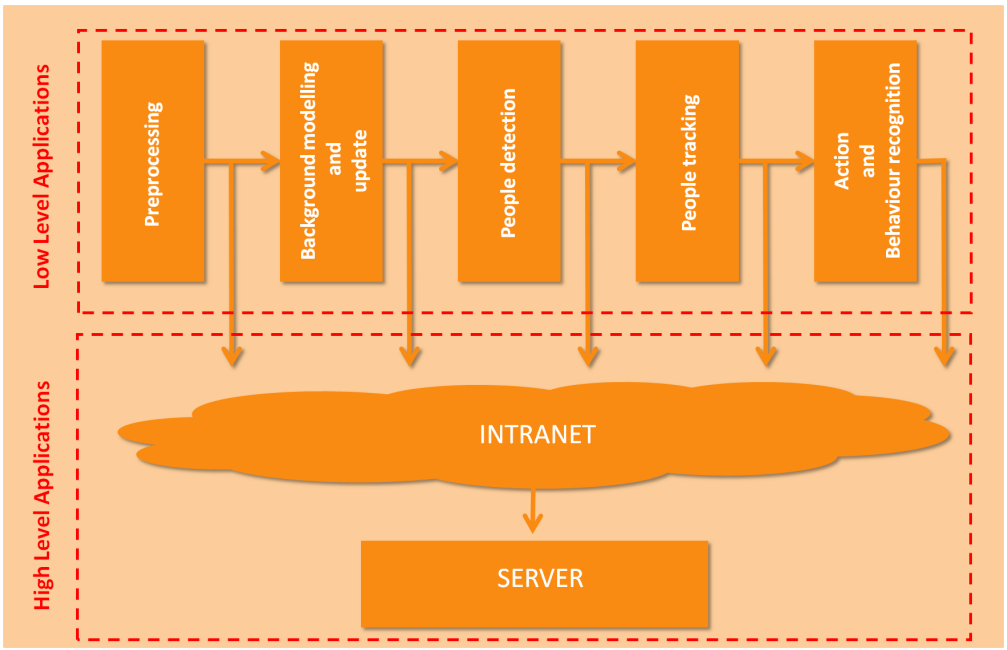


FIGURE 4.1: Schema of the low level processing engine using the floorimage model.

In addition to the floor image model, we studied and tested an alternative method based on Center Of Pressures (COP), which does not require a grid distribution of the sensors.

4.1 Floorimage data processing

4.1.1 Definitions and Floorimage acquisition

Let Ψ be the whole data captured from a generic sensing floor. As a mandatory constraint, the sensing elements are spatially arranged on a grid; then the captured data Ψ can be stored as a 3D matrix $\Psi = \{\psi(x, y, t)\}, x \in [1 \dots W], y \in [1 \dots H], t \in [1 \dots T]$. W and H define the size of the whole floor grid and T is the number of temporal samples.

Each element $\psi(x, y, t)$ is a digital version of the value sampled at time t from the sensor $S(x, y)$. The spatial neighborhood relations among sensors are preserved in the matrix Ψ.

Due to anomalies of the coat below the floor or irregular placement of the tiles above it, each matrix element $\psi(x, y, t)$ may have a proper output range different from the others. A linear transformation should be included in the capturing step to normalize the sensor response $\hat{\psi}(x, y, t)$, as in the following equation:

$$\psi(x, y, t) = \alpha_{x,y} \cdot \hat{\psi}(x, y, t) + \beta_{x,y}. \tag{4.1}$$

In real implementations, the whole data Ψ are collected by multiple sensor modules working in parallel for scalability issues. The modules can be freely disposed on the floor to cover large areas. However, since a grid distribution of the modules leads to an easier

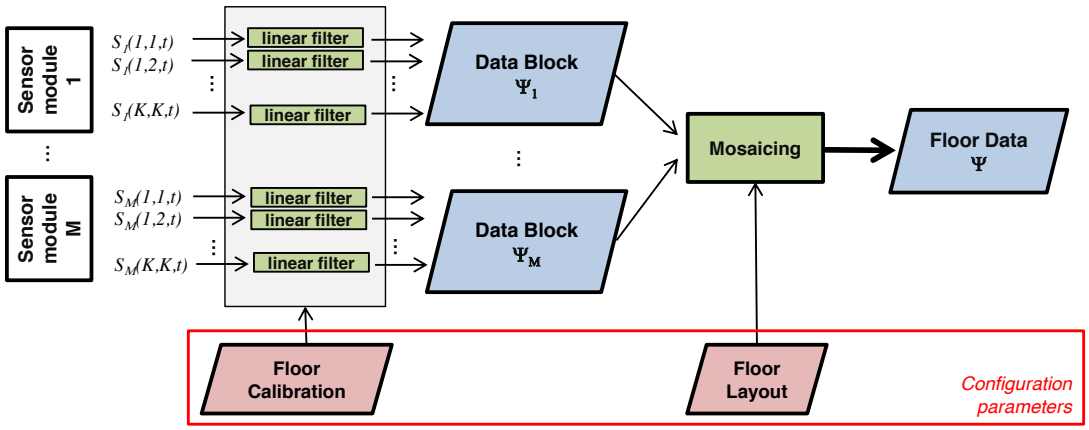


FIGURE 4.2: From sensors to floorimages: the output value normalization and the module mosaicing step used to generate the whole floor data Ψ.

integration and processing of the data, we adopt this grid assumption. Therefore, Ψ is obtained by mosaicing the sub-matrices Ψ_m, $m \in [1 \dots M]$ sampled from the M modules.

A schema of the acquisition phase is reported in Figure 4.2. Each sensor $S_m(x, y)$ delivers a digital value $\hat{\psi_m}(x, y, t)$ at every instant t. The index m refers to the hardware module which contains the sensor, while the coordinates x, y indicate the position of the sensor within the module using the local reference system (see Figure 4.3(c)).

Chapter 6 describes an automatic system to estimate the *Sensor Calibration* and the *Floor Layout* configuration parameters (red

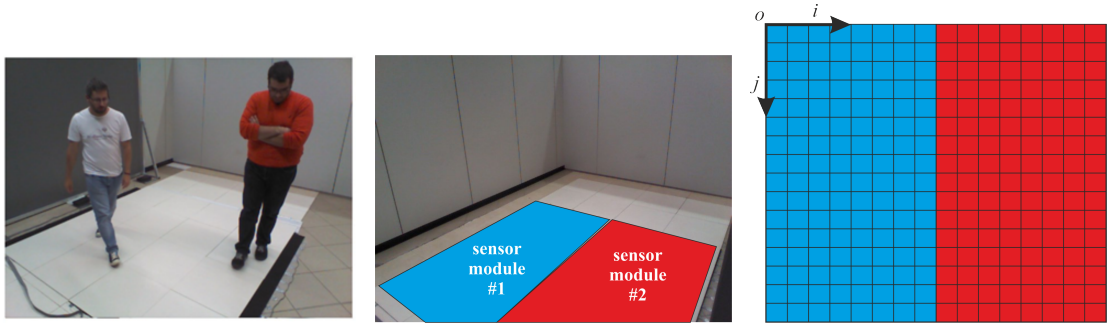
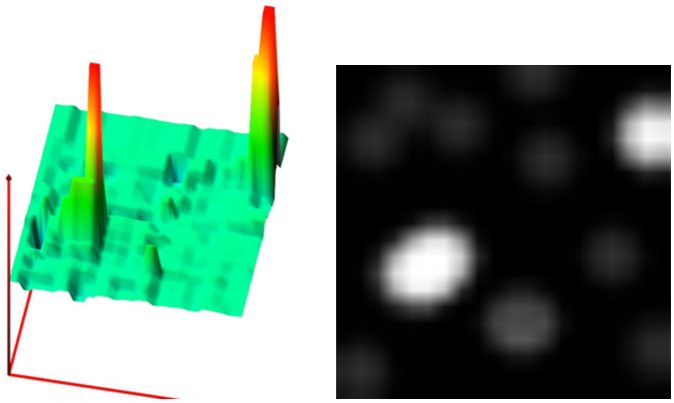


FIGURE 4.3: A sensing floor prototype (a) composed by two modules (b) of 8x16 sensors each. The complete Florimage is a matrix of 16x16 pixels (c).

parallelograms in Figure 4.1) directly from training data. The pixel value $\psi(i, j, t)$ is proportional to the corresponding value digitized by the capture board ($sensordata[r, c]$) and scaled to fit the range [0-255] of 8-bit gray-level images (see Fig. 4.4).

4.1.2 Background modeling and update

Floor image pixel values include contributions due to deadweight, tile weight and noise, in addition to the pressure applied by an object or a person. Moreover, the presence of background objects in the scene (e.g., a chair or a bag left on the floor) makes the sensor calibration unfeasible. Thus, floor images are firstly pre-processed in order to filter out those undesired contributions. The operation



(a) (b)

FIGURE 4.4: Graph of the sensor values (a) and the foreground image (b) obtained with the setup shown in Figure 4.3.

is akin to the background subtraction task included in computer vision systems for people detection with fixed-cameras [33, 23].

The foreground image $F = \{f(i,j,t)\}$ is computed as follows:

$$f(i,j,t) = \begin{cases} \psi(i,j,t) - b(i,j,t) & \text{IF } (\psi(i,j,t) - b(i,j,t)) > Th \\ 0 & \text{OTHERWISE} \end{cases} \tag{4.2}$$

where $b(i,j,t)$ are the pixels of the background model B available at time t (floor-background) and Th is a fixed threshold.

In most of the real tests, we have adopted a statistical background model with a very low updating rate. Each background pixel $b(i,j,t)$ is estimated as the mean value of N_b samples. A

uniform sampling within a sliding window is performed to speed-up the background estimation step and to reduce storage requirements, as defined in Equation 4.3.

$$b(i,j,t) = \text{mean}\left\{\psi(i,j,t-\Delta t),\ \ldots,\psi(i,j,t-N_b\Delta t)\right\} \tag{4.3}$$

where N_b is the number of samples used and Δt is the temporal interval between two consecutive samples. In the following experiments we have set $N_b = 10$ and $\Delta t = 30$ s.

4.1.3 People detection and tracking

The spatial resolution of sensing elements is set to 125 mm, as described in Chapter 2. It is reasonable to presume that at least one sensor has a non-zero response when a person is walking on the floor. On the contrary, each person usually stimulates more than one sensor with each foot. The final aim is the estimation of the correct number of people on the floor as well as their position at each time. We follow the well-established tracking-by-detection paradigm, largely adopted in people surveillance through computer vision [8, 7].

In computer vision, people tracking is very challenging because there are many sources of uncertainty for the object locations, such

as measurement noise, clutter, changing background, and significant occlusions. In order to cope with these difficulties, tracking-by-detection approaches have become increasingly popular, driven by the increasing reliability of object detection algorithms.

Such methods involve the continuous application of a detection algorithm in individual frames and the association of detections across frames. In video surveillance, this technique is in contrast to background modeling-based trackers. The main challenge when using an object detector for tracking is that the detector output is unreliable and sparse, i.e., detectors only deliver a discrete set of responses and usually yield false positives and missing detections. Thus, the resulting association problem between detections and targets is difficult.

People detection on floor data has some advantages, in particular thanks to the (reasonable) absence of occlusions, a very slow modification of the background due to environment factors (such as temperature and humidity, which can be compared to the illumination changes in computer vision), a favorable signal-to-noise ratio. At the same time, floor data has two additional drawbacks. The first one is related to the unavoidable bi-modal interaction of each person with the floor. The contribution of each person's foot, in fact, has a temporal intermittent behavior which is not strictly related to other foot (see Fig. 4.5). Specially in crowded situations, it could be very hard to estimate the person position and number

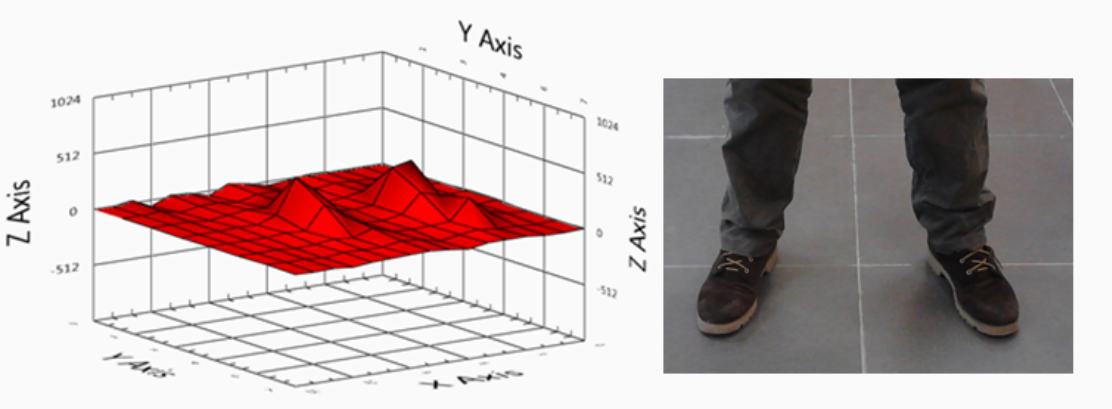


FIGURE 4.5: Picture of a person standing on the sensing floor and the corresponding captured values. The two peaks correspond to the two feet, contemporaneously touching the floor during static poses.

from the positions of the feet touching the floor only. In addition, floor data is subject to the "Confusion Problem" [11]: confusion happens when there are similarly-appearing objects close to the object of interest. The similarity causes trackers to confuse the object of interest with the others.

We solved both the problems with constraints on the spatial distance among detections. The non-zero values of the foreground image are clustered using a Gaussian mean shift [12], starting from all the local maxima as seeds. The variance parameter σ^2 of the kernel has been set to $50^2\ cm^2$, which is large enough to cover a typical human step and to consider all possible data contributions due to both the feet of a walking person (see Figure 4.6). The selected σ value allows to filter out noisy peaks within the Kernel radius. If two

or more people are too close each other (the inter-person distance $\overline{P_A P_B}$ of Figure 4.6 is comparable with the σ value), the mean-shift algorithm will generate a single detection for the whole group.

The association of detections to people (i.e., the tracking algorithm) is based on positions only. The tracker should handle two main challenges: the motion of the person and the absence of interactions with the floor during aerial phase of a running or a walking. Given the detections D_i at frame t and the current set of tracks T_j, we first compute the Euclidean distance matrix $M(i,j) = D_2(D_i, T_j)$. The detection to track association is provided using the schema proposed in [39]. For each frame, some detections may be assigned to tracks, while other detections and tracks may remain unassigned. The assigned tracks are updated using the corresponding detections. The unassigned tracks are marked invisible. Finally, unassigned detections begin new tracks. Each track keeps count of the number of consecutive frames where it remained unassigned. If the count exceeds a specified threshold, the tracking algorithm assumes that the object left the floor and it deletes the track.

Since our applications are not required to work in very crowded situations, the implemented tracking algorithm handles neither groups (i.e., people closer to each other than a foot step) nor abrupt position changes (e.g., people leaping around). The reader can refer to [49] for more complex tracking schemes, if required by the

application.

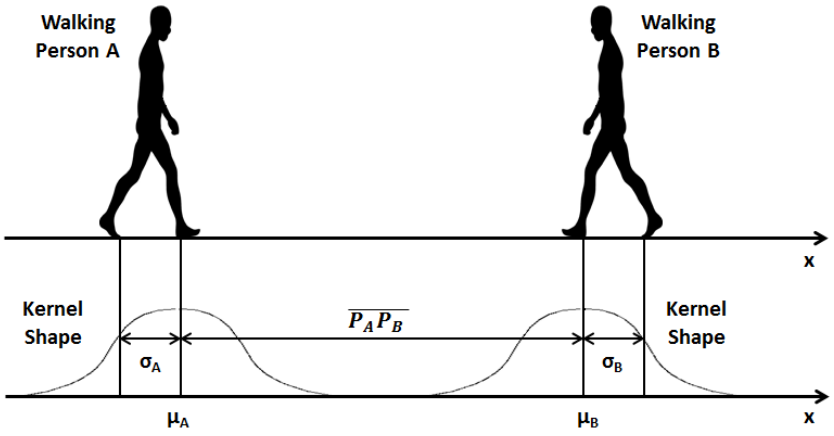


FIGURE 4.6: The Gaussian Kernel used in the mean shift clustering of people positions. The inter-person distance $\overline{P_A P_B}$ between two users should be greater to the σ kernel parameter in order to be correctly distinguished.

4.1.4 Action and event recognition

The recognition of people actions and behaviors using video cameras has been deeply addressed in the past [35]. A very common solution is based on the classification of spatio-temporal descriptors (STD) [25, 20]. We applied the same approach to floor images. In video analysis, spatio-temporal patches can be obtained by stacking a set of rectangular sub-images of interest, which have

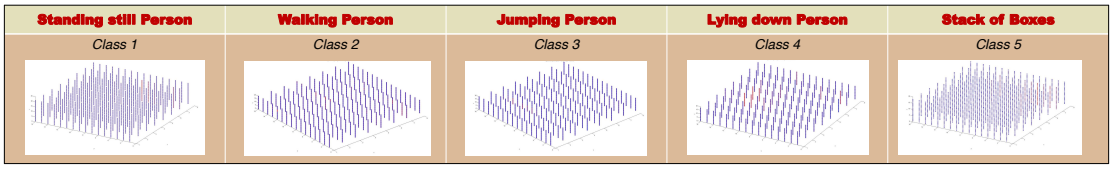


FIGURE 4.7: Examples of spatio-temporal patches extracted during different action classes.

been cropped from each frame (see Fig. 4.7). Similarly, the selection of the rectangular spatial regions on floor images may follow two different approaches, called *region-based* and *person-based* in the following lines. In the region-based approach, floor images are always cropped on the same fixed region, such as the square area covered by a tile. In this case, the region is manually selected independently from its content. As a consequence, the patch may contain a person, an object or even nothing. If a person or an object is located within the region, its position with respect to the patch center is not fixed. This first method does not require people detection and tracking.

In the person-based approach, instead, the region of interest follows and is centered around the estimated position of a detected and tracked person. Differently from the previous case, the person is always in the center of the patch. This assumption leads to the selection of different feature sets and to the recognition of different types of action. We have addressed the region-based action recognition approach in the previous work [27], while in this section we describe the person-based method.

Generic people actions depend both on the spatial movements of the person over the floor and on the type of interaction with the floor itself. For example, a person walk can be recognized using the trajectory on the floor, while a jump also requires the temporal analysis of the data. For this reason, two corresponding categories of features have been extracted, called *tracking-based* and *patch-based*, respectively. Tracking-based features Φ' are defined as follows:

$$\begin{aligned} \Phi' &= \{\phi'_1, \phi'_2\} \\ \phi'_1 &= \textstyle\sum_{k=t-\Delta t}^{t+\Delta t} |x_{k+1} - x_k| + |y_{k+1} - y_k| \\ \phi'_2 &= |x_{t+\Delta t} - x_{t-\Delta t}| + |y_{t+\Delta t} - y_{t-\Delta t}| \end{aligned} \tag{4.4}$$

where (x_t, y_t) is the position of the person on the floor at time t and $2\Delta t$ is the patch size along the temporal axis.

The two features ϕ'_1 and ϕ'_2 take into account the motion of the person on the floor. ϕ'_1 is proportional to the length of the person trajectory, while ϕ'_2 is related to the distance between the source and the target position of the person within the considered patch. For example, the combination of both the features allows to distinguish among a static person, a person walking straight and one swinging on the same position.

Patch-based features Φ'' are computed on the three dimensional matrix $\mathbf{P}(i, j, t)$, which contains all the foreground values included in the selected spatio-temporal patch.

Let $M(t)$ be the mean value of $\mathbf{P}$ at time t and $b(t)$ the coordinates of the barycenter of the slice of $\mathbf{P}$ extracted at time t. Since the patch is extracted around the person's position estimated by the tracker, the barycenter $b(t)$ will be close to the center of the patch. However, $b(t)$ is also influenced by the person pose and the corresponding non-uniform distribution of his weight. $C(t)$ is the covariance matrix of the vectors $\{x, y, P(x, y, t)\}$. Let $\{e_1(t), e_2(t), e_3(t)\}$ be the three eigenvalues of the matrix $C(t)$.

The features $\Phi'' = \{\phi''_1, \ldots, \phi''_{16}\}$ are extracted as average, min or max of the previous defined values as follows:

$$
\begin{aligned}
&\phi''_1 = \tfrac{1}{n} \cdot \textstyle\sum_{t=1}^{n} M(t) \\
&\phi''_2 = \tfrac{1}{n} \cdot \textstyle\sum_{t=1}^{n} \left(M(t) - \phi_1\right)^2 \\
&\phi''_3 = \tfrac{1}{n} \cdot \textstyle\sum_{t=2}^{n} |M(t) - M(t-1)| \\
&\phi''_4 = \tfrac{1}{n} \cdot \textstyle\sum_{t=2}^{n} \left(|M(t) - M(t-1)| - \phi_3\right)^2 \\
&\phi''_5 = \tfrac{1}{n} \cdot \textstyle\sum_{t=2}^{n} ||b(t) - b(t-1)|| \\
&\phi''_6 = \tfrac{1}{n} \cdot \textstyle\sum_{t=1}^{n} (||b(t) - b(t-1)|| - \phi_5)^2 \\
&\{\phi''_7 \ldots \phi_9\} = \tfrac{1}{n} \cdot \textstyle\sum_{t=1}^{n} \{e_1(t) \ldots e_3(t)\} \\
&\{\phi''_{10} \ldots \phi''_{12}\} = \min_t \{e_1(t) \ldots e_3(t)\} \\
&\{\phi''_{13} \ldots \phi''_{15}\} = \max_t \{e_1(t) \ldots e_3(t)\} \\
&\phi''_{16} = \tfrac{1}{n \cdot M \cdot N} \cdot \textstyle\sum_{t=2}^{n} \sum_{x=1}^{M} \sum_{y=1}^{N} |P(x, y, t) - P(x, y, t-1)|
\end{aligned}
\tag{4.5}
$$

where ϕ''_1 and ϕ''_2 are the mean and variance of $M(t)$ over the temporal interval; ϕ''_3 and ϕ''_4 are the mean and variance of the $M(t)$ variations. ϕ''_5 and ϕ''_6 take into account the movement of the barycenter. Finally, ϕ''_7 to ϕ''_{15} evaluate the average, the minimum and the maximum three eigenvalues of the covariance matrix. Finally, ϕ''_{16} is the Mean of Absolute Differences (MAD) between consecutive values sensed by the same sensor. The concatenation $\Phi = \{\Phi', \Phi''\}$ of the tracking-based and patch-based features is used as input to a set of supervised Random Forest classifiers.

Random forests [6] are an ensemble learning method for classification (and regression) that operate by constructing a multitude (forest) of decision trees at training time. Each tree is learned on a random subset of the training data, generating a weak classifier. The output class of the forest is the mode of the classes output generated by the individual trees.

Since the proposed framework is conceived for human-computer interactions, a small set of natural and basic actions has been selected in this work. In particular, we aim at simulating common input devices such as a multi-touch screen or a mouse. People locations on the floor define the positions of the input commands, while jumps are mapped to *click* events. The *lying on the floor* class has been included in the action recognition classifier to enable surveillance applications. Finally, an *empty* class has been added to handle errors of the detection and tracking system. A

TABLE 4.1: Description of the considered classes.

Class	Description
class 0	Absence of people or weights on the floor
class 1	Presence of a standing person on the floor
class 2	Presence of a walking man on the floor
class 3	Presence of a jumping man on the floor (single jump)
class 4	Presence of a lying person on the floor

qualitative and a quantitative analysis of the selected features and the implemented classifiers is provided in an experiment proposed in [27]. Using an experimental environment consists of a FA-v2 Florim Age system of 12 square meters, data from the floor have been acquired at a scan rate of 10 Hz. Three classifiers have been trained using the proposed feature set proposed, extracted from 100 acquisitions of 2 seconds for each class listed in Table 4.1.

In addition to the results obtained with Random Tree classifiers, the system performances using linear Support Vector Machines (SVM) and Nearest Neighbor classifiers have been evaluated. The classification accuracy is computed adopting the leave-one-out methodology and applying the one-against-all configuration for the binary classifiers. Table 4.2 shows the obtained results, highlighting a very promising accuracy (average of 97%).

In order to evaluate the efficacy of the selected features to discriminate a standing person (class 1) from a fixed obstacle of similar

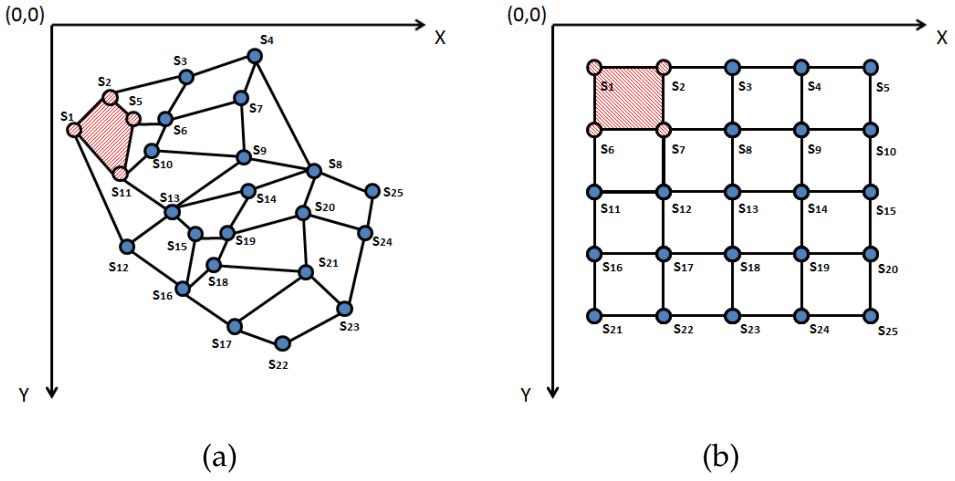
TABLE 4.2: Accuracy results using Support Vector Machine

	SVM	**Random Trees**	**N.N.**
class 0 vs all	98.8%	**99.8%**	
class 1 vs all	95.8%	**92.6%**	
class 2 vs all	94.4%	**98.0%**	85.8%
class 3 vs all	92.0%	**96.2%**	
class 4 vs all	97.6%	**98.4%**	

TABLE 4.3: Accuracy results of class 1 vs class 5

	Accuracy
SVM (class 1 vs class 5)	89.5%
RF (class 1 vs class 5)	98.8%
NN (Euclidean)	85.8%
NN (Manhattan)	85.6%

size and weight, an additional class (class 5, hereinafter) has been introduced. To this aim, the system has been tested on further 100 acquisitions from the selected region with the presence of a static object having a comparable weight. The accuracy of the three classifiers to discriminate between standing people and fixed objects are reported in Table 4.3.



(a) (b)

FIGURE 4.8: Schema of two different placements of 25 sensors with (a) a random distribution or (b) a grid. Examples of Floor Cells with 4 sides and 4 vertices are colored in red.

4.2 The COP MODEL

4.2.1 Data Model

At each capturing interval t, all the sensors $s \in S$ provide a corresponding discrete response $V_s(t)$. The whole area covered by the set of sensors can be partitioned into convex polygonal cells, having the sensors as vertices. The type of polygonal decomposition strictly depends on the sensor layout and the spatial resolution. Two very different cases are reported in Figure 4.8. If the sensors

are placed with a regular grid distribution as usual, a possible decomposition is composed by rectangular cells as reported in Figure 4.8(b). The obtained areas are called **Floor Cells** hereinafter, and are represented by the set of sensors placed at the vertices: $FC_i \subset S, (i = 1, 2, \cdots, M)$.

The state of a Floor Cell FC_i is represented by a $3D$ point P_i $(i = 1, 2, \cdots, M)$ as follows:

$$P_i(t) = \begin{bmatrix} P_i^x(t) \\ P_i^y(t) \\ P_i^z(t) \end{bmatrix} = \left[\frac{\sum_{\forall s \in FC_i} X_s \cdot V_s(t)}{\sum_{\forall s \in FC_i} V_s(t)}, \frac{\sum_{\forall s \in FC_i} Y_s \cdot V_s(t)}{\sum_{\forall s \in FC_i} V_s(t)}, \sum_{\forall s \in FC_i} V_s(t) \right]^T . \tag{4.6}$$

The first two coordinates P_i^x, P_i^y are respectively equivalent to the plane coordinates of the COP, while the third one P_i^z is the intensity of the GRF associated to the floor cell.

When no pressures is exerted on a floor cell FC_i, the location P_i^{eq} of the 3D point is only influenced by the sensor calibration and the dead weight of the tiles. Instead, when a person walks on the floor, the corresponding pressure moves the point P_i from the equilibrium state P_i^{eq} toward a new position $P_i(t)$. From Eq. 4.6, the projection of P_i on the ground plane falls within the FC_i convex-hull.

At time t, a floor cell FC_i can be considered in an excited state (i.e., a person or an object is located on the corresponding region) if

the following condition occurs:

$$||N_i \cdot (P_i(t) - P_i^{eq})||_2 \geq TH \tag{4.7}$$

where TH is an application defined threshold, $||\cdot||_2$ is the Euclidean norm, N_i indicates the following normalization matrix:

$$N_i = diag\left(d_{x,i}^{-1}, d_{y,i}^{-1}, d_{z,i}^{-1}\right). \tag{4.8}$$

The normalization matrix is defined for each floor cell FC_i and takes into account the geometrical extent of the cell itself. $d_{x,i}$ and $d_{y,i}$ are the dimensions of the FC_i convex-hull, while $d_{z,i}$ is the maximum variation of the GRF intensity and is estimated during a calibration phase of the capturing board. As a consequence of the random spatial distribution of the sensors, even a single person may trigger more than one floor cell. Thus, a cluster of neighbor floor cells will switch to a non-equilibrium condition for each person located on the sensing floor. The temporal analysis and tracking of these clusters allows to detect and track people on the floor, as detailed in the following section. A visual example is reported in Figure 4.9. The sensor values captured at the equilibrium and with a walking person are reported on the left and center graphs. On the right, a plot of the vectors $P_i(t) - P_i^{eq}$ evaluated in equation 4.7 are shown.

4.2.2 People Detection and Tracking on COP models

Let $C_j(t) = \{FC_k\}$ be a cluster of neighbor floor cells which are simultaneously excited at time t. Eq. 4.7 filters out contributions due to the noise and assures that the cluster has been generated by a person on the floor. His position $(B_j^x(t), B_j^y(t))$ on the floor at time t can be estimated from the 3D points P_i associated to the floor cells included in the cluster as in equation (4.9):

$$\begin{aligned} M_j(t) &= \sum_i P_i^z(t) \\ B_j^x(t) &= \frac{1}{M_j(t)} \sum_i P_i^x(t) \cdot P_i^z(t) \\ B_j^y(t) &= \frac{1}{M_j(t)} \sum_i P_i^y(t) \cdot P_i^z(t) \end{aligned} \quad , \tag{4.9}$$

The set of clusters is obtained with a connected component labeling of all the excited cells. Two FCs are defined as connected if their intersection contains at least one sensor. For a uniform grid distribution, this assumption is similar to the 8-connection of pixels (See Fig. 4.8(b)).

Successive detections of the same person are temporally tracked with a nearest neighbor matching based on positions only. The main purpose of the tracking step is the recovering of people's positions in some short temporal slots, during which the floor is not able to detect them. For example, when a person changes the front feet during a walking or, more clearly, during a jump, the pressure

exerted on the floor is null.

Given the detections D_i at frame t and the current set of tracks T_j, we first compute the Euclidean distance matrix $\Gamma(i,j) = D_2(D_i, T_j)$. The detection-to-track association is provided using the schema proposed in section .

4.2.3 Experimental Evaluation

To evaluate the proposed method, we have exploited a *Florim Age Device* distributed by Florim Ceramiche SpA [1] and described in [26]. The device is covered by tiles of $600\,mm \times 600\,mm$, thin enough ($4.5\,mm$) to allow the sensing elements below them to capture the presence of walking people. The sensors are distributed on a regular grid of 16 rows by 20 columns. The 320 sensing units covers a rectangular area of 4 square meters.

The experimental environment is also equipped with a Microsoft Kinect sensor. The acquisition of the floor data and of the Kinect sensor are synchronized with an external trigger (set to work at 10Hz in these experiments). The people detection and tracking capabilities of the Kinect subsystem have been exploited to automatically generate the ground truth, composed by the set of people positions on the floor. The Extrinsic calibration parameters of the

[1] http://www.slim4plus.it/en/floor-sensor-system/

Kinect device with respect to the floor have been estimated to transform the 3D coordinates of people feet joints into floor coordinates.

Using the experimental setup, 6 data sequences[2] of walking people have been acquired, involving 1 to 7 individuals. People weights are also ranging from 45 Kg to 100 Kg.

The algorithm has been evaluated counting the number of True Positive (TP), False Positive (FP) and False Negative (FN) detections. A detection provided using the algorithm reported in Section 4.2.2 is counted as a TP if the Kinect sensor provides a corresponding person position closer than 25 cm. The precision $Pr = TP/(TP + FP)$ and accuracy $Ac = TP/(TP + FN)$ metrics are also computed as usual.

Thanks to the regular distribution of the sensing elements on a grid, $M = 285$ floor cells FC_i of 4 sensors each have been generated using the schema reported in Figure 4.8. For each floor cell FC_i, the coordinates of the equilibrium state point P_i^{eq} were estimated by averaging a short sequence of sensor values captured with the empty floor. The threshold TH of Equation 4.7 has been set to 0.65 by maximizing the precision and recall of the detection algorithm. The corresponding ROC curve obtained at different threshold values is reported in Figure 4.10.

[2]Dataset available at http://imagelab.ing.unimore.it/go/sensingFloor

TABLE 4.4: Detection algorithm results

	Seq1		Seq2		Seq3		Seq4		Seq5		Seq6	
	PIM	COP	PIM	COP	PIM	COP	PIM	COP	PIM	COP	PIM	COP
Nframes	2353		1936		1504		1060		1311		1901	
Precision	0.79	**0.91**	0.53	**0.87**	0.77	**0.95**	**0.79**	0.76	0.79	**0.89**	0.71	**0.87**
Accuracy	0.86	**0.92**	0.53	**0.86**	0.78	**0.96**	**0.83**	0.81	0.81	**0.91**	0.65	**0.84**
TP	1245	1204	1333	1253	1287	1286	699	699	1028	1027	1793	1921
FP	326	121	1161	192	393	65	183	218	280	126	737	285
TN	861	1024	18	479	87	173	210	220	157	271	88	314
MD (cm)	20.68	**20.44**	24.56	**20.37**	17.37	**14.63**	19.70	**19.02**	**15.40**	15.69	21.03	**19.11**

TABLE 4.5: Tracking algorithm results

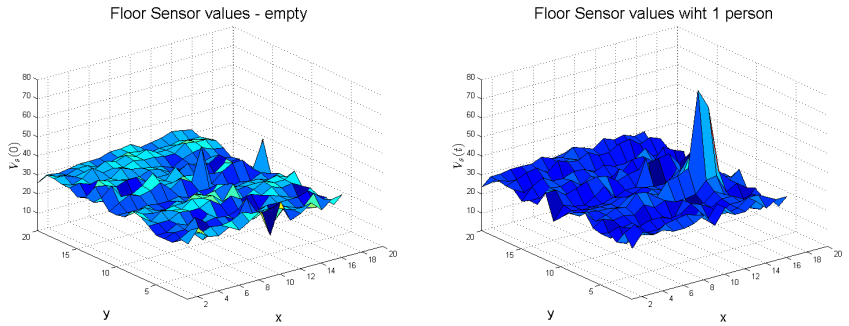
	Seq1		Seq2		Seq3		Seq4		Seq5		Seq6	
	PIM	COP	PIM	COP	PIM	COP	PIM	COP	PIM	COP	PIM	COP
Nframes	2353		1936		1504		1060		1311		1901	
Precision	0.84	**0.94**	0.57	**0.90**	0.80	**0.96**	**0.82**	0.79	0.83	**0.91**	0.75	**0.90**
Accuracy	0.88	**0.94**	0.56	**0.88**	0.81	**0.97**	**0.85**	0.83	0.85	**0.92**	0.68	**0.86**
TP	1232	1203	1305	1249	1284	1286	698	699	1026	1027	1768	1920
FP	239	82	999	143	326	48	157	190	213	105	598	223
TN	905	1027	35	482	95	174	217	221	173	272	106	318

We tested and compared the COP method with respect to the PIM algorithm on the six sequences. Table 4.4 and Table 4.5 report the values of all the estimated performance parameters for each sequence, with or without the tracking stage, respectively. In each table, the best results in term of precision and accuracy are highlighted. The precision and accuracy obtained with the COP model are higher than those obtained with the PIM one, except for the 4-th sequence.

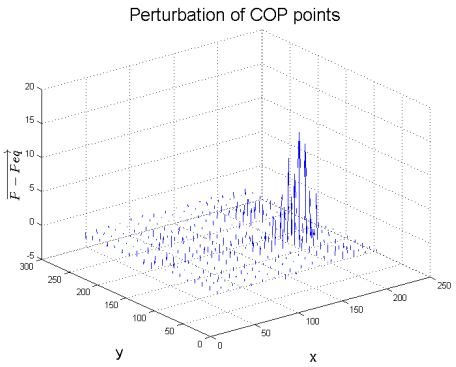
The mean distance between the detection and the ground truth positions is also reported in the last row of Table 4.4. The closest

mean distances are those obtained with the COP based detection algorithm.

(a) (b)



(c) (d)



(e)

FIGURE 4.9: Visual example of the COP vectors. (a) and (b) pictures of the empty floor and with a walking person. The sensor values captured at the equilibrium (c) and with the walking person (d). On the right (e), a plot of the vectors $P_i(t) - P_i^{eq}$ evaluated in equation 4.7.

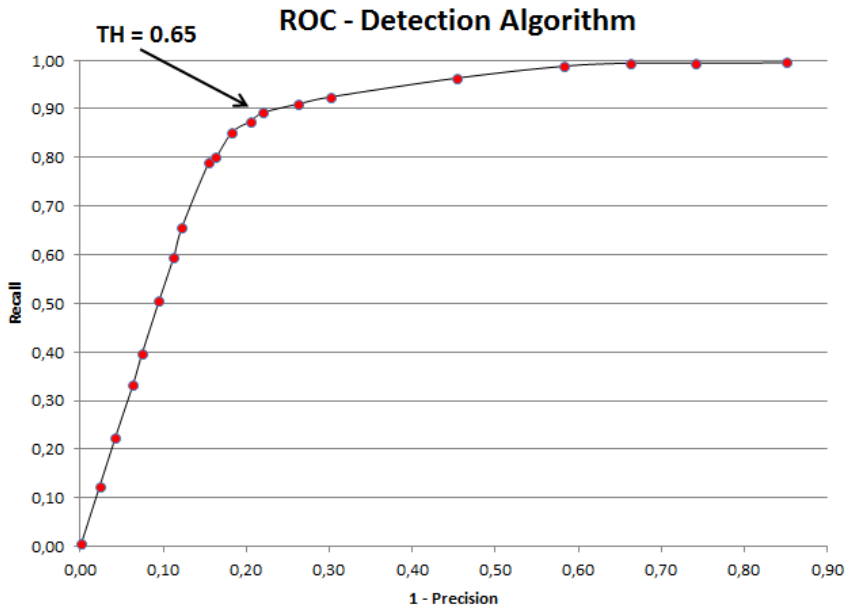


FIGURE 4.10: ROC curve at different thresholds of the detection algorithm proposed in Section 4.2.2 on a calibration sequence with a person of 45 Kg on the sensing floor.

Chapter 5

High-Level Applications

The low-level processing described in Chapter 4 generates events and commands to be captured and exploited by the high-level applications, which are basically unaware of the physical details of the capturing device. This general and flexible architecture enables a plethora of applications, ranging from surveillance to entertainment. The "camouflage" of the sensors beneath the tiles plays a fundamental role in the development of both collaborative and non-collaborative tasks (See Table 5.1).

The collection of data for offline processing, statistical analysis and logging are the basic features required by non-collaborative applications, such as surveillance or people flow analysis inside marketplaces or exhibitions. Introducing a more sophisticated processing level, specific events and information can be extracted in real-time to trigger alarms or actions. For example, the automatic detection of a queue at the counter may alert additional cashiers.

In the collaborative case, instead, the user explicitly wants to interact with the system by means of the floor, expecting a real-time feedback to the action or movement done over the floor. We have divided the applications in two categories, depending on the level of interaction and the corresponding complexity of the processing engine. Basic applications only require the instantaneous position of the people as input, while more data are needed in the second category.

This chapter describes some real use cases related with real-time surveillance, human-computer interaction, and zone statistics. In the first case, algorithms for action recognition and human behavior understanding have been proposed and tested. In the second case, the floor has been used as input device for games and interactive applications. Finally, the sensing floor technology has been exploited to develop people counting and zone statistics systems.

5.1 Use case: People Surveillance

Traditional surveillance systems are composed of a network of visual sensors and one or more processing and storage nodes. Despite their large diffusion, two major drawbacks still characterize vision-based systems. First of all, the complete coverage of wide areas is reached only using a redundant number of cameras.

TABLE 5.1: Sensing floor application taxonomy.

Non Collaborative	
Data Collection	*Real Time Event Detection*
Offline processing Statistical analysis Log data	Surveillance People counting Statistical Flow analysis
Collaborative	
Basic Interaction	*Full Interaction*
Single person position detection Simple event detection [e.g., jump]	Human Behavior Understanding Multiple person interactions

Placement constraints and the handling of occlusions due to furniture, objects and people usually impose the adoption of multiple views, especially in indoor environments. Furthermore, privacy issues strongly limited the usability and also the user acceptability of surveillance systems. Completely automatic infrastructures only mitigate the problem by storing and exploiting anonymised or aggregate data. The alternative solution obtained with sensing floors is cheap enough to allow the coverage of wide areas, the sensors do not change the design or the appearance of the floor. In addition, the data collected from the sensors do not contain identifying elements such as faces or biometric details, assuring a complete compliance with the user privacy.

Snapshots of surveillance demo applications are reported in Figure 5.1. In the first two images (Figure 5.1(a) and Figure 5.1(b)),

the floor is used to detect people inside a safe area and to trigger the Microsoft Kinect sensor. Visual data are not captured and stored if people do not enter the safe zone. In the second case, using high level information from the the tracking and action recognition steps proposed in chapter 4.2.2 and 4.1.4, the floor is used to detect (abnormal) behavioral patterns and to activate rescue procedures in case of people falls. The Florim Age system can be integrated into home environments to assist old or disabled people. Its purpose is to monitor the inhabitant's position and behavior. The presence of a lying person, combined with the detection of additional people in the same room or home, can be exploited to trigger internal alarms as well as calls for assistance request (see Figure 2.4).

5.2 Use Case: Human-Computer Interaction

In addition to explicit HCI interfaces based on voice, vision and touchscreens, sensing environments are becoming an effective and feasible way to interact with virtual worlds. Sensing environments bypass the use of non-natural input devices such as keyboard and mouse, that force users to learn the computer interface paradigm. In addition, they can also avoid the need of semi-natural interfaces like touchscreens and make humans completely free to move and interact with virtual or physical items and with other humans. To

test the capability of the sensing floor as input device for interactive and collaborative interfaces, we have created four sample applications described in the following.

Virtual Street View: the setup of the virtual street view is composed by a pad made with the sensing floor technology surrounded by projection screens. The position of a user within the floor controls a "Virtual Street View" journey. After the selection of a starting and destination position on a Google Map (see Fig. 5.2(a)), frontal and lateral frames of the entire route are captured from Google Street View. As the user walks toward the frontal screen, the projected frames depict a place closer to the path destination (see Fig. 5.2(b)). Walking back, the views are collected from places closer to the starting position, instead.

The application is based on three modules: the OSC client, the web based application, and the bridge within the two previous units (see Fig. 5.3). The OSC client interacts with the OSC server (i.e., the low-level processing unit) to receive the user location on the floor. In turn, the client forwards the current position and corresponding events to the web browser thanks to the bridge, which exploits the *node.js* [51] and the *socket.io* frameworks [40]in order to establish suitable socket ports.

The web based application recovers the journey frames using the Google Maps Javascript APIs [14] and visualizes the output

on a set of browser windows using the emerging web-standard *WebGL*.

Human Media Player: the setup is the same of the Foot painting application. A video or a picture sequence is projected on the screen. The position of the user on the floor controls the current media playing position. The application increases or decreases the video player speed as the user approaches the screen or moves further away from it. Moreover the user can select a video or a picture from a list of them simply moving right or left. A snapshot of the human Media Player is reported in Figure 5.4(a).

Virtual Navigator: the setup of the Virtual Navigator is the same of the Foot Painting Application. The sensing floor is partitioned into nine regions corresponding with cursor control keys of a joypad (see Figure 5.4(b)). After a brief training stage in which the user learn how to interact with the application (Figure 5.4(c)), his position on the floor allows to control movements of a character into a 3D virtual environment (Figure 5.4(d)). The 3D framework was developed exploiting Unity resources as game engine which is distributed by Unity Technologies [19].

Foot painting: a projector screen is placed near to the sensing floor. A famous painting or a picture is selected and partitioned

into small rectangular portions. Each of them is associated to a region on the floor. The position of a detected person triggers the projection of the corresponding image portion for few seconds. The user is thus encouraged to quickly walk all around the floor to unveil the entire image. Some snaps from the described application are reported in Figure 5.5. The top row shows pictures of the sensing floor and of a user walking on it. The graphs in the central row depict the sensor responses, while the output of the revealed painting is reported on the last row. Since the last frame (d) has been acquired few seconds after the previous one (c), the painting areas activated in (j) and (k) are no more visible in the corresponding final output (l).

Since entertainment applications were proposed on real museum installation and on public conferences and showrooms we had the possibility to test their performance. Virtual reality frameworks are difficult to evaluate quantitatively, since it is not possible to define a ground-truth and a corresponding metric. Systems are usually evaluated by means of user interviews, where the *presence* of interaction is measured with subjective rates. We have followed the definition and the protocol provided by Witmer and Singer [64], extracting 11 questions among the 32 proposed. The unselected questions are referred to applications with audio interactions, which are not included in our setup. The survey has been

submitted to a set of 50 users (having different weight, age and sex), after an interaction session with the prototype. The users provided a score on a scale from 1 to 7 to each question, where a lower score corresponds to a worse presence. Table 5.2 reports the eleven questions together with the average scores assigned by the users.

The questions mainly belongs to two categories. On the one hand, the first category contains questions related to the control capabilities and are marked as *c* in the third column of the table. These questions evaluate how easy and naturally the user can control the application using the floor. The second category of questions, on the other hand, are more related to the subjective experience and are marked with the symbol *p* in the table.

The high average rate of questions related to the control confirms the performance of the whole system and in particular of the proposed sensing floor. Moreover, the score of question Q11 is 2.0, which indicates the low interference of the device with the virtual performance. However, the users have indicated a moderate presence rate, as highlighted by the corresponding scores to Q4 and Q8Probably, the crowding situation of the presentation events and the limited quality of the visual setup have negatively influenced the virtual immersion (see questions Q7 and Q10).

TABLE 5.2: Presence questionnaire.

Q1	How much were you able to control events?	*c*	6.6
Q2	How responsive was the environment to actions that you performed?	*c*	6.7
Q3	How natural did your interactions with the environment seem?	*c*	6.5
Q4	How much did the visual aspects of the environment involve you?	*p*	4.1
Q5	How natural was the mechanism which controlled movement through the environment?	*c*	6.4
Q6	Were you able to anticipate what would happen next in response to the actions that you performed?	*c*	6.5
Q7	How compelling was your sense of moving around inside the virtual environment?	*p*	5.8
Q8	How involved were you in the virtual experience?	*p*	4.5
Q9	How proficient in moving and interacting with the virtual environment did you feel at the end of the experience?	*c*	6.8
Q10	How much did visual display quality interfere or distract you from performing assigned tasks or required activities?	*p*	6.0
Q11	How much did the control devices interfere with the performance of assigned tasks or with other activities?	*c*	2.0

5.3 Use Case: Zone Statisics

Always more often smart cameras are used as people counter devices to measure the number and direction of people traversing a certain passage or entrance in public environment. For example, people counting systems in the retail environment are used to calculate the *conversion rate*, i.e., the percentage of visitors that make purchases. This is an important performance indicator for a store which provides more valuable information than traditional methods, which only take into account sales data. Together, traffic counts and conversion rates can reveal important sales information, such as why a specific store is experiencing more sales, whether or not year-over-year sales are down, why fewer people are visiting a store, or why fewer people are making purchases. Moreover, museums and libraries use visitor counts as evidence when making applications for finance, for use when planning for seasonal staffing, and other strategic operational decisions. In cases where tickets are not sold, counting is either automated, or staff keep a log of how many clients use different services. Despite their large diffusion, this kind of systems are characterized by the same occlusion and coverage drawbacks listed in section 5.1 for vision-based surveillance systems. For this reason the Florim Age solution with its Zone Statistics application can be considerate an optimal alternative solution to the problem.

If we consider an environment (a public building, a museum, a shop, etc.) equipped with the Florim Age system with an architecture similar to that proposed in chapter 3, the whole environment can be divided into sub-floors which sizes are those of the building rooms. At each sub-floor corresponds a floor pressure image and can be identified by means of a specific id number assigned by the associated local processing unit. As we have just observed, for each sub-floor a real-time people detection and tracking surveillance is ever possible, but the Zone Statistics application can enrich information content from the floor data adopting some simple statistical strategies.

During the day, for each sub-floor after each time t (usually 1 minute but this time can be manually set by the user) each local processing unit send to a database hosted in the server a data packet which contain aggregate information according to the following format:

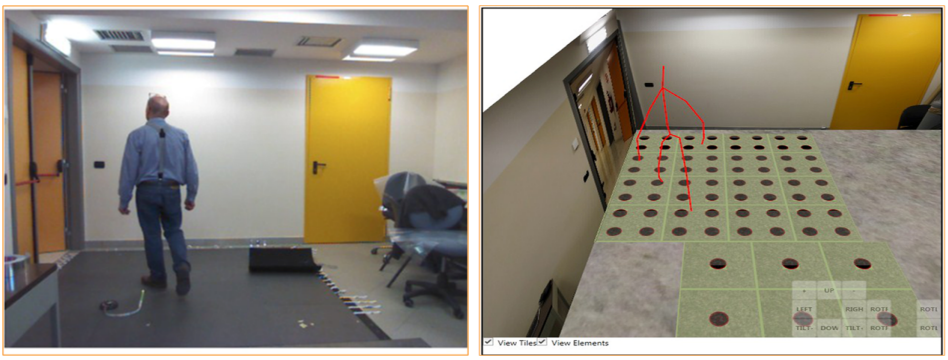
- **Data-type**: a string which specifies one of the four possible data types. The "raw" type is an image obtained as sum of the images sampled during the acquisition time t. The "bksup" is the sum of the foreground images obtained after the background suppression during the acquisition time t. Finally, the "det" and the "track" types are images in which each pixel represents the percentage of the acquisition time in which the

corresponding floor image pixel was respectively involved into the detection or the tracking of a person.

- **Time-stamp start**: a string which specifies the acquisition start time.
- **Time-stamp stop**: a string which specifies the acquisition stop time.
- **Frame-count**: the number of frames acquired during the acquisition time t.
- **Duration**: the acquisition time duration in ms.
- **Data**: values of data processed according to the data type.
- **Id**: the id number of the analyzed sub-floor.
- **Width**: this attribute specifies the width of the analyzed floor image.
- **Height**: this attribute specifies the height of the analyzed floor image.

Since the sum of the pixels values of a "track" image corresponds with the number of tracked people inside the analyzed room during the specific acquisition time, by means of the Zone Statistics application, which can recall stored information from the server,

it is possible to know minute-over-minute the number of people tracked into each room of the building. Moreover, selecting a start and stop date analysis, the Zone Statistics application performs a sum of all the "bksup" images collected in this time interval to obtain a heatmap which strongly characterize people movements inside the analyzed room. At the end of each day, the Zone Statics application fetches the data from the "daily database" and re-aggregates them using the same data-type but coarser time windows (e.g. 1 hour). This approach allows to summarize information and reduce the overall amount of data that must be kept for longer periods of time to allow further analysis. This newly aggregated information is stored in a second database referred to as "long term database", which can be queried by users through the Zone Statistics application. In Figure 5.6 a Snapshot of the application.

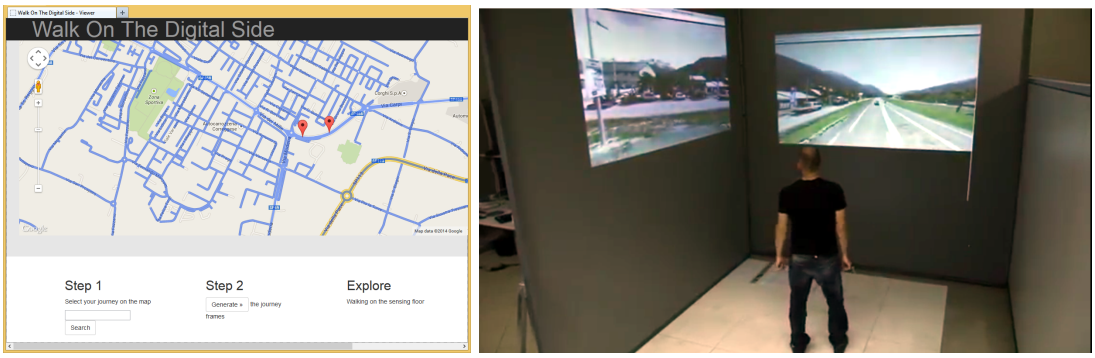


(a) (b)

(c) (d)

(e)

FIGURE 5.1: Surveillance applications. The floor detects people inside safe areas and it triggers the capturing of visual details.(a) the input image, (b) a virtual reconstruction with the skeleton of the user captured using the Kinect device. (c), (d) The presence of a person is detected and an alarm is triggered after a fall detection (e).



(a) (b)

FIGURE 5.2: The Street View application: a walking on the sensing floor becomes a virtual journey. a) The journey selection and b) the virtual tour.

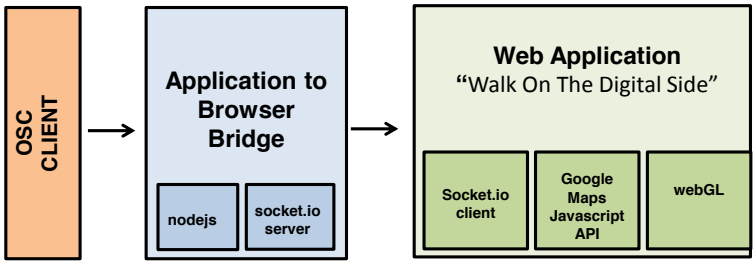
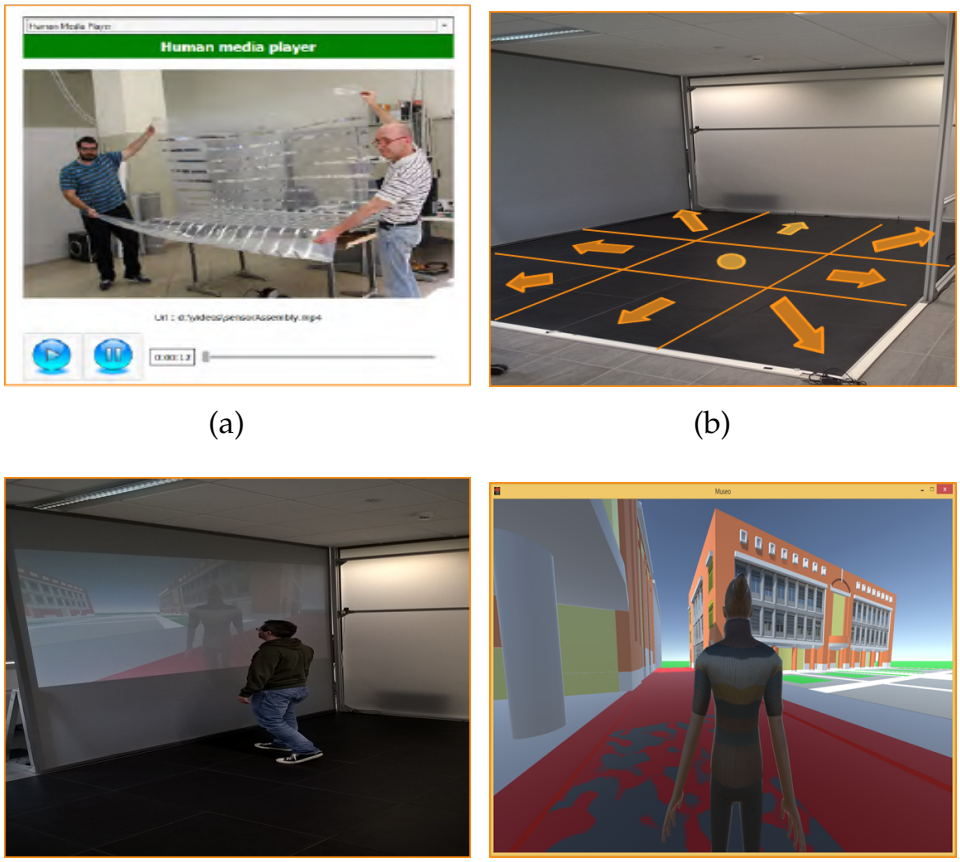


FIGURE 5.3: The Virtual Street View modules: the OSC client and the web based application, connected through the node.js bridge.



(a) (b)

(c) (d)

FIGURE 5.4: Human-Computer Interaction applications. (a) Screenshot from the Human Media Player Application. (b) For the Virtual Navigator the sensing floor is partitioned into nine regions corresponding with cursor control keys of a game-pad. (c) A user can control movements of a character into a 3D virtual environment (d).

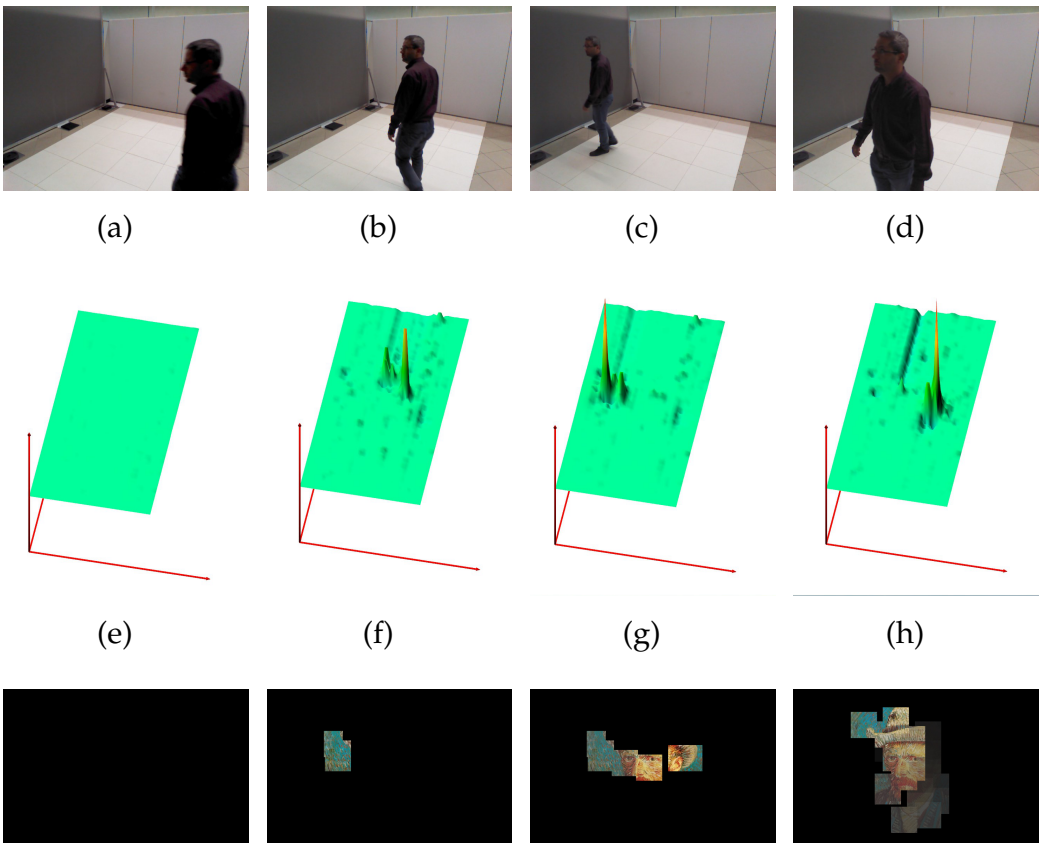


FIGURE 5.5: Four frames from the foot painting demo. Top row: input image from a camera. Central row: a graph of the sensor values. Bottom row: the application output.

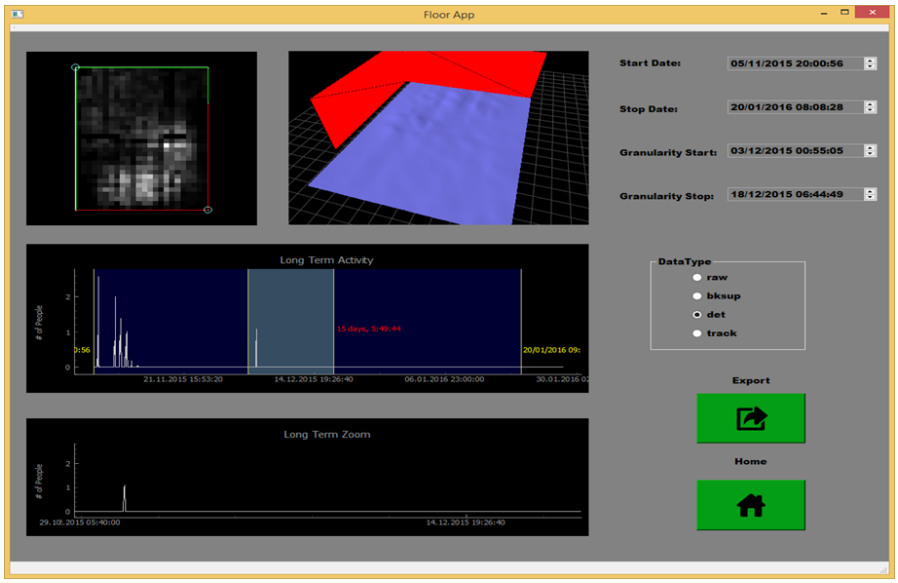


FIGURE 5.6: Zone Statistics Application. A Snapshot of the Zone Statistics application: 2D and 3D Heatmaps of the analyzed room on the top, a graph with the number of tracked people as a function of time on the bottom.

Chapter 6

Floor Configuration

Despite the last version of the sensing floor has been developed to allow an easy and fast hardware installation, two software calibration steps are still mandatory in order to exploit the sensor data, in particular if multiple modules are required. This calibration should be done after the floor placement and cannot be performed in laboratory once for all.

The first requirement is the assessment of the spatial configuration of the installed modules, in order to correctly merge the contribution of each one. Then, a calibration of each sensor is needed to recover the specific range of each item, which depends on several factors including the position of the tiles and the planarity of the basement.

In this chapter, we propose and describe an automatic system to estimate the *Sensor Calibration* and the *Floor Layout* configuration

parameters (red parallelograms in Figure 4.1) directly from training data.

6.1 Sensor calibration

The calibration step is carried out at sensor level. No spatial relations are exploited. Due to several factors, each sensor may provide values within different output ranges. Figure 6.1 shows the temporal response of a sample sensor. The graph on the top plots the temporal sequence of sensor responses. During the sequence, a person walked over the sensor about ten times, generating the same number of positive peaks. When the person raises his feet from the floor, the corresponding sensor provides negative peaks, since the sensor coverage is subjected to a temporary swing behavior. Similar responses are also generated if the coverage is somehow unsteady or if a person is walking near to the sensor. For the rest of the time, the sensor output is related to the dead load and is perturbed by a Gaussian measurement noise.

We dealt the sensor calibration as a classification problem. Given a new sample from a sensor, we want to classify it as belonging to one of the following classes: *negative-peak*, *background*, and *positive-peak*. A mixture of three Gaussian distributions is fitted to the data captured from each sensor as a training sequence. The parameters obtained using the Expectation Maximization (EM)

algorithm are stored and used as input for a Bayesian classifier, which also allows a binarization of the floor data into background-foreground masks (positive peaks are treated as foreground points, and the others as background). In addition, the parameters of the mixture are exploited to compute the linear transformation parameters α and β of Eq. 4.1.

Given a data sequence $\{\psi(x,y,t)\}_{t=1...T}$, the EM fitting algorithm returns the 3D mean and variance vectors $\mu = [\mu_1\ \mu_2\ \mu_3]$ and $\sigma = [\sigma_1\ \sigma_2\ \sigma_3]$, corresponding to the three components depicted in the bottom graph of Figure 6.1:

$$\begin{aligned} &\hat{\psi}(x,y,t) = \alpha_{x,y} \cdot \psi(x,y,t) + \beta_{x,y} = \frac{\psi(x,y,t)-L_1}{L_2-L_1} \\ &L_1 = \mu_2 - \sqrt{\sigma_2}, \quad L_2 = \mu_3 + 2 \cdot \sqrt{\sigma_3} \end{aligned} \tag{6.1}$$

where $[L_1, L_2]$ is the normalized range of each sensor output. The indexes x, y have been omitted for the sake of readability. In Figure 6.1.top the mean of the Gaussian component related to foreground values is drawn in green, while the automatic threshold used by the classifier to distinguish between background and foreground points is drawn in red. The Threshold Th is set as the intersection value between the background and the foreground Gaussian curves (see Figure 6.1.bottom).

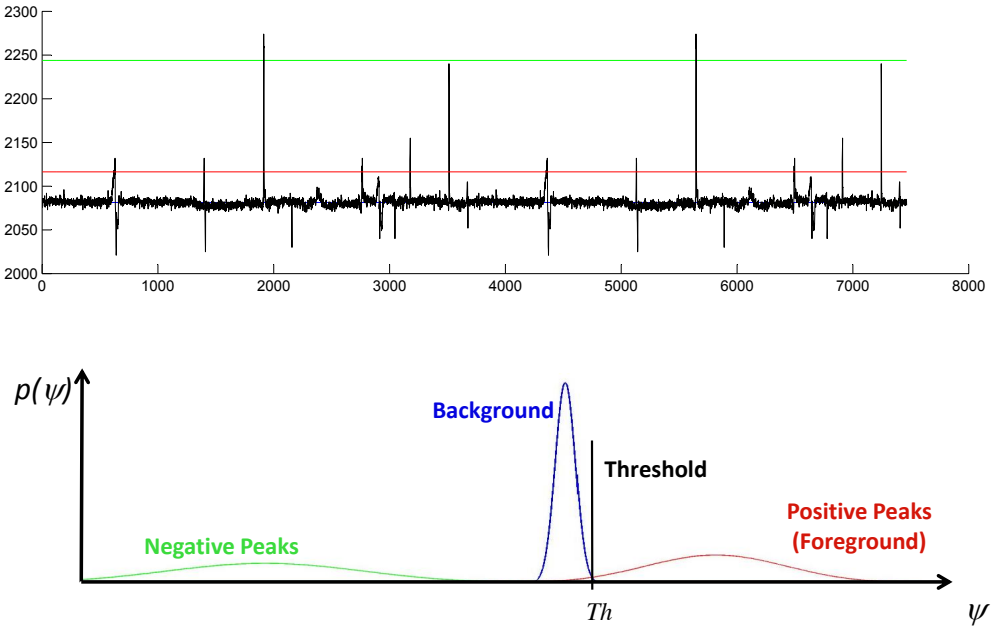


FIGURE 6.1: Plot of a real sequence of data captured from a single sensor (top). The EM fitting algorithm automatically recovers the three Gaussian distributions shown in the sketch (bottom).

6.2 Layout discovery

The layout discovery algorithm is based on the work of Pomeranz *et al.* [34]. They proposed a system to automatically solve a jigsaw puzzle. After splitting an image into a scrambled set of tiles, the algorithm find the correct position occupied by each in the source

image. The method trusts on the similarity at the borders of neighbor tiles. Thus, key elements of the approach are a suitable compatibility measure between couples of tiles and a greedy optimization algorithm to reduce the computational complexity. In fact, the brute force approach is not allowed due to the factorial complexity of the problem with respect to the number of tiles.

Similarly to the jigsaw puzzle, we need to discover the correct position of each portion of the floor (floor tile). To this aim, we can exploit the similarity of the temporal data sequences captured by the sensors placed near to the edges of the tiles. Given two random floor tiles $\Psi_i, \Psi_j \in \Psi$, we need a metric to measure their compatibility given a spatial relation $R \in \{l, r, u, d\}$ (i.e., left, right, up, down) between them. In other words, we need a suitable compatibility metric which indicates how likely the tile Ψ_j is placed on the left/right/up/down side of the reference tile Ψ_i, based on a training data.

If the spatial distance between two neighbor sensors is low enough, the correspondent temporal series of the captured data will be highly correlated. This correlation is enhanced thanks to the ceramic tiles, which spatially smooth the sensor response. Let $C(\Psi_i, \Psi_j, R)$ be a compatibility metric estimated between the two tiles, given the spatial relation R. The compatibility of two tiles can

be computed as a function of their dissimilarity [34]:

$$C(\Psi_i, \Psi_j, R) \propto \exp\left(-\frac{D(\Psi_i, \Psi_j, R)}{quartile(i, R)}\right), \tag{6.2}$$

where $quartile(i, R)$ is the quartile of the dissimilarity between all other parts in relation R to part Ψ_i.

In case of images, the dissimilarity $D(\cdot)$ of two tiles can be measured by summing up the color differences of the pixels along the parts' abutting boundaries [9]. Since floor tiles are temporal sequences, their dissimilarity metric could take into account temporal as well as spatial data missing correlations along the parts' abutting boundaries.

6.2.1 Dissimilarity metrics

The first *Dissimilarity metric* D_1 for floor data can be defined as following, where the right ('*r*') relation is used:

$$D_1(\Psi_i, \Psi_j, r) = \sum_{t=1}^{T} \sum_{x=1}^{K} d\left(\psi_i(x, K, t), \psi_j(x, 1, t)\right), \tag{6.3}$$

where $d(\cdot, \cdot)$ is a distance function between two sensor values. The Euclidean distance has been used to this aim in our system, but more complex may improve the results in some particular cases [34].

Figure 6.2 shows the subset of sensors (colored in green) used in the computation of the Dissimilarity metric in the case of *right* relation.

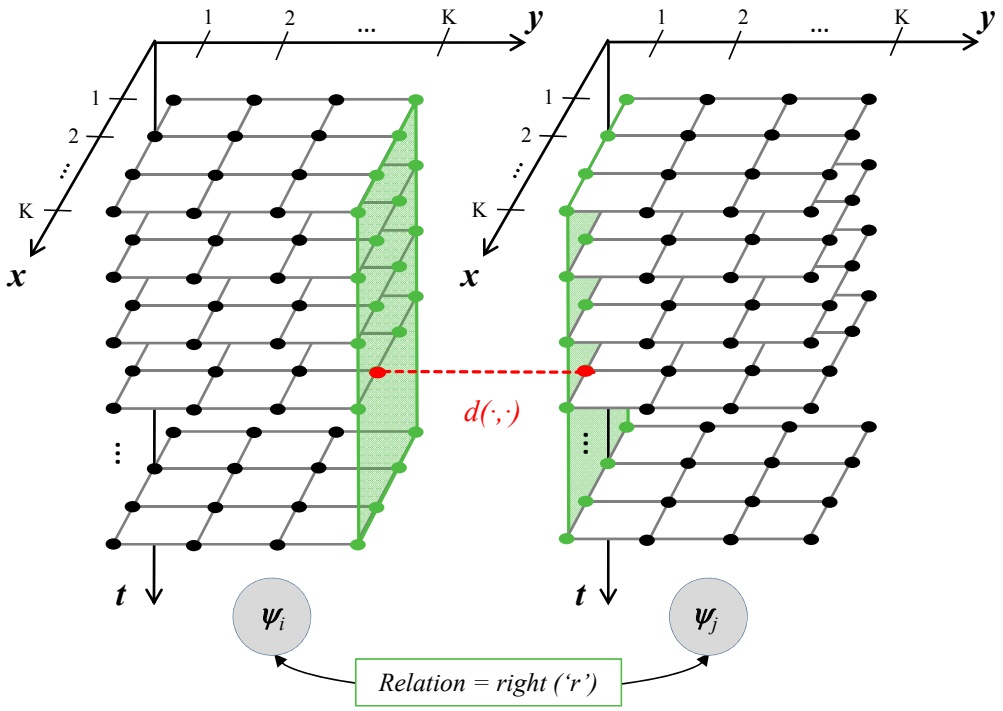


FIGURE 6.2: Computation of the Dissimilarity metric in the case of *right* relation. The temporal series of each sensor colored in green are taken into account.

The Dissimilarity metric D_1 does not take into account some useful information available thanks to the automatic calibration step described in Section 6.1. Thus, we propose and test other two dissimilarity metrics. Exploiting the linear filtering, Equation 6.3

can be modified to compute the distances between the normalized sensor values:

$$D_2(\Psi_i, \Psi_j, r) = \sum_{t=1}^{T} \sum_{x=1}^{K} d\left(\hat{\psi}_i(x, K, t), \hat{\psi}_j(x, 1, t)\right). \tag{6.4}$$

Furthermore, the automatic calibration step indicates the best threshold values to classify each floor data element into background or foreground. If available, the binarized values of Equation 4.2 allow to define a suitable Dissimilarity metric D_3:

$$D_3(\Psi_i, \Psi_j, r) = 1 - \frac{CoOcc(F_i, F_j, r)}{Occ(F_i, F_j, r) + \epsilon}, \tag{6.5}$$

where $CoOcc$ and Occ are the number of *CoOccurrences* and *Occurrences* of foreground pixels at the selected borders of Ψ_i and Ψ_j and are computes as:

$$\begin{aligned} CoOcc(F_i, F_j, r) &= \textstyle\sum_{t,x} F_i(x, K, t) \wedge F_j(x, 1, t) \\ Occ(F_i, F_j, r) &= \textstyle\sum_{t,x} F_i(x, K, t) \vee F_j(x, 1, t), \end{aligned} \tag{6.6}$$

where the sum limits are omitted for the sake of readability. The ϵ term as been included in Equation 6.5 to avoid division by zero.

The dissimilarity D_3 is related to the number of times two neighbor sensors are contemporaneously excited (i.e.,when they

are synchronously stepped by a person), but it does not handle temporal causalities (i.e., when they are stepped by a person one immediately after the other). Common correlation and convolution filters are able to discover uni-directional relations and are time consuming. The alternative we propose is based on binary morphological operations. In particular, we have applied a binary dilation of the foreground multidimensional matrices F using a cubic structuring element of size 3. In addition to temporal causalities, the dilation operation allows to spatially enlarge the neighborhood of each sensor during the computation of the number of *CoOccurrences* and *Occurrences*. In the experimental section, we call D_4 the Dissimilarity metric computed as in Equation 6.5 but on the dilated foreground matrices.

6.3 Experimental evaluation

The automatic configuration and calibration system has been tested on different sequences captured from a FlorimAge setup installed in our laboratory. The floor is composed by 12 independent modules of 8x8 sensors each, placed as in Figure 6.3. The whole floor data is thus a grid of 32x24 sensors. To increase the number and difficulty of the tests, we have also split each modules into many finer modules. In particular, we tested the system with $K = 2,\ 4$ and 8, generating $M = 12,\ 48$ and 192 blocks, respectively. The

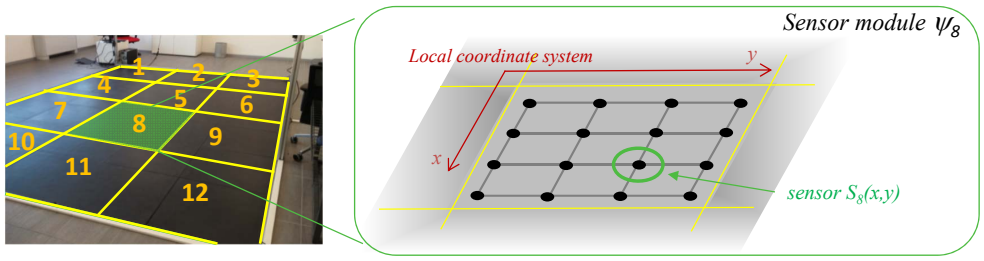


FIGURE 6.3: An example of a sensing floor composed by 12 modules. Each module is made of a grid of sensors $S_m(x, y)$.

blocks are randomly scrambled before the tests, keeping their original position as ground truth.

We have collected 14 sequences, characterized by different durations, people weight, count and motion patterns. These sequences were collected emulating real conditions. For example, to simulate a calibration phase during a sensing floor installation, we collected 2 sequences with a person following a preset pattern to completely scan all the sensor units under the floor. To simulate a crowd situation, 9 sequences with walking/standing people have been acquired, involving 1 to 6 individuals randomly moving on the sensing floor. The individuals were also free to leave the sensorized area. Finally, to emulate the use of the system in points of passage such as corridors or gangways, 3 sequences have been collected with people walking according to a predefined unidirectional flow. Table 6.1 reports a summary of the adopted dataset.

TABLE 6.1: Dataset summary

# seq.	Duration[1] [mm:ss]	Duration[1] frames	People count [2]	Description
1	06:13	3731	1	Complete scan of the floor by a single person
2	05:05	3051	1	
3	02:23	1436	2	Unidirectional flow of people
4-5	01:00	600	1	
6	00:23	236	1	Random walk of a single person
7-11	01:17	776	1	
12-13	01:50	1200	2	Random walk of groups of people
14	02:15	1356	6	

[1] For aggregate rows, duration and frames are average values.
[2] Number of people simultaneously on the floor

Performance evaluation metrics: the *Direct Comparison* (DC) is the most important metric for the evaluation of the proposed system. Given an output layout assignment, the DC value indicates the fraction of blocks correctly placed over the total number of blocks. A perfect match leads to a DC value equal to 1. As stated in [9], this score is not complete and can be misleading. DC is equal to zero even if the layout has been shifted by one row or column only. In these cases, the *Neighbor Comparison* (NC) metric provides a better performance evaluation. NC is computed as the fraction of mutual relations correctly found between couples of blocks.

Figures 6.4 and 6.5 shows the DC and NC scores obtained on each sequence using the four proposed metrics D_1-D_4. The correct layout is always discovered using the original block size ($K = 8$)

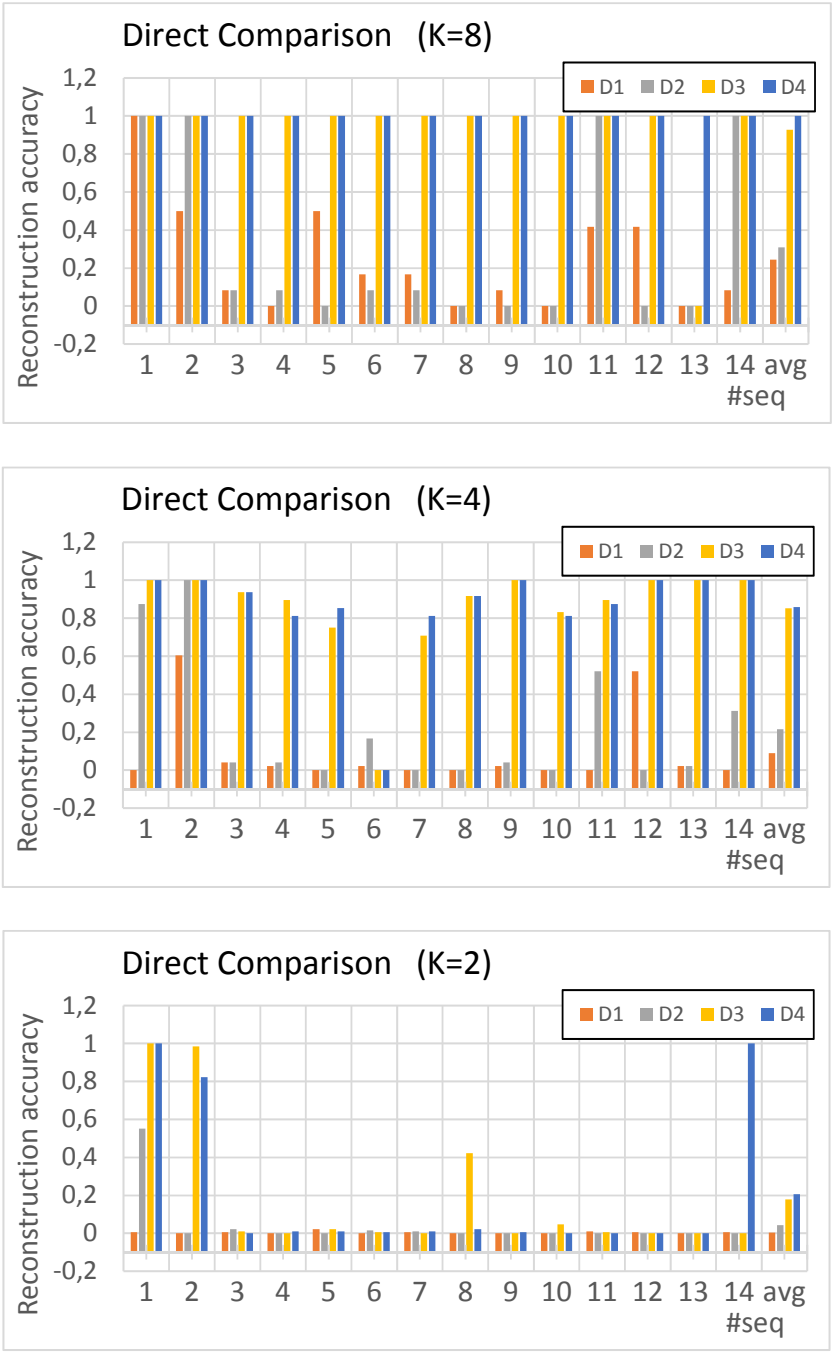


FIGURE 6.4: Performance evaluation of the proposed system in terms of DC metric for each video sequence and block size K.

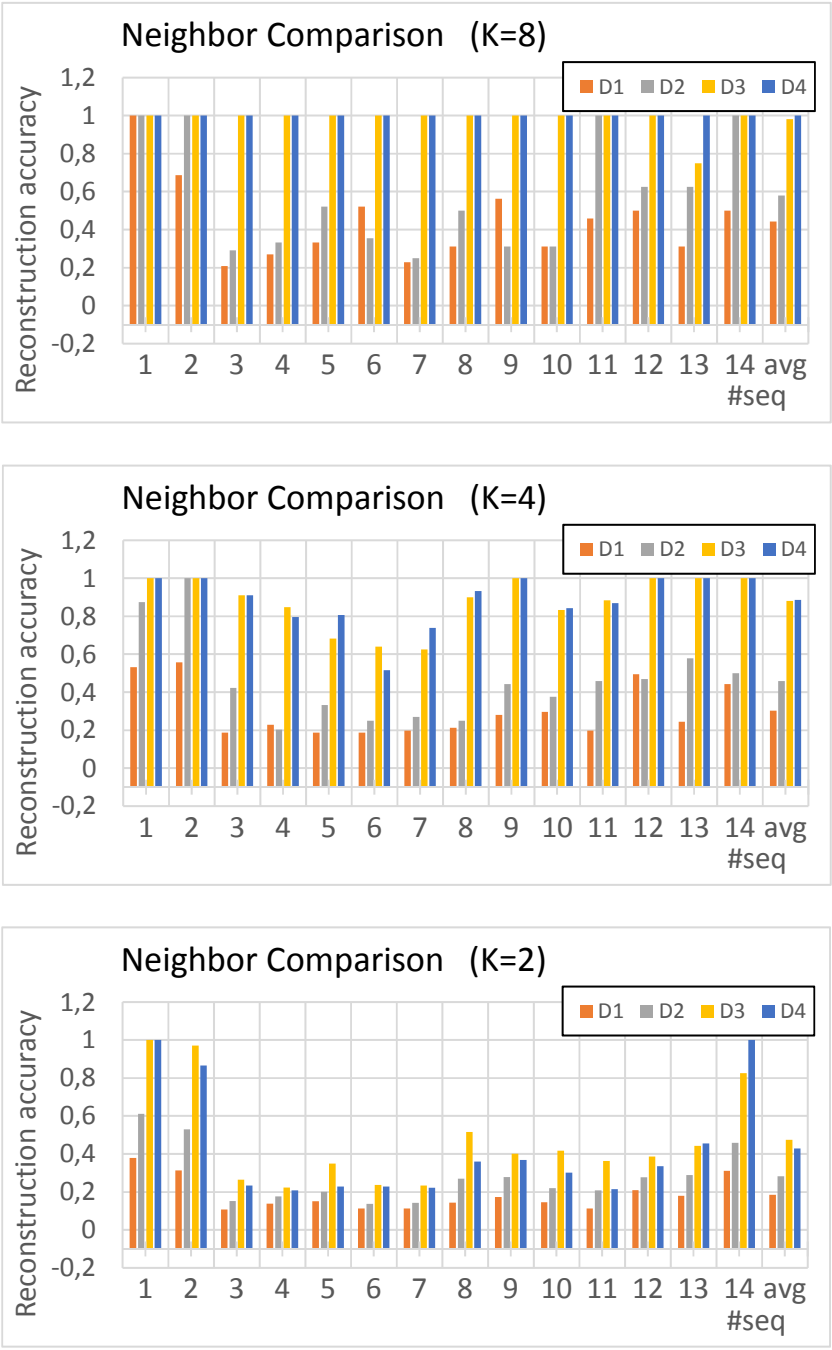


FIGURE 6.5: Performance evaluation of the proposed system in terms of NC metric for each video sequence and block size K.

and exploiting the foreground metrics D_3 or D_4. The other metrics reach good results only on sequences 1 and 2, which contain a lot of handoffs between tiles. Using smaller block sizes, the system performance decreases. However, if the training sequence is complete enough (sequences 1 and 2), the proposed system is still able to find the correct tile placement. Average performances are reported in the last bin of each graph. The metric D_4 has provided the best average result in all the cases.

Since the sensor calibration and layout discovery need to be computed only once, no time constraints are usually imposed in real application. On average, the solver took less than one second to provide a solution with $K = 8$, while about 100 seconds to place the 192 tiles obtained with $K = 2$. The code has been written using Matlab and without including specific code optimization.

Chapter 7

Conclusions

In this doctoral thesis, we have described an innovative sensing floor architecture which can be integrated in Intelligent Environment projects with the aim at creating new forms of natural interactions between humans and physical or virtual worlds.
The solution is based on the "Florim Age" technology, a new sensing floor device, which acquires matrices of pressure points and converts them to gray level images. The developed prototype will be soon available as a commercial product. It allows to create low-cost setups with standard ceramic tiles and can be easily integrated with data coming from other sensors, such as 3D cameras, temperature sensors or pressure sensors and so on.
The low cost of production and the high scalability make the proposed solution very promising also from the commercial point of view, allowing wide installations in both private and public spaces. The complete system can be easily moved thanks to the innovative

characteristics of the floating floors. Temporary exhibitions, shopping centers, and stores may exploit this technology to monitor people flows and crowds instead of common vision-based surveillance systems. In particular, thanks to the automatic system proposed in chapter 6 to configure wide area sensing floors, the spatial placement of the sensor devices can be automatically estimated from the data, i.e., using a training sequence instead of needing a manual configuration.

The floor is still able to detect people and filter out the noise, as proved in chapter 4. The floor system is not affected by occlusions, as vision systems are. If many people are walking on the floor, each person will generate a peak on the sensor data. A corresponding detection and track will be generated by the low-level processing system. Of course, there are limits defined by the spatial resolution of the sensor. If two people are closer than this threshold, they will be merged in a single detection. Nevertheless, the same problem affects traditional vision system based on motion, when groups of people are walking close together. Spatial resolution is currently set around 125mm. This value has been selected not only to reduce the hardware equipment, but also because higher frequencies are denied by the ceramic layer. This resolution limits the application of the proposed floor in some fields, such as medical rehabilitation and postural analysis. However, it also reduces the amount of data captured and processed. For example, a sensing floor covering a

room of one hundred square meters generates a floor image of 6400 pixels only, while a very large environment of ten thousand square meters produces an image smaller than 1 Megapixel. Real time processing and storing of floor images is thus reasonable even on low cost processing units. Data transmission does not have tight bandwidth requirements, even in very large installations.
As proved in chapter 4.1.4, the floor can be used to recognize people actions and behaviors. Machine learning algorithms have been applied to this aim. In particular, SVM and Random Forest classifiers have been exploited to recognize some people actions such as jumps and falls.
Finally, the system can be used as input device in a plethora of applications and in this work we presented some use cases. Surveillance applications in which intruders and people falls detections are the pursued objectives (e.g., in privacy-protected places such as toilets and bedrooms); Human-Computer Interaction applications in which the floor has been used as gamepad controller for interactive applications, virtual reality and edutainment games. At last, the Zone Statistics application in which long term data collection and statistical analysis methods are used to characterize people movements and common flows, for example in commercial or cultural heritage expositions.

List of Publications

Journals:

- Roberto Vezzani, Martino Lombardi, Augusto Pieracci, Paolo Santinelli, and Rita Cucchiara. "A General-Purpose Sensing Floor Architecture for Human-Environment Interaction". In: *ACM Trans. Interact. Intell. Syst.* 5.2 (June 2015), 10:1–10:26. ISSN: 2160-6455

Conferences:

- Martino Lombardi, Roberto Vezzani, and Rita Cucchiara. "Detection of Human Movements with Pressure Floor Sensors". In: *Proceedings of the 18th International Conference on Image Analysis and Processing*. Vol. 9280. Genoa, Italy, Sept. 2015, pp. 620–630

- R. Vezzani, M. Lombardi, and R. Cucchiara. "Automatic configuration and calibration of modular sensing floors". In: *Advanced Video and Signal Based Surveillance (AVSS), 2015 12th IEEE International Conference on*. 2015, pp. 1–6

- R. Vezzani, M. Lombardi, A. Pieracci, P. Santinelli, and R. Cucchiara. “Sensing floors for privacy-compliant surveillance of wide areas”. In: *Proceedings of the 10th IEEE International Conference on Advanced Video and Signal-Based Surveillance*. Krakow, Poland, Aug. 2013

Workshops:

- M. Lombardi, A. Pieracci, P. Santinelli, R. Vezzani, and R. Cucchiara. “Human Behavior Understanding with Wide Area Sensing Floors”. In: *Proceedings of the 4th International Workshop on Human Behavior Understading (HBU2013)*. Barcelona, Spain, Oct. 2013

Patents:

- International patent WO 2014141166-A1. Title: "Substrate for a sensitive floor and a method for displaying loads on the substrate". Publication date: 18/01/2014. Depository: Claudio Lucchese. Inventors: Rita Cucchiara, Martino Lombardi, Augusto Pieracci, Paolo Santinelli, Roberto Vezzani.

Technical Reports:

- "Florimage Project" within the Regional Operational Program POR FESR, Softech-ict, University of Modena and Reggio Emilia, Florim Ceramiche spa:

1. V. Cannillo, R. Cucchiara, M. Lombardi, A. Pieracci, P. Santinelli, R. Vezzani. Deliverable D2: "Sensors";
2. V. Cannillo, R. Cucchiara, M. Lombardi, A. Pieracci, P. Santinelli, R. Vezzani. Deliverable D3: "Energy and Communications";
3. V. Cannillo, R. Cucchiara, M. Lombardi, A. Pieracci, P. Santinelli, R. Vezzani. Deliverable D4:"Embedded Processing";
4. V. Cannillo, R. Cucchiara, M. Lombardi, A. Pieracci, P. Santinelli, R. Vezzani. Deliverable D5: "Applications";

- "Dicet Project" within PON R&C project DICET-INMOTO (PONO4a2-D), Softech-ict, University of Modena and Reggio Emilia, University of Salento.

1. R. Cucchiara, M. Lombardi, R. Vezzani, G. Serra. Deliverable D2: "Sensing Floors and Applications".

Bibliography

[1] M.D. Addlesee et al. “The ORL active floor [sensor system]”. In: *Personal Communications, IEEE* 4.5 (Oct. 1997), pp. 35–41. ISSN: 1070-9916.

[2] J. Anlauff, T. Großhauser, and T. Hermann. “TacTiles: a low-cost modular tactile sensing system for floor interactions”. English. In: *Proc. of the 6th Nordic Conf. on Human-Computer Interaction: Extending Boundaries*. ACM, 2010, pp. 591–594. ISBN: 978-1-60558-934-3.

[3] Luigi Atzori, Antonio Iera, and Giacomo Morabito. “The Internet of Things: A survey”. In: *Computer Networks* 54.15 (2010), pp. 2787 –2805. ISSN: 1389-1286.

[4] Juan Augusto et al. “"Intelligent Environments: a manifesto"”. In: *Human-centric Computing and Information Sciences* 3.1 (2013), p. 12. ISSN: 2192-1962.

[5] L. Bixio et al. “Distributed Cognitive Sensor Network Approach for Surveillance Applications”. In: *Advanced Video and*

Signal Based Surveillance, 2009. AVSS '09. Sixth IEEE International Conference on. Sept. 2009, pp. 232–237.

[6] Leo Breiman. "Random Forests". English. In: *Machine Learning* 45.1 (2001), pp. 5–32. ISSN: 0885-6125.

[7] M.D. Breitenstein et al. "Online Multiperson Tracking-by-Detection from a Single, Uncalibrated Camera". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 33.9 (2011), pp. 1820–1833. ISSN: 0162-8828.

[8] M.D. Breitenstein et al. "Robust tracking-by-detection using a detector confidence particle filter". In: *Computer Vision, 2009 IEEE 12th International Conference on*. 2009, pp. 1515–1522.

[9] Taeg Sang Cho, S. Avidan, and W.T. Freeman. "A probabilistic image jigsaw puzzle solver". In: *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*. June 2010, pp. 183–190.

[10] Young Seek Cho et al. "Development of smart LED lighting system using multi-sensor module and bluetooth low energy technology". In: *Sensing, Communication, and Networking (SECON), 2014 Eleventh Annual IEEE International Conference on*. June 2014, pp. 191–193.

[11] D.M. Chu and A.W.M. Smeulders. "Thirteen Hard Cases in Visual Tracking". In: *Advanced Video and Signal Based Surveillance (AVSS), 2010 Seventh IEEE International Conference on*. 2010, pp. 103–110.

[12] Dorin Comaniciu and Peter Meer. "Mean Shift: A Robust Approach Toward Feature Space Analysis". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 24.5 (May 2002), pp. 603–619. ISSN: 0162-8828.

[13] J Delsing and P Lindgren. "Sensor communication technology towards ambient intelligence". In: *Measurment Sci Technol* 16 (2005), pp. 37–46.

[14] Google Developers. *Google Maps Javascript API V3*. 2014. URL: https://developers.google.com/maps/documentation/javascript.

[15] Théodore Du Moncel. *Le téléphone, le microphone et le phonographe*. Hachette, 1882.

[16] EIA. *Packaging Material Standards for ESD Sensitive Items*. 1988.

[17] Dave Evans. "The internet of things: How the next evolution of the internet is changing everything". In: *CISCO white paper* 1 (2011), pp. 1–11.

[18] A. Gatto et al. “Indoor people localization and tracking using an energy harvesting smart floor”. In: *Mechatronic and Embedded Systems and Applications (MESA), 2014 IEEE/ASME 10th International Conference on*. Sept. 2014, pp. 1–5.

[19] Will Goldstone. *Unity game development essentials*. Packt Publishing Ltd, 2009.

[20] L. Gorelick et al. “Actions as Space-Time Shapes”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 29.12 (Dec. 2007), pp. 2247–2253. ISSN: 0162-8828.

[21] Jayavardhana Gubbi et al. “Internet of Things (IoT): A vision, architectural elements, and future directions”. In: *Future Generation Computer Systems* 29.7 (2013), pp. 1645 –1660.

[22] Vlado Handziski et al. “TWIST: A Scalable and Reconfigurable Testbed for Wireless Indoor Experiments with Sensor Networks”. In: *Proceedings of the 2Nd International Workshop on Multi-hop Ad Hoc Networks: From Theory to Reality*. REALMAN ’06. ACM, 2006, pp. 63–70. ISBN: 1-59593-360-3.

[23] H. Hassanpour, M. Sedighi, and A. Manashty. “Video Frameï£¡s Background Modeling: Reviewing the Techniques”. In: *Journal of Signal and Information Processing* 2 (2 2011), pp. 72–78.

[24] M. Kalantari et al. "A New Approach for Modeling Piezoresistive Force Sensors Based on Semiconductive Polymer Composites". In: *Mechatronics, IEEE/ASME Transactions on* 17.3 (2012), pp. 572–581. ISSN: 1083-4435.

[25] I. Laptev. "On Space-Time Interest Points". In: *Int. J. Comput. Vision* 64.2-3 (Sept. 2005), pp. 107–123. ISSN: 0920-5691.

[26] M. Lombardi et al. "Human Behavior Understanding with Wide Area Sensing Floors". In: *Proceedings of the 4th International Workshop on Human Behavior Understading (HBU2013)*. Barcelona, Spain, Oct. 2013.

[27] Martino Lombardi et al. "Sensing floors for privacy-compliant surveillance of wide areas". In: *Proceedings of IEEE Conference on Advanced Video and Signal-Based Surveillance*. Kraków, Poland, Aug. 2013.

[28] Martino Lombardi, Roberto Vezzani, and Rita Cucchiara. "Detection of Human Movements with Pressure Floor Sensors". In: *Proceedings of the 18th International Conference on Image Analysis and Processing*. Vol. 9280. Genoa, Italy, Sept. 2015, pp. 620–630.

[29] L. Middleton et al. "A floor sensor system for gait recognition". In: *Fourth IEEE Workshop on Automatic Identification Advanced Technologies*. 2005, pp. 171–176.

[30] Dorothy Monekosso, Paolo Remagnino, and Yoshinori Kuno. *Intelligent Environments: Methods, Algorithms and Applications*. Springer, 2008.

[31] J.M.A. Morgado and A. Konig. "Low-power concept and prototype of distributed resistive pressure sensor array for smart floor and surfaces in intelligent environments". In: *Systems, Signals and Devices (SSD), 2012 9th International Multi-Conference on*. Mar. 2012, pp. 1–6.

[32] Joseph Paradiso et al. "The magic carpet: physical sensing for immersive environments". In: *CHI '97 Extended Abstracts on Human Factors in Computing Systems*. CHI EA '97. Atlanta, Georgia: ACM, 1997, pp. 277–278. ISBN: 0-89791-926-2.

[33] M. Piccardi. "Background subtraction techniques: a review". In: *IEEE International Conference on Systems, Man and Cybernetics*. Vol. 4. 2004, 3099–3104 vol.4.

[34] D. Pomeranz, M. Shemesh, and O. Ben-Shahar. "A fully automated greedy square jigsaw puzzle solver". In: *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*. June 2011, pp. 9–16.

[35] Ronald Poppe. "A survey on vision-based human action recognition". In: *Image and Vision Computing* 28.6 (2010), pp. 976 –990. ISSN: 0262-8856.

[36] Andrea Prati and Rita Cucchiara. "Video Analysis for Ambient intelligence in Urban Environments". In: *Intelligent Environments: Methods, Algorithms and Applications*. Ed. by Y. Kuno ed. P. Remagnino D. Monekosso. Apr. 2010.

[37] Weimin Qiu et al. "Design of intelligent greenhouse environment monitoring system based on ZigBee and embedded technology". In: *Consumer Electronics - China, 2014 IEEE International Conference on*. Apr. 2014, pp. 1–3.

[38] R. Rajalingham, Y. Visell, and J.R. Cooperstock. "Probabilistic Tracking of Pedestrian Movements via In-Floor Force Sensing". In: *Canadian Conference on Computer and Robot Vision*. 2010, pp. 143–150.

[39] Krishnan Rangarajan and Mubarak Shah. "Establishing motion correspondence". In: *CVGIP: Image Understanding* 54.1 (1991), pp. 56 –73. ISSN: 1049-9660.

[40] Guillermo Rauch. *Socket.IO: the cross-browser WebSocket for realtime apps.* 2012. URL: http://socket.io/.

[41] Bruce Richardson et al. "Z-Tiles: Building Blocks for Modular, Pressure-sensing Floorspaces". In: *CHI '04 Extended Abstracts on Human Factors in Computing Systems*. CHI EA '04. Vienna, Austria: ACM, 2004, pp. 1529–1532. ISBN: 1-58113-703-6.

[42] H. Rimminen et al. “Detection of falls among the elderly by a floor sensor using the electric near field”. In: *Information Technology in Biomedicine, IEEE Transactions on* 14.6 (2010), pp. 1475–1476. ISSN: 1089-7771.

[43] D. Savio and T. Ludwig. “Smart Carpet: A Footstep Tracking Interface”. In: *Int'l Conf. on Advanced Information Networking and Applications Workshops*. Vol. 2. May 2007, pp. 754–760.

[44] G Schiele, M Handte, and C Becker. “Pervasive Computing Middleware”. In: *Handbook of Ambient Intelligence and Smart Environments* (2009).

[45] R. Schneiderman. “Internet of Things/M2M??-??A (Standards) Work in Progress”. In: *Modern Standardization:Case Studies at the Crossroads of Technology, Economics, and Politics*. Wiley-IEEE Standards Association, 2015, p. 288. ISBN: 9781119043492.

[46] Y.L. Shen and C. S. Shin. “Distributed Sensing Floor for an Intelligent Environment”. In: *IEEE Sensors Journal* 9.12 (2009), pp. 1673–1678. ISSN: 1530-437X.

[47] Yu-Lin Shen and Chow-Shing Shin. “Distributed Sensing Floor for an Intelligent Environment”. In: *Sensors Journal, IEEE* 9.12 (Dec. 2009), pp. 1673–1678. ISSN: 1530-437X.

[48] M. Shimojo, M. Ishikawa, and K. Kanaya. "A flexible high resolution tactile imager with video signal output". In: *Proc. of IEEE International Conference on Robotics and Automation, 1991*. Vol. 1. Apr. 1991, pp. 384–389.

[49] A. Smeulder et al. "Visual Tracking: an Experimental Survey". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 36.7 (July 2014), pp. 1442–1468.

[50] P. Srinivasan et al. "A pressure sensing floor for interactive media applications". In: *Proc. of the Int'l Conf. on Advances in computer entertainment technology*. Valencia, Spain, 2005, pp. 278–281. ISBN: 1-59593-110-4.

[51] L. M. Surhone, M. T. Tennoe, and S. F. Henssonow. *Node.Js*. Mauritius, 2010. URL: http://nodejs.org.

[52] H.Z. Tan, L.A. Slivovsky, and A. Pentland. "A sensing chair using pressure distribution sensors". In: *Mechatronics, IEEE/ASME Transactions on* 6.3 (Sept. 2001), pp. 261–268. ISSN: 1083-4435.

[53] D.I. Tapia et al. "SYLPH: An Ambient Intelligence based platform for integrating heterogeneous Wireless Sensor Networks". In: *Fuzzy Systems (FUZZ), 2010 IEEE International Conference on*. July 2010, pp. 1–8.

[54] M. Valtonen, J. Maentausta, and J. Vanhala. "TileTrack: Capacitive human tracking using floor tiles". In: *IEEE International Conference on Pervasive Computing and Communications*. Mar. 2009, pp. 1–10.

[55] Thijs HJ Van Osch et al. "Inkjet printing of narrow conductive tracks on untreated polymeric substrates". In: *Advanced Materials* 20.2 (2008), pp. 343–345.

[56] Jean-Philippe Vasseur and Adam Dunkels. *Interconnecting smart objects with ip: The next internet*. Morgan Kaufmann, 2010.

[57] R. Vera-Rodriguez et al. "Comparative Analysis and Fusion of Spatiotemporal Information for Footstep Recognition". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 35.4 (2013), pp. 823–834. ISSN: 0162-8828.

[58] R. Vezzani, M. Lombardi, and R. Cucchiara. "Automatic configuration and calibration of modular sensing floors". In: *Advanced Video and Signal Based Surveillance (AVSS), 2015 12th IEEE International Conference on*. 2015, pp. 1–6.

[59] R. Vezzani et al. "Sensing floors for privacy-compliant surveillance of wide areas". In: *Proceedings of the 10th IEEE International Conference on Advanced Video and Signal-Based Surveillance*. Krakow, Poland, Aug. 2013.

[60] Roberto Vezzani et al. "A General-Purpose Sensing Floor Architecture for Human-Environment Interaction". In: *ACM Trans. Interact. Intell. Syst.* 5.2 (June 2015), 10:1–10:26. ISSN: 2160-6455.

[61] Mark Weiser. "Hot topics-ubiquitous computing". In: *Computer* 26.10 (Oct. 1993), pp. 71–72. ISSN: 0018-9162.

[62] K. Weiss and Heinz Worn. "The working principle of resistive tactile sensor cells". In: *Proc. of IEEE International Conference on Mechatronics and Automation*. Vol. 1. 2005, 471–476 Vol. 1.

[63] Mike Willis. "Current Inkjet Technology and Future Directions". In: *Inkjet Technology for Digital Fabrication* (), pp. 343–362.

[64] Bob G. Witmer and Michael J. Singer. "Measuring Presence in Virtual Environments: A Presence Questionnaire". In: *Presence: Teleoper. Virtual Environ.* 7.3 (June 1998), pp. 225–240. ISSN: 1054-7460.

[65] C.R. Yu et al. "Human Localization via Multi-Cameras and Floor Sensors in Smart Home". In: *IEEE International Conference on Systems, Man and Cybernetics, 2006.* Vol. 5. 2006, pp. 3822–3827.

[66] X.-W. Zhang et al. "Time dependence of piezoresistance for the conductor-filled polymer composites". In: *Journal of Polymer Science B Polymer Physics* 38 (Nov. 2000), pp. 2739–2749.