Alma Mater Studiorum · Università di Bologna
Dipartimento di Informatica - Scienza e Ingegneria (DISI)

Dottorato di Ricerca in
Computer Science and Engineering

Ciclo XXXII

Settore Concorsuale: 09/H1
Settore Scientifico Disciplinare: ING-INF/05

Human Action Recognition and Monitoring in Ambient Assisted Living Environments

Presentata da
ANTONIO MAGNANI

Coordinatore Dottorato
Prof. DAVIDE SANGIORGI

Supervisore
Prof.ssa ANNALISA FRANCO

Tutor
Prof. DARIO MAIO

Esame Finale Anno 2020

A Paola, Claudio e Giulio Regeni,
che la verità possa finalmente "esondare dalle coscienze".

Acknowledgements

I want to thank Prof. Annalisa Franco for her supervision during my PhD: her assistance has provided an invaluable contribution to both my academic and personal growth. A special thanks to Prof. Dario Maio for having directed me towards this path and for his support during these four years. A particular mention to Prof. Michael Madden and Dr Ihsan Ullah for hosting me in Galway during one of the most exciting experiences of this journey. A heartfelt thanks to Luca and my PhD colleagues and friends, especially Michele and Roberto. An infinite thanks to my whole family, and to whoever had the strength and the patience to stand by me on this path. Last but not least, my most significant recognition goes to Caterina, who was able *to turn on the light, even in the darkest moments.*

Abstract

Population ageing is set to become one of the most significant challenges of the 21st century, with implications for almost all sectors of society. Especially in developed countries, governments should immediately implement policies and solutions to facilitate the needs of an increasingly older population. Ambient Intelligence (AmI) and in particular the area of Ambient Assisted Living (AAL) offer a feasible response, allowing the creation of human-centric smart environments that are sensitive, adaptive and responsive to the needs, habits and behaviours of the user. These intelligent environments aim to enhance the quality of life of the elderly in a domestic context, increasing their autonomy and reducing their dependence on the healthcare system. In such a scenario, understand what a human being is doing, if and how he/she is interacting with specific objects, or whether abnormal situations are occurring is critical. This thesis is hence focused on two related research areas of AAL: the development of innovative vision-based techniques for human action recognition and the remote monitoring of users behaviour in smart environments. The former topic is addressed through different approaches based on data extracted from RGB-D sensors. A first algorithm exploiting skeleton joints orientations extracted from Microsoft Kinect is proposed. This approach is then extended through a multi-modal strategy that includes the RGB channel to define a number of temporal images, capable of describing the time evolution of a specific action. Finally, the concept of template co-updating concerning action recognition is introduced. In fact, it is known that the model created with different techniques in dynamic contexts has relatively

limited validity, both because they are created typically starting from a minimal set of examples and also because time inevitably brings changes that can not be covered by the initial model. Such limitations can be partially overcome by the introduction of template updating techniques that should hopefully take place in a completely unsupervised way. Indeed, exploiting different data categories (e.g., skeleton and RGB information) improve the effectiveness of template updating through co-updating techniques. The action recognition algorithms have been evaluated on CAD-60 and CAD-120, achieving results comparable with the state-of-the-art. Moreover, due to the lack of datasets including skeleton joints orientations, a new benchmark named Office Activity Dataset has been internally acquired and released.

Regarding the second topic addressed, the goal is to provide a detailed implementation strategy concerning a generic Internet of Things monitoring platform that could be used for checking users' behaviour in AmI/AAL contexts.

Contents

List of Figures

List of Tables

Chapter 1

About this thesis

The continuous advances in sensing technologies and networking infrastructures enable the development of intelligent software which can provide real-time analysis of specific situations of interest in a home environment, intending to enhance the quality of life of the occupants. In the specific field of health-care, particular attention is generally devoted to systems able to detect and recognise a different kind of situations and, eventually, to provide prompt alarms. Hence, this work contributes to the extremely broad paradigm of Ambient Intelligence, focusing on a specific and increasingly relevant application scenario that is Ambient Assisted Living. The main focus of this work is the development of a monitoring system with specific characteristics:

- *Unobtrusive*: the proposed solutions and algorithms should be transparent to the user and collect information as he/she is performing the usual daily activities. Vision-based techniques are preferable from this point of view with respect to sensor-based approaches, whether they are wearable or environmental. The former requires the user to wear (and to manage) some hardware device, and the latter does not offer general solutions for the recognition of a comprehensive set of scenarios.
- *Non-cooperative*: the user cooperation is not always possible (particularly in the health-care domain), and the needed information should

possibly be collected in an unsupervised way. Even in this case, the adoption of vision-based solutions is undoubtedly preferable.

- *Real-time*: efficiency is crucial to process continuously the stream of data acquired live and to provide timely warnings.

- *Adaptive*: the human behaviour continuously evolves, and the system must be able to automatically update its internal model to keep consistent performance in the time.

Some of the above requirements and considerations led us to the adoption of camera sensor for data acquisition. More specifically, we have evaluated different approaches that use the Kinect sensor, a low-cost solution suitable for home environments.

In Figure 1.1 we give an overview of an Ambient Assisted Living monitoring system. In our vision, the system continuously acquires and analyses the video stream acquired by Kinect (or an array of camera). The *activity detection* module analyses the data to notice the presence of humans in the room and possible ongoing activities; then *activity recognition* is performed by an ad-hoc module able to identify actions of interest on the basis of a set of activity templates. In case of specific action classified as dangerous, proper warnings will be raised. The successful recognition of actions enables the possibility of unsupervised updating the existing templates to make them more robust and effective. Moreover, even being aware of the great success of neural networks coupled with deep learning techniques in many applications, we choose to design action recognition approaches based on hand-crafted features. In the specific context of this thesis, in fact, the acquisition of a large amount of training data typically needed for network training is quite difficult and unlikely. The home environment is usually characterised by a very limited number of users, and also most of the reference benchmarks for indoor action recognition reproduce a "small-size" scenario, with few users and few activity samples per user. We are confident that in this scenario also "traditional" computer vision techniques can achieve good results and real

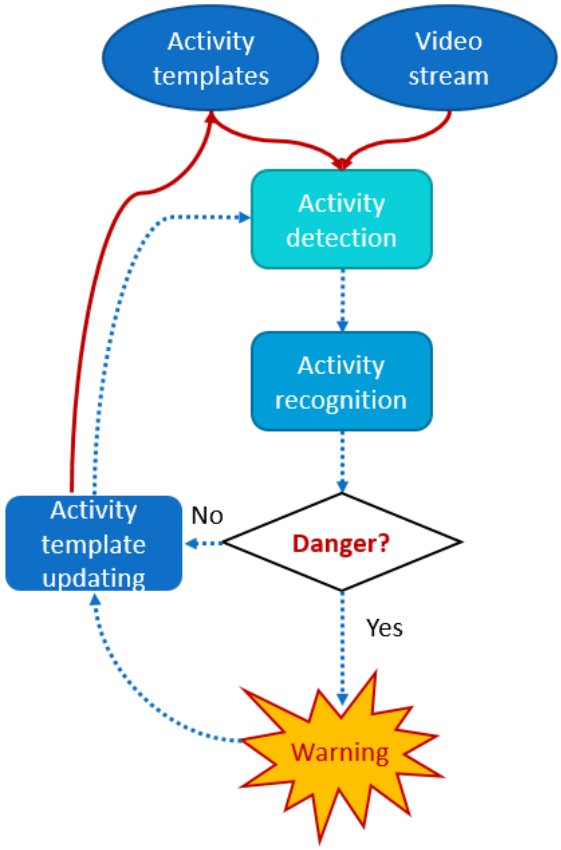


Figure 1.1: Flowchart of an Ambient Assisted Living monitoring system.

time processing capabilities even with limited computational power.

The information collected by this system, as well as any alarms, must be available remotely by potential stake-holders (e.g., caregivers, health-care professionals or, eventually, relatives). Starting from this premise, and from the close link between the Internet of Things and Ambient Intelligence, we propose a generic framework designed for the monitoring of heterogeneous sensor networks. Indeed, one of the objectives of this work is to provide a detailed full-stack implementation strategy concerning a platform for monitoring sensors of various kinds. The fundamental idea is to exploit this platform to remotely check user behaviour in sensitive contexts and offer a

prompt reaction from potential accidents (e.g., falls). By taking advantage of the proposed techniques for human activities recognition and template updating, our platform could provide useful information about the actions performed by patients or particular categories of users such as the elderly in several Ambient Assisted Living contexts.

1.1 Structure of this thesis

This thesis is organised in four parts.

Part I introduces the work, defining the context in which it is placed. It presents the salient aspects of the broad fields of research on Human Action/Activity Recognition and Internet of Things, setting out some of the main concepts adopted in the respective parts of the thesis.

Part II presents the first contributions of the thesis, focusing on the modules of activity recognition and template updating depicted in Figure 1.1. These approaches are based on data extracted from RGB-D sensors. In particular, an innovative handcrafted-feature action recognition approach based on joint orientations is introduced. Secondly, a multi-modal strategy for human action recognition (based on skeletal and RGB data) is illustrated. Finally, a multi-modal template co-updating approach is presented. Besides, the state of the art of RGB-D based action recognition is discussed.

Part III provides a detailed implementation strategy concerning an Internet of Things monitoring solution. This generic framework was initially designed for urban contexts (a real case study is presented), but its interoperability also makes it suitable for monitoring Ambient Assisted Living contexts and, more generally, smart homes. Also, this part describes some of the most commonly adopted Internet of Things platforms in order to provide the reader with a comparison with the presented open-source framework.

Part IV draws conclusions and paves the way for future contributions.

1.2 List of publications

The publications included in this thesis are the following:

- Franco A., Magnani A., Maio D., *"Joint Orientations from Skeleton Data for Human Activity Recognition"* in proceedings 19th International Conference on Image Analysis and Processing (ICIAP17), Catania, September 2017;

- Calderoni L., Magnani A., Maio D., *"IoT Manager: a Case Study of the Design and Implementation of an Open Source IoT Platform"* in proceedings IEEE 5th World Forum on Internet of Things 2019 (WF-IoT2019), Limerick, Ireland, April 2019;

- Calderoni L., Magnani A., Maio D., *"IoT Manager: an Open Source IoT framework for Smart Cities"* in Journal of Systems Architecture, 2019, vol. 98, pp. 413-423;

- Franco A., Magnani A., Maio D., *"A multimodal approach for human activity recognition based on skeleton and RGB data"* in Pattern Recognition Letters, 2020, vol. 131, pp. 293-299;

- Franco A., Magnani A., Maio D., *"Template co-updating in multimodal human activity recognition systems"* to appear on proceedings 35th ACM/SIGAPP Symposium On Applied Computing (SAC), Brno, Czech Republic, March 2020.

Part I

Background and Motivations

Chapter 2

Context

> *"The most profound technologies are those that disappear. They weave themselves into the fabric of everyday life until they are indistinguishable from it."*
>
> – Mark Weiser, 1999

2.1 An Ageing Society

By 2050, there will be about 10 billion people on the planet (United Nations, b). Human history tells us that it took thousands of years (from the appearance of man until 1800) before the world population reached the first billion, but it took only a couple of centuries to reach today's 7.7 billion. The second billion was reached in 130 years (1930), the third billion in 30 years (1960), the fourth billion in 15 years (1974) and the fifth billion in only 13 years (1987). In the 20th century alone, the world population rose from 1.65 billion to 6 billion, in 1970 there were about half of the people who are there today. After this peak, the growth rate has progressively slowed down, but nevertheless, the world population is growing, although not everywhere in the same way. The growth of the world population in the last two centuries is, in fact, due to advances in medicine and improved living standards, which have significantly reduced infant and maternal mortality and increased life

expectancy. Demographic change is polarising in two directions: part of the world has a negative growth rate and a robust ageing population (in Italy and Japan the average age is 48 years); on the other hand, in emerging countries, the growth rate is still very high, and the average age is quite low (in all African countries is between 16 and 20 years). One of the main consequences of this unprecedent growing trend, according to World Population Prospects data (United Nations, b), is that the number of older adults - aged 60 and over - is expected to more than double by 2050 and more than triple by 2100, rising from 962 million in 2017 to 2.1 billion in 2050 and 3.1 billion in 2100. Worldwide, life expectancy at birth increased by 3.6 years between 2000-2005 and 2010-2015 (from 67.2 to 70.8 years) and is projected to increase to around 77 years by 2050 and to reach 83 years by 2100 (United Nations, a). Finally, the population aged 60 and over is growing faster than the other age groups.

As can be expected, population ageing is set to become one of the most significant social transformations of the 21st century, with implications for almost all sectors of society, from the labour market and the economic system (financial services, demand for goods and services, such as housing, transport and social protection) to the foundations of society, such as family structures and intergenerational links. The need to anticipate this demographic shift is more evident than ever and, especially in developed countries, governments should immediately implement policies and solutions to facilitate the needs of an increasingly older population.

2.1.1 Challenges

(Rashidi and Mihailidis, 2013) highlighted several challenges for society and, in particular, for the health-care system:

- *Increase in diseases*: one of the main consequences of the increase in the number of older adults is the apparent increase in age-related diseases. Among these, surely the two most common neurological disor-

ders (Alzheimer's disease and Parkinson's disease) for which currently there is no cure.

- *Increase in health-care costs*: there will be a tangible rising in health-care costs. For example, Italian senior citizens – that are 22% of the total population – use more than 57% of the health-care budget. In the coming decades, the ageing of the population will put a strain on current health care models.

- *Shortage of caregivers*: The increase in the number of seniors will not result in a linear increment in the number of trained professionals to work with the ageing population. Necessarily, family members will have to replace caregivers with a series of different complications (e.g., high levels of emotional distress, physical health problems).

- *Dependency*: with the increase in age-related diseases, the number of individuals unable to live autonomously will also rise. In Italy, 2.5 million older adults have functional limitations of some kind (mobility, autonomy, communication, etc.) and are partially or totally not self-sufficient. In such a context, the question arises of how it will be possible to offer quality services to dependent older adults.

- *Larger impact on society*: as a society, we will not be able to provide adequate assisted living or skilled nursing facilities in relation to the number of older adults. Moreover, the need to assist a family member has a direct impact on the labour market, affecting phenomena such as unemployment, absenteeism and downgrading.

Given these challenges, it is apparent that possible solutions must be sought through a change of social and technological paradigm, relying on new and promising lines of research.

2.2 Ambient Intelligence (AmI)

The term Ambient Intelligence (AmI) defines a relatively new paradigm of the information and communication technologies aimed at strengthening the person capabilities with the realization of *"digital environments that are adaptive, sensitive and responsive to the needs, habits, gestures and emotions of the users"* (Acampora et al., 2013). However sophisticated and futuristic this could seem, the outstanding developments of the research regarding Sensor Networks, Multi-Agent Systems, Pervasive Computing and Artificial Intelligence, make it possible to design and implement solutions to various issues of everyday life. In addition, the constant proliferation of devices of various types and nature (e.g. wearable devices, smart objects), the affordability of sensors and actuators, complete the panorama making this scenario extremely topical.

AmI's vision, emerging but increasingly relevant, is primarily characterized by the idea of rendering the environment in which the user interacts intelligent. A first, crucial, question could be raised: what is meant by *intelligent*? While this question can be answered in different ways, the definition of AmI given above highlights several key features that could suggest a possible answer. These characteristics can be translated into pointing out a necessary *proactiveness* of the digital environment to the stimuli and events of the observed context. Intelligence is hence closely related to the ability of the system to understand when take sensible action. Therefore, making an analogy, the ambient behaves like a trained human assistant who is able to recognise the user, to know (or eventually learn) the needs of him/her. This assistant will intervene only when necessary and will refuse to act if it does not consider it appropriate (Augusto et al., 2010). Clearly, the intelligence lies in the software that drives the environment.

On the hardware side, the physical infrastructure – consisting of networks, sensors and actuators – that supports an AmI system is referred to as *smart environment* (Cook and Das, 2005). Among other features, this physical layer must be as *transparent* as possible for users. Indeed, it can be said that

AmI marries the *disappearing computer* vision coined by (Weiser, 1999), both from a conceptual and a physical point of view (Augusto, 2007; Cook et al., 2009).

Among the main factors that are allowing AmI to become extremely popular, there is undoubtedly the explosion of the Internet of Things (IoT) paradigm. The phenomenon we are witnessing, that is the growing proliferation of cheap ubiquitous sensors that can potentially be integrated into any context, is becoming the fuel for a multitude of AmI scenarios (Ricciardi et al., 2017). It is enough to think of the many application examples that we are going through and getting used to: a house where heating, lighting, entertainment or security are managed autonomously depending on the presence or absence of individuals, transportation efficiency and assistance to the driver thanks to sensors onboard (e.g., image processing of the driver's face), a smart classroom where students can benefit from customized assisted learning services and all the application scenarios typical of the smart city context (Pellicer et al., 2013).

In this broad panorama, we want to focus our attention on an application domain of primary interest and extremely topical for scientific research. The adoption of the AmI paradigm, and consequently the application of ICT technologies, can help to address some of the aforementioned challenges through the so-called Ambient-Assisted Living (AAL, also knows as Active-Assisted Living) tools.

2.3 Ambient Assisted Living (AAL)

These environments pursue a person-centred conception: the distributed network of sensors and actuators creates a transparent layer able to proactively interact with the user to improve his quality of life. In general, these augmented environments can be used to prevent, cure and improve the well-being of users. Indeed, they may be targeted at users with more or less severe needs, and act in different aspects to help older adults to age at home.

For example, AAL tools can instil awareness in the elderly by monitoring their health conditions (e.g., medication management or medication reminder tools) or can provide enhanced safety through fall detection systems (Zhang et al., 2015), video surveillance systems (Yano et al., 2019) or emergency response systems (Nikoloudakis et al., 2016). Other tools assist in carrying out daily activities, typically through the monitoring of the *Activities of Daily Living* (ADLs) (Debes et al., 2016; Nguyen et al., 2016) and eventually issuing reminders (De Benedictis et al., 2015). In Italy, it is estimated that one in five older adults (over 65 years of age) suffers from depression. This number doubles for the elderly over the age of 80. To tackle this phenomenon and, in particular, to avoid alienation and isolation of the elderly, several solutions of AAL are pursuing a better connection and communication with family and friends (Pinto et al., 2019). Furthermore, considering the category of users primarily concerned by these technologies, users' engagement is one of the highest priority. Of course, this vision and the solutions highlighted can be met only with the involvement of experts from different backgrounds (i.e, technology, health, social sciences) (Florez-Revuelta and Chaaraoui, 2016).

European Union has recognised the importance of the AAL domain and is fostering research into it through the Horizon 2020 programme. During the eight years of the programme, over 4€ billion has been invested in the *"Health, Demographic Change and Well-being"* challenge. Moreover, the EU directly supports AAL projects by co-funding the AAL Joint Programme (AALJP) which has as its primary goal to foster the emergence of innovative ICT-based products, services and systems enabling the *"ageing well at home"* vision.

The AALJP also defines the objectives of the AAL paradigm:

- to increase the autonomy, self-confidence and mobility of people in order to extend the time they can live in their preferred environment;
- to support the maintenance of the health and functional capacities of the elderly;

- to promote a better and healthier lifestyle for individuals at risk;
- to enhance security, prevent social isolation and create networks of support around older people;
- to support caregivers, families and care organisations;
- to get more out of the investments in the ageing society.

The AALJP's website lists more than 200 projects funded in the ten years of the association's existence. Several of these projects have seen the development of commercial solutions that are about to be released into the market. Of course, AAL, as well as all AmI application scenarios, is not just a flourishing field for commercial solutions. From an academic point of view, the survey mentioned above by (Rashidi and Mihailidis, 2013) well summarises the main classes of algorithms and applications area, which are the principal object of scientific research. As for algorithms, the following categories are highlighted: *Human Activity Recognition* (HAR), which aims to recognise human activity/action patterns from various sensor data, *Context Modeling*, that is the capacity to represent the different information deriving from the context (e.g., sensor information, temporal/spatial information, user profiles and preferences), *Anomaly Detection*, whose purpose is the search for patterns that differ from canonical behavior, *Location and Identity Identification*, which deals with monitoring and offering location-based services to the elderly (if necessary as a result of an identification procedure), and *Planning*, which makes it possible to assist the user with respect to daily plans and activities (particularly useful, for example, in patients with dementia). With regard to AAL's application areas, one of the most important is undoubtedly represented by the *Health and Activity Monitoring* tools, which allow to observe health parameters and assist users in carrying out daily activities; another one consists of the *Wandering Prevention* tools, i.e. those solutions that try to prevent and mitigate wandering in patients with dementia; finally, there are the *Cognitive Orthotics* tools that facilitate, for example, the medication management (possibly in an autonomous manner).

In the aforementioned scenario, determining and monitoring what is happening in an environment is critical – in particular, recognising what a human being is doing, if and how he/she is interacting with specific objects, or whether abnormal situations are occurring is the key to the successful realisation of several AmI/AAL applications (Chen and Nugent, 2019).

Chapter 3

Human Activity Recognition: an overview

> *"The recognition of human activities will lead to a number of applications, including personal assistants, virtual reality, smart monitoring and surveillance systems, as well as motion analysis in sports, medicine and choreography."*
>
> – J.K. Aggarwal, 2005

Human Activity Recognition (HAR) is one of the most active and promising research topics in recent years. The purpose of HAR techniques and algorithms is to determine what one or more people are doing in a given context, using data from different sensors or cameras. In a very recent book, (Chen and Nugent, 2019) summarise this complex and articulated process as the composition of the following fundamental steps:

1. choose and deploy specific sensors to actor/s, objects or environments in order to monitor and capture human behaviours, eventually considering state changes of the environment;

2. define computational activity templates in a way that allows software systems to conduct reasoning and manipulation;

3. process perceived information exploiting aggregation and fusion to define a high-level abstraction of context or situation;

4. design and develop algorithms to infer activities from collected sensor data;

5. carry out pattern recognition to ascertain the performed activity.

This chapter introduces the main characteristics and challenges related to the HAR domain and regarding the appropriate choice and deployment of sensors (step 1). The remaining four steps will be the subject of the chapters contained in Part II.

The growing popularity of HAR systems is undoubtedly due to the many areas of real-world applications. In addition to the AAL, and more generally to AmI contexts, among the most common scenarios can be found:

- **Intelligent Visual Surveillance:** Traditional surveillance systems require the presence of a human agent who continuously observes what appears in one or more monitors. It is explicit how important it is to be able to offer intelligent and autonomous public security services. Intuitively, the idea is to exploit the network of cameras distributed in urban contexts or public environments to create surveillance systems capable of tracking and detecting potentially dangerous activities or anomalous situations – such as an example, an individual who is abandoning a bag at the airport or jumping a turnstile in the subway. Action recognition or prediction algorithms can be crucial to significantly increase public safety, triggering appropriate alarms and allowing a sudden intervention of the authorities.

- **Human-Computer/Robot Interaction (HCI/HRI):** The ability to understand gestures and activities is fundamental in the realization of natural interfaces between computers/robots and humans. There are countless examples: from interfacing tools to control the presentation of slides through appropriate sensors, to a humanoid robot that acts as a personal assistant, perhaps in an AAL context.

- **Entertainment:** In recent years, the gaming industry has deployed numerous entertainment devices based on the recognition of human activities. These devices make possible a video-ludic experience that does not necessarily require the presence of a controller. Even the scientific community has greatly benefited from the spread of these cost-effective sensors. Among others, it is certainly worth mentioning the Microsoft Kinect RGB-D sensor, widely adopted and studied in the academic community (Aggarwal and Xia, 2014).

- **Autonomous Driving Vehicles:** In this context, two different scenarios can be distinguished: *i)* the monitoring of human beings in the vicinity of an autonomous driving vehicle, *ii)* the recognition of tasks performed by the driver. In the first case, the challenge is to create algorithms capable of analyzing human body motion with the ultimate goal of predicting a person's intentions in a short period of time (Kong and Fu, 2018). In the latter case, it is crucial to suddenly determine the degree of attention of the driver. This also applies to self-driving vehicles, when it is necessary to transfer control to the human driver (Braunagel et al., 2015).

Of course, what makes it possible to recognise human behaviour in a given scenario strictly depends on the type of sensor adopted. Concerning the application domain object of this work, it is possible to use different technologies that can be mainly distinguished into two macro-categories: *sensor-based* and *vision-based*.

3.1 Sensor-based

As stressed in Chapter 1, we focus on vision-based solutions. Nevertheless, for the benefit of reader and to offer a more comprehensive view, the most relevant aspects and modalities of sensor-based activity recognition are introduced. For a more detailed analysis of sensor-based modalities, interested readers can refer to (Patel and Shah, 2019; Chen and Nugent, 2019).

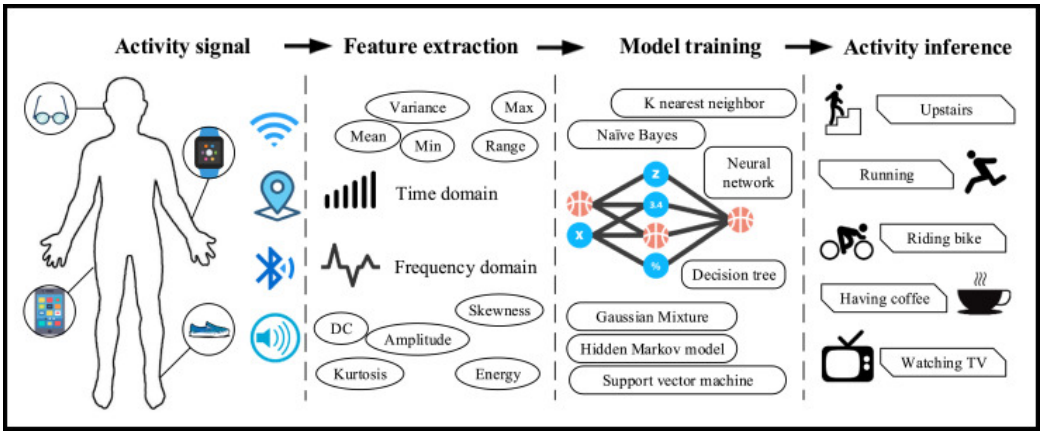


Figure 3.1: Flowchart of sensor-based activity recognition using canonical machine learning approaches (Wang et al., 2019a).

As depicted in Figure 3.1, these approaches take advantage of time series of data collected from different types of sensors. Unlike vision-based solutions, the usual representation of information is a one-dimensional signal. From these raw signal inputs (*activity signal*), on the one hand, features are manually extracted. These are typically based on statistical reference metrics – average, variance, amplitude – and are the input for the training of traditional machine learning models. On the other hand, there are approaches based on deep learning techniques. Deep features are automatically learned from raw sensor data, replacing the feature extraction phase of traditional approaches with the building of the model. The recent work by (Wang et al., 2019a) reviewed sensor-based approaches based on deep learning techniques, distinguishing by sensor-modality and comparing them with traditional approaches. In the next subsections are introduced the main sensor-modalities and strategies concerning this macro-category of approaches.

3.1.1 Body-worn sensors

Nowadays, people are accustomed to the so-called *wearable-devices*: smartwatches, fitness wristbands, smart clothes are accessories worn daily by millions of individuals, thus allowing continuous monitoring of the activities

carried out. Although not definable as wearable-device, smartphones also fall within the category of body-worn sensors, as they allow the acquisition of data from inertial sensors commonly found in all models (i.e., accelerometer, gyroscope). In particular, the ability to measure acceleration and angular velocity is crucial to infer ADLs and sports activities. In the most common scenarios, including the well-known fitness app, the device collects data from the various sensors and classifies them locally. A different strategy, instead, involves the transmission of data to a processing centre – laptop, smartphone, remote server – using appropriate communication technologies (Bluetooth, Wi-Fi, Zigbee, etc.).

Concerning health-care, the recent work by (Wang et al., 2019b) provides an excellent overview of the state of the art of approaches based on body-worn sensors. Particular attention is paid to inertial data and to the different features that have been proposed. Compared to cameras, these devices are available at a lower cost, require fewer data processing and limited computational resources. On the other hand, they require to be worn and run continuously, which is difficult if not impossible in many application scenarios. Among the other cons, there are acceptability and willingness to use these sensors as well as issues such as battery life, the effectiveness and the positioning of inertial sensors. Indeed, the correct positioning of accelerometers in relation to the human body is subject of debate (Cleland et al., 2013). In particular, while inertial sensors typically positioned in the centre of mass of the human body (i.e., lower back and waist) offer excellent results in terms of accuracy for the classification of specific activities (e.g., sitting, walking, lying and falling), on the other hand it is challenging to achieve general approaches that can classify common ADLs without adding other inertial sensors in strategic position, as an example wrists or legs (Attal et al., 2015). Clearly, such a choice would further affect the cons mentioned above, significantly increasing the degree of obtrusiveness.

3.1.2 Object sensors

The purpose of these sensors is to detect the specific use of a particular object. The human activity carried out is inferred by exploiting the movement of the object. For example, a contact sensor applied to a window/drawer/-door can be used to determine its opening, just as an accelerometer applied to a jug can be used to infer the pouring water activity. Among the most common object sensors, there are Radio Frequency Identifiers (RFID) frequently used in smart homes and AAL environments as they provide fine-grained information for the recognition of more complex behaviours (Alsinglawi et al., 2017). The main disadvantage of these sensors is their deployment: difficult to apply to the multitude of human-object interactions and situations that may occur. Nevertheless, the adoption of these sensors can often complement information from other sensing modality.

3.1.3 Ambient sensors

In this case, the focus is posed on subject-environment interaction and resulting contextual information. The environment is augmented with sensors of different nature, whose primary purpose is to observe significant state changes. Such information can be extremely significant; for example, (Litvak et al., 2008) proposed a fall detection system based on floor vibrations and an array of microphones able to discriminate between objects and human fall events. Again using acoustic sensors, it is possible to discriminate some types of ADLs characterised by specific sound patterns – drying hair, working on the computer – or even distinguish between different activities related to the preparation and consumption of various dishes (Sim et al., 2015). Multiple ambient sensors can be set up as Wireless Sensor Network (WSN), (Tunca et al., 2014) proposed a system of HAR in AAL context including photocells, digital distance sensors, sonar distance sensors, contact sensors, temperature sensors, pressure mats and infrared receivers. The main advantages of these sensors, as well as to object sensors, are their unobtrusiveness and being

privacy-preserving. However, as for object sensors, also their distribution can be difficult; moreover, the noise to which these sensors are subjected can profoundly affect the robustness of the approaches, already limited to a few, specific, activities.

3.1.4 Hybrid sensors

The possibility of combining different types of sensors modalities allows the recognition of extremely complex activities, even in contexts where the presence of multiple individuals is expected. (Vepakomma et al., 2015) propose A-Wristocracy, a smart environment explicitly designed for AAL scenarios. The proposed framework combines ambient sensors, object sensors and, as the name may suggest, a wrist-worn wearable device. Although these multi-modal solutions are capable of classifying fine-grained activity with high accuracies, the problems highlighted in the various sensing modalities remain.

With this in mind, the peculiarities of the sensors described above may be integrated with vision-based approaches. The complementarity of the information can allow a significant increase in accuracy compared to the single sensing modes (Ehatisham-ul-Haq et al., 2019).

3.2 Vision-based

Vision-based activity recognition techniques do not require the use of special devices and the only source of information is represented by cameras placed in the environment which continuously acquire video sequences. Considering the different real-world scenarios highlighted at the beginning of this chapter, it is apparent that the adoption of vision-based sensors allows broader applicability from several perspectives. The most prominent example is undoubtedly the surveillance of public places, where it is not possible to apply alternative solutions. In general, regardless of whether they are installed indoors or outdoors, the cameras can provide more comprehensive

environmental information than other sensors. The cost reduction of many camera sensors and the possibility to implement general solutions – independent of the specific activity carried out – have contributed to increasing the interest for these approaches making vision-based HAR one of the most exciting research area in the field of computer vision and machine learning. Over the past few decades, this popularity has led to a real explosion of scientific contributions, making it challenging to explore the *mare magnum* of the state of the art. Besides, this has dramatically increased confusion about the terminology used. Indeed, a first question that arises spontaneously to those who approach vision-based HAR is: what precisely defines the term *activity*? One of the firsts and main topics of debate was the search for a shared nomenclature and a taxonomy that allows defining the granularity of the tasks carried out. Although it may seem a simple question to answer, numerous taxonomies and definitions have been proposed in the literature, particularly concerning discrimination between *action* and *activity*. Specifically, (Aggarwal and Cai, 1999) in one of the first surveys on human-motion analysis distinguished three different areas regarding the interpretation of human motion: motion analysis involving human body parts, tracking of human motion with one or more cameras, recognising human activities. Particularly in the latter area, the terms *action* and *activity* are used interchangeably. Few years before, (Bobick, 1997) tried to clarify this aspect proposing a hierarchy that distinguishes three different levels of abstraction: *movements*, *actions* and *activities*. This hierarchy is refined, among others, by (Poppe, 2010) which proposes the distinction between:

- **action primitive**: considered as an atomic movement involving the use of the limbs (e.g., *raising an arm*). Also known as *motion* in other works;

- **action**: composition of *action primitives*, possibly in a repeated pattern (e.g., *clapping hands*);

- **activity**: a sequence of *actions* to which a specific semantic interpre-

tation can be attributed (e.g., *Jumping hurdles* is an activity composed by the starting, running and jumping actions).

In other reviews, the concept of activity is defined if it involves several people, alternatively is considered as action (Turaga et al., 2008). Among the many possible taxonomies, we will adopt the one provided by (Chaaraoui et al., 2012). In detail, the authors make a distinction based on the degree of semantics (motion, action, activity and behaviour) in relation to the amount of time needed in the analysis. According to this view, an action implies the performance of some simple human primitives (e.g., *sitting*) that require a time frame in the order of seconds. Vice versa, similarly to Poppe's taxonomy, an activity is defined as a sequence of actions in a time frame ranging from minutes to hours (e.g., *cooking*). Finally, behaviours represent the highest semantic level of this taxonomy, including habits and lifestyles sampled in relatively long temporal periods (longer than hours). Although the adoption of the HAR acronym is often linked to Human Activity Recognition, in the context of this work it is more appropriate to refer to the concept of Human Action Recognition.

Compared to the sensor-based approaches described in Section 3.1, the perception of data is different. Of course, the data provided by visual-sensors are not one-dimensional signals but are in the form of 2D or 3D set of data, respectively to represent an image or a video. Analogously to other sensor modalities, both traditional machine learning and deep learning techniques can be adopted. Indeed, the general approach is to extract/learn image features from raw video data and then apply classification algorithms but, regardless of choice to adopt a handcrafted representation-based or learning-based approach, several domain-specific problems are transversal to both strategies. Among the main technical challenges of vision-based HAR, there are:

- *Intra-class and Inter-class variations*: as is intuitable, each person can perform the same action in a very different way than others. Consider

a simple action as running: a person can run by raising heels, can keep elbows distant from the body or may run more or less fast than someone else. Furthermore, the same action could be captured from different points of view. The different ways in which a task is carried out, the assumed poses, and the point of view, are essential factors; they determine a considerable increase in intra-class variability and the consequent difficulty in representing the same category of actions in a comprehensive way. Another significant issue is inter-class variations. As we will see in several chapters of the Part II of this work, the similarity between the different categories of actions is one of the most complex challenges as some motion-patterns are difficult to distinguish (e.g., drinking from a bottle, answering the phone).

- *Environment and recording settings*: many problems with the robustness of HAR algorithms are due to background noise. Indeed, many approaches work well in indoor environments but struggle in outdoor environments, typically characterised by a higher dynamic background (Kong and Fu, 2018). Other environment-related issues may be lightings conditions changes, the partial or the total occlusion of the interested person and the localisation of a subject in a cluttered or dynamic environment. Some of these problems can be attenuated through the adoption of multiple cameras: in these cases, it is possible to offer a combined and consistent representation that allows to avoid the occlusion of the subject (concerning a particular point of view) or to make him more easily localised. The use of moving cameras makes it more challenging to address these issues.

- *Temporal variations*: a significant problem in real-world solutions is the ability to determine when an action takes place. Typically, HAR approaches evaluate time-segmented actions, delegating this task to action detection algorithms. However, the high variation in the execution rate of actions can affect their dimensional extent. A robust HAR

approach should guarantee invariance concerning the different rates of execution.

- *Obtaining and labelling training data*: the variations described above also have a substantial impact on the collection of sufficiently comprehensive datasets. The massive demand for labelled data by deep learning approaches has led to the spread of datasets of considerable size. However, several of these are domain-specific (e.g., Sports-1M by Karpathy et al. (2014)) or with annotations generated by retrieval methods (e.g., Youtube-8M by Abu-El-Haija et al. (2016)). In order to overcome these problems, it would be necessary to design algorithms that can learn actions in a non-supervised way or that can incrementally update the templates of the various classes of actions.

Certainly, other challenges of a non-technical nature should be discussed. The main issues are privacy-related implications. Especially in a sensitive environment like AAL, the idea of having a camera array installed in the home can be a significant obstacle for many people. In particular, the already common adversity of seniors towards technology would clash with the idea of being spied on continuously. This could lead to complete resistance to such approaches (Demiris et al., 2009). It is crucial to offer solutions able to preserve the user's privacy. Moreover, this feature must be fully understood and accepted by the user of a possible AAL remote monitoring service.

For many years, research has focused on the adoption of traditional RGB cameras, both for their extreme diffusion and for the high costs of other types of video-based sensors. In the last decade, we have witnessed a proliferation of low-cost depth sensors, including the well-known Microsoft Kinect. The spread of these sensors has intensely stimulated and influenced human motion analysis. One of the main reasons is that depth information allows mitigating some of the technical challenges described above, such as environmental variations in brightness, the presence of shadows and cluttered background. Besides, they enable the real-time extraction of skeletal joint positions (Shotton et al., 2011) that represent, since the dawn of human mo-

tion analysis, a fascinating object of study. Indeed, the expressiveness of Joint-Based representations was already introduced in the 70s through the Johansson's moving light-spots experiment in what is considered the very first work of Human Motion Analysis (Johansson, 1973). The experiment was aimed to study visual information and possible motion patterns deriving from different bright spots distributed on the human body. Observing the video collected by Johansson (few frames depicted in Figure 3.2), some typical limbs motion patterns are evident.

Figure 3.2: Johansson's moving light-spots experiment: handshake between two persons[1].

Moreover, a growing number of light-spots contributes to increasing the perception of the actions carried out. Although Johanson's pioneering experiment was a psychology study, it is the cornerstone of much of the literature on action recognition. In particular, many works based on skeletal information have been inspired by the bright-spots representation.

[1]Johansson's Experiment YouTube video

3.2.1 RGB-D sensors

In this thesis, approaches based on RGB-D sensors and information extracted from them (RGB, depth and skeletal) will be considered. The most common RGB-D sensors on the market are currently limited by a depth information extraction range of about 6-7 meters. This makes their use mainly possible in indoor environments.

In the context of AAL, a vast majority of works and datasets have focused on multi-modal approaches and in particular on the extraction of information from RGB-D sensors. While the adoption of these sensors allows alleviating some significant low-level challenges typical of traditional RGB approaches, on the other hand, the occlusion remains a significant problem that can worsen with the extraction of skeletal information. The problem

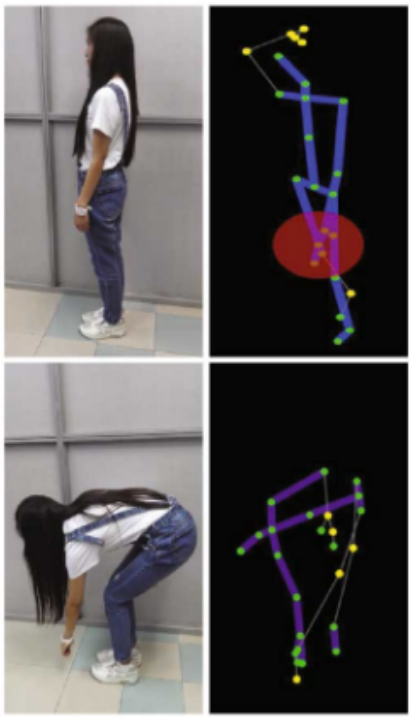


Figure 3.3: Skeleton corruption problem in RGB-D sensor caused by self-occlusion (Zhang et al., 2017).

.

of corrupted skeletal models, mainly due to tracking errors, environmental occlusion or self-occlusion, is well known in the literature. An example of a noisy skeleton due to the self-occlusion of some body parts is shown in Figure 3.3. To counter this problem and increase the reliability of extracted skeletal data, (Chaaraoui et al., 2013b) propose the feature fusion between skeletal features and silhouette-based features. Finally, another problem that must be considered is *depth camouflage* (an example is reported in Figure 3.4). This circumstance occurs when the foreground objects are very close in depth to the background. In this regard, (Camplani et al., 2017) published a dataset to evaluate the robustness of approaches of foreground movement objects detection compared to the significant challenges observed in RGB-D contexts.

Figure 3.4: Depth camouflage problem examples: as observable some foreground objects, including a subject, are hardly distinguishable from the background (Camplani et al., 2017).

.

Despite these challenges, the multiple advantages due to the extraction of structural information about the environment and the subjects allow a more straightforward implementation of recognition view-invariant systems. Last but not least, the RGB-D sensors, if limited to the extraction of skeletal information and the use of depth data, are much more privacy preserving than traditional video-cameras. This ability can be easily explained – and

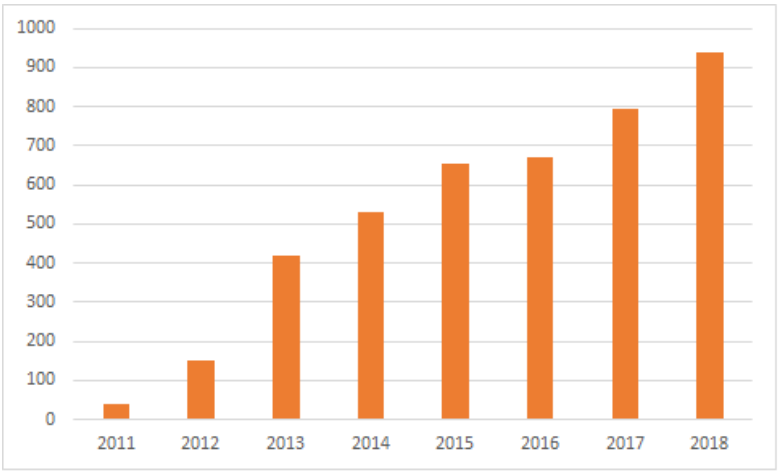


Figure 3.5: Number of publications per year extracted from the Scopus database by searching the keyword "RGB-D".

especially shown – even to the most adverse inhabitants to the idea of being filmed.

Microsoft Kinect has allowed considerable development of scientific research on vision-based HAR and defined the study of more performing devices and technologies for the demands of various academic and industrial research groups. In Figure 3.5 is reported the number of indexed scopus publications that contain the keyword "RGB-D".

Figure 3.6: Azure Kinect DK Camera[2].

In the end, what had to be a simple video-ludic entertainment device led to a real revolution, enabling the reduction of costs and the consequent proliferation of these sensors. While this thesis is being written, Microsoft is starting to release the newer Azure Kinect (see Figure 3.6). This promising device marries the vision of this work, both by optimizing data representation/acquisition – primarily depth channel and, consequently, skeletal information – and, in particular, by trying to create a rapid interface between the device and IoT cloud platform.

[2]Azure Kinect DK Website.

Chapter 4

Internet of Things: an overview

> *"Today computers — and, therefore, the Internet — are almost wholly dependent on human beings for information. (...) The problem is, people have limited time, attention and accuracy — all of which means they are not very good at capturing data about things in the real world. And that's a big deal. We are physical, and so is our environment."*
>
> – Kevin Ashton, *"That 'Internet of Things' Thing"*, 2009

Kevin Ashton first coined the term Internet of Things (IoT) in 1999. His vision, partially similar to *Ubiquitous Computing*, indicated a wholly connected futuristic world in which *intelligence* and communication skills are integrated into the surrounding environment. Following this definition, there is a partial overlap with the concept of Ambient Intelligence introduced in Section 2.2. Indeed, for several years, it has been difficult to distinguish between different terminologies indicating similar paradigms such as IoT, AmI, Smart Cities, Pervasive and Ubiquitous Computing. Restricting this analysis to the concept of the IoT, a more precise and formal definition is offered by (Guinard and Trifa, 2016):

> "*The IoT is a system of physical objects that can be discovered, monitored, controlled, or interacted with by electronic devices that communicate over*

various networking interfaces and eventually can be connected to the wider internet."

One element that needs to be clarified about the terminology used is the potential overlap that the reader may find at this point. Indeed, as noted in Section 2.2, a key feature of AmI environments – regardless of their granularity – is the presence of a physical infrastructure consisting of sensors, actuators and networks that act in a collaborative way. This element – which we have defined as smart environment – constitutes a specific intersection with the above definition of the IoT. Of course, the IoT adds a pervasive connectivity that could potentially be excluded in the definition of a smart environment. Finally, what links the IoT paradigm to the creation of *intelligent environments* is the presence of a software layer that provides intelligence and decision-making capabilities (Mahmood, 2019). Interested reader could find a more comprehensive description of this terminology in (Augusto et al., 2013).

Nowadays, this world of interconnected objects is representing a point of no return for both the industry and research communities due to an unprecedented proliferation in the number of sensors and the broad spectrum of domains in which they can be exploited. In their 2015 report, McKinsey estimated the potential economic impact of the IoT to be between $4 trillion and $11 trillion annually by 2025 (Siow et al., 2018). This is widely understandable considering that the Business Insider Intelligence 2019 report, projects that there will be more than 64 billion IoT devices by 2025[1]. The report also focuses on two technologies that will revolutionize the market and IoT research by 2025, making it difficult if not impossible to predict future scenarios for this paradigm: Blockchain and 5G networks.

[1]Business Insider Intelligence - IoT Report 2019, preview

4.1 IoT for AAL

The IoT is now a consolidated reality of our daily lives: from cars to fitness sensors, from air conditioning systems to cameras, it is increasingly common to stumble upon devices that can communicate data with each other (Yeo et al., 2014). This vision can scale from the domestic domain to urban and regional scenarios, where sensor networks have become a common feature (Atzori et al., 2010; Bellavista et al., 2013; Siow et al., 2018).

The continuous development of scientific research in this broad field and the significant shift coming on the horizon are and will be extremely beneficial even for the AAL domain. These technologies can strongly foster the *ageing-well-at-home* vision, allowing a real transition from the traditional model of healthcare – centralized on specific buildings – to a model focused on the patient/elderly domestic environment. As an example, exploiting the capabilities and features of the IoT enables continuous communication between older adults and healthcare professionals or caregivers, a key feature of many monitoring scenarios; of course, this pervasive connectivity involves the elderly but, above all, the things and the environment in which he/she interacts, not necessarily with an explicit awareness. The apparent benefit is the possibility of using the Internet to periodically communicate – or immediately, in cases of chronic conditions – the collected data (Dohr et al., 2010).

In such a scenario, many technical and ethical challenges emerge (McCullagh and Augusto, 2011). Limiting our discussion to the former, (Gomes et al., 2017) highlight some of the main ones and emphasize how the IoT and cloud computing integration in an AAL system can be decisive in dealing with some of them, such as:

1. **Comprehensiveness of scenarios**: as highlighted in Section 2.3, the AAL scenarios are multiple. One of the main challenges is the difficulty of developing AAL systems able to comprehend the whole range of scenarios. Many factors have an impact in designing general solu-

tions: the level of patient mobility; the degree of cognitive and physical skills; the distance from healthcare professionals; and user location. An AAL system is comprehensive if it allows supporting different scenarios or possible variations, allowing patients to move from one scenario to another without affecting the available AAL services and without any loss of information.

2. **Reliable communication**: one of the most critical aspects of an AAL system, primarily when oriented towards patient monitoring and emergency communications, is the reliability of message transmission. Clearly, even delaying the delivery of relevant information can have serious consequences. Therefore, special attention is needed in the creation of protocols to support the routing and reliable delivery of messages carrying patient information.

3. **Heterogeneous Technologies**: developing software infrastructures that allow interaction with different sensors and actuators or that allow data-level integration is an open and extremely complex challenge. Besides, in a context in which several AAL environments are monitored, it can be assumed that each of them relies on different technologies and creates independent subsystems. This assumption should not be an obstacle to remote monitoring; hence, it is necessary to offer solutions that are as flexible as possible.

4. **Scalability**: AAL's services are becoming increasingly widespread, and it will be necessary to respond to ever-increasing demand. An AAL infrastructure must be prepared to accept a large number of environments and users. This implies more connections and a larger volume of data collected by the various sensors. Scalability implies the ability of the system to continue to offer services while meeting relevant requirements such as responsiveness, even in the face of increasing demand.

5. **Power Management**: it is essential to manage energy consumption wisely, particularly in the case of mobile sensors because of the use of

batteries. Besides, energy wastage due to continuous data collection should be avoided (e.g., monitoring of user-less environments).

The combination of IoT and cloud computing, together with the increasing attention paid to AAL systems, has allowed a plethora of different solutions and platforms. A general review of the huge literature on IoT-based AAL solutions goes beyond the scope of this section but interested readers can refer to (Dang et al., 2019) and (de Morais Barroca Filho and de Aquino Junior, 2017) for good recent surveys.

Here, some of the main IoT-Based AAL projects – from an IoT platform perspective – will be presented. Among these, it is necessary to mention the UniversAAL IoT[2] platform, the primary outcome of the UniversAAL project (Ram et al., 2013) whose objective was to provisioning IoT AAL services. The platform defines an open-source semantic framework that allows, in particular, the communication between universAAL-enabled services and sensors based on an ontological description of their data models. One of the main features is the possibility of communication between applications and sensors, regardless of the node in which they reside. Indeed, they can immediately communicate if they reside in the same node, alternatively through gateways or RESTful API. (Bassoli et al., 2017) propose an interesting WiFi-based architecture for continuous monitoring in AAL environments. In particular, the authors analyzed possible countermeasures to power consumption due to the adoption of WiFi based sensors (e.g., a bed-occupancy sensor), while proposing a physical architecture (based on TI-CC3200 SoC) for power saving. In this work, it has been adopted a commercial cloud service, specifically IBM Bluemix. The primary goal of (da Silva et al., 2015) is to provide a platform for monitoring environmental conditions to protect subjects predisposed to asthma attacks. To this end, air quality is analysed using a WSN based on temperature and humidity sensors distributed in the different rooms of the monitored environments. The various sensors are interconnected to a local gateway via a ZigBee network. In (Almeida et al.,

[2]UniversAAL IoT Home Page

2019), the authors propose an IoT-aware AAL system for elderly monitoring. The system is defined by a general architecture for unobtrusively collecting data coming from a heterogeneous sensing infrastructure. The data collected concern the user motility described as the user body activities carried out (e.g., motion, rest, sleep, walking, etc.), the user/environment interaction (e.g., with home appliances or public services) and indoor/outdoor localisation. Indeed, the data collection from the sensing infrastructure takes place at home and city level as this interesting work is part of a larger research project called City4Age[3]. The primary goal of City4Age is to create Ambient Assisted Cities or age-friendly cities through the enhancing of the early detection of risk related to frailty and Mild Cognitive Impairment, and providing a personalised intervention that can help the elderly population to improve their daily life and also promote positive behaviour changes. The proposal of (Hail and Fischer, 2015) is focused on the study of an efficient and intelligent communication paradigm for IoT and AAL. To this end, the authors propose an IoT-AAL architecture via Information-Centric Network approach. In (Cubo et al., 2014), the authors exploit Google's cloud platform to define a framework that allows remote access and monitoring of data at run-time. Among the possible scenarios is identified the automatic notification in case of emergency. Specifically, the coupling between an accelerometer and a surveillance camera is realised: if a fast movement is detected, the camera starts to send the video streaming to a care centre.

Finally, the architecture proposed in (Balampanis et al., 2016), based on the cloud services offered by Fiware, introduces Microsoft Kinect as a fundamental component distinguishing two possible scenarios: *i)* Hospitalised Patient, *ii)* Rehabilitation. In the former case, Kinects are placed in each room in a strategic position (e.g., in front of the bed) to monitor, through the skeletal information, any movements of the patient or requests for help. The latter scenario studies the platform's adoption for user's remote rehabilitation monitoring by a hypothetical physiotherapy centre. In such a context,

[3]City4Age Home page

Kinect is used to monitor the exercises carried out at home by the patient; the doctor can thus monitor the rehabilitation process based on records of the patient's movement history and the time incurred. The primary purpose is to make the patient autonomous and independent.

Several IoT-based healthcare approaches, as well as those presented, are designed on the basis of the classic 4-layer framework (Wan et al., 2017):

- **Sensing Layer**: collect information about the environment and its inhabitants, using a variety of sensors and smart devices such as those described in Chapter 3;
- **Network Layer**: it conceives various wireless communication technologies and techniques that enable pervasive computing, to efficiently collect, exchange and transmitting data;
- **Data Processing Layer**: aggregating, processing and analysis of sensed information, and the possible transformation into meaningful knowledge, such that users/environments information may be identified. This layer has to play the role of middleware between the physical world and services;
- **Application/Service Layer**: which delivers direct services to users.

Our proposal can also be compared to this reference framework. However, one of our goals is to provide a solution as transparent and detailed as possible concerning all layers, trying to break free from possible commercial black-boxes, typically adopted at the level of middleware. Also, we will try to give an answer to a well-known problem in IoT monitoring systems.

4.2 The interoperability issue

The advantages of adopting such technologies are clear, but the continuous spread of sensor networks has generated various and inconsistent environments. From the one hand, this condition poses several security threats

(Conti et al., 2018; Palmieri et al., 2017); on the other hand, the integration of uncorrelated sensor networks or heterogeneous objects could be anything but simple (Partynski and Koo, 2013; Gambi et al., 2016). We can imagine these sensor networks as pieces of a puzzle: in some cases their integration will be trivial while sometimes it could be extremely complicated. Suppose we want to make the data produced by different architectures accessible in an agile way through a single compact solution. For example, imagine an integration between data derived from several IoT-AAL solutions or, scaling up, from different urban sensor networks like Santander's network[4], the new *Array Of Things* in Chicago[5] and data from the Smart Citizen platform[6]. In such scenarios, we would inevitably face a number of problems, both due to the different nature of the nodes of the networks and to the different technologies and architectures adopted. The examples of sensor networks mentioned above offer a variety of sensors, as well as communication and storage protocols that are not shared. The absence of a clear design methodology that is widely adopted also makes this task rather difficult. With this principle in mind, several major (see Section 9.1) have released IoT platforms that address some of the needs mentioned above. Most of these solutions are offered in a ready-to-use fashion, lacking transparency and providing limited technical information along with high-level architectures and generic communication flows (Ray, 2016). This is clearly understandable in relation to business models: it would be unreasonable for a major to reveal relevant technological details and design choices adopted. Therefore, as stressed before, using such platforms implies a dependency on a sort of black-box.

In Part III, we tackle this issue by providing the scientific community with a possible solution for monitoring IoT environments comprised of heterogeneous sensor networks.

[4]http://maps.smartsantander.eu/
[5]https://arrayofthings.github.io/
[6]https://smartcitizen.me/

Part II

Human Action Recognition with RGB-D sensors

Chapter 5

Related works

Human action recognition is a very active research area and summarising the existing approaches is a quite hard task. Focusing on vision-based approaches, good reviews of the literature are provided in the recent surveys by (Liu et al., 2019) and (Herath et al., 2017). Many works adopt common RGB cameras to acquire information from the environment, but undoubtedly the widespread diffusion of low-cost RGB-D sensors, as stressed in Subsection 3.2.1, greatly boosted the research on this topic. Again, the most attractive feature of RGB-D sensors is the ability to capture depth images, coupled with the possibility of tracking rather accurately skeletons of individuals in the scene. The skeleton representation provided by Kinect, for example, consists of a set of joints, each described in terms of position and orientation in the 3D space. Such information is extremely useful for human activity analysis as confirmed by many approaches in the literature.

A first criterion to categorise the existing approaches is the input data type; most of the works exploit either the RGB images, depth data or skeleton information. It is worth noting that, in reality, the three categories are overlapped to some extent; methods exploiting a single data category have become quite rare and many works combine different information to improve robustness. Each method is then included in the category related to the main information exploited.

5.1 Action recognition from RGB images

The literature on purely RGB approaches is extremely vast. Here, it is summarised in relation to the RGB-D context and to some relevant concepts that have been employed in other information channel approaches. Many works adopt a representation of human actions based on a 3D volume, where the human pose and its variations are described both in space and time. The 3D volume is then encoded in different ways. (Gorelick et al., 2007) and (Yilmaz and Shah, 2005) use shape features, other approaches are based on optical-flow representation, for example in (Wang and Mori, 2011). The approach proposed in (Chaaraoui et al., 2013a) relies on the extraction of features derived from the points belonging to the contour of the human silhouette, determined by background subtraction. A specific action is then encoded by a holistic descriptor defined as a sequence of key poses. Finally, many works adopt local representations in place of holistic descriptors to better deal with noise. Space-Time Interest Points (STIPs) (Laptev, 2005) extends the *Harris* corner detector to *3D-Harris* detector. This kind of detector relies on points with significant spatial variations and non-constant motions (an example in Figure 5.1). STIPs have been used in several works and represent an interesting category of approaches, which demonstrated a good robustness to image variations. Different techniques for keypoints detection have been proposed, see for instance (Scovanner et al., 2007; Yeffet and Wolf, 2009), as well as different approaches for descriptor computation such as Histograms of Optical Flow (Laptev et al., 2008) and Histogram of Oriented Gradients features (Klaser et al.; Wang et al., 2009).

Several recent approaches exploit the potentialities of deep learning for activity recognition. Often the concept of 3D convolution (Ji et al., 2010) is used to capture temporal dynamics in a short period of time; other works model temporal dynamics by using multiple streams (Simonyan and Zisserman, 2014; Feichtenhofer et al., 2016; Carreira and Zisserman, 2017; Girdhar et al., 2017). A few works suggest (Khaire et al., 2018; Qi et al., 2018) the combined use of RGB, depth and skeletal data to improve action recognition

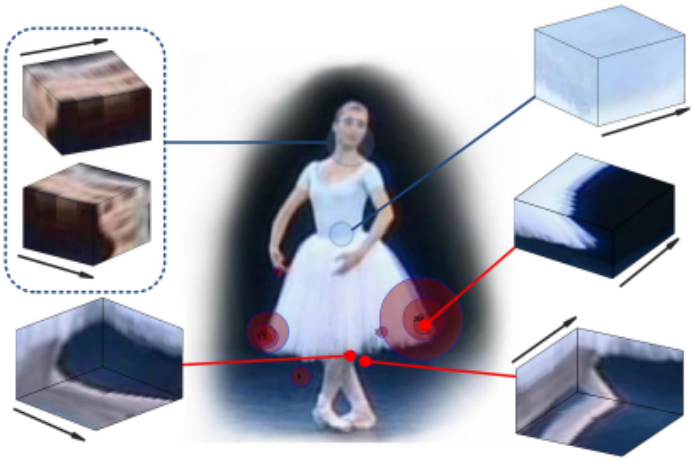


Figure 5.1: Example of Spatio-Temporal Interest Points (marked in red). The spatial changes with respect to the time axis are marked with an arrow. From the 3D volumes, it is clear that the dancer keeps her head still throughout the video. Despite the numerous spatial features, no STIPs are detected in her face or in her life (Herath et al., 2017).

accuracy.

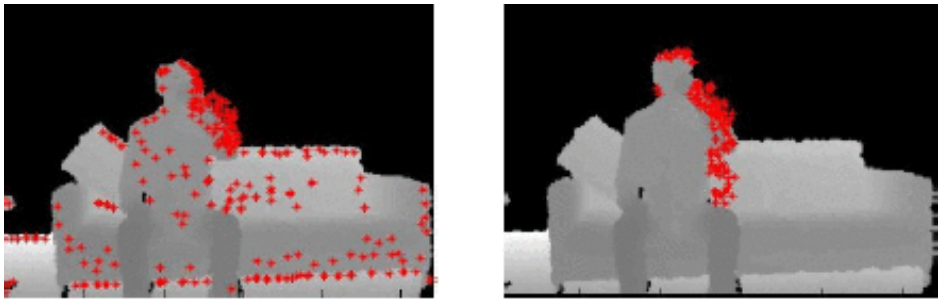
5.2 Action recognition from depth data

(Yang and Tian, 2014a) propose an approach aimed at extracting features from depth data only. In particular, they introduce a spatio-temporal depth sub-volume descriptors. In order to characterise the local motion and shape information, a polynormal is determined by the clustering of hyper-surface normals extracted from each depth sequence. To define the final representation of the depth map, the polynormals are aggregated into the Super Normal Vector (SNV), a simplified version of the Fisher kernel representation. The authors have also evaluated a possible integration with skeleton joint trajectories, in order to improve recognition results in sequences with many human movements. In (Gupta et al., 2013) is presented a silhouette-based descrip-

tor which couples depth and spatial information to define human poses. The authors create a codebook of body poses and describe a new posture in terms of similarity to a codeword. In this approach, the human-body silhouette is extracted through background subtraction using the first frames of a sequence. Therefore, these frames should not include users in order to avoid affecting the robustness of the approach. In (Rahmani et al., 2014) is proposed the Histogram of Oriented Principal Components (HOPC) descriptor which is able to describe the shape and motion information from a sequence of 3D points. The authors evaluate two different settings: holistic and local. In the holistic approach, the sequence of 3D pointclouds is split into spatio-temporal cells, and each cell is described by accumulating and normalising the individual HOPC descriptors belonging to that cell. A sequence descriptor is formed by concatenating the different cells' HOPC descriptors. In the latter approach, local HOPC are extracted at candidate Spatio-Temporal Key-Points (STKPs), and quality evaluation is performed to rank the STKPs. Another example of holistic descriptor is the Histogram of Oriented 4D Normals (HON4D) (Oreifej and Liu, 2013). HON4D describes the depth sequence by means of a histogram that captures the distribution of the surface normal orientation in the 4D space of time, depth, and spatial coordinates.

In (Li et al., 2010) is presented an approach based on a bag of 3D points and on an action graph. The former one, is used to represent the most relevant postures; the latter one, is exploited to describe the dynamics of the actions, in which each node represents a particular posture. The authors emphasise that good recognition accuracy has been achieved using only 1% of the 3D points. In particular, the points used are obtained by planar projections of the 3D depth maps and through the extraction of those that are on the human body contours.

(Xia and Aggarwal, 2013) explored the use of depth-based STIPs. The authors defined an interesting solution to deal with depth noise. They proposed a correction function in order to avoid some typical problems of depth



(a) without correction function (b) with correction function

Figure 5.2: Depth-based STIPs: noise correction example. The points are projected on top of one frame of the action *drink* (Xia and Aggarwal, 2013).

frames, such as the boundary of objects – due to the transition between foreground and background – or "depth holes" – caused, for example, by particular materials or fast movements. An example of the noise correction function application is shown in Figure 5.2.

Several works have used depth information in multi-modal solutions. Such an example, the previously mentioned work by (Chaaraoui et al., 2013b) proposed a bimodal solution composed by the concatenation of silhouette-based and skeletal features. (Chen et al., 2015) proposed the 3.5D depth video representations that corresponds to the outcome of reconstructing 3.5D information from depth spatio-temporal features, learned through Convolutional Neural Networks (CNNs), and the skeleton data (3D joint positions). Data are fused at the kernel level using an ensemble Multi-Kernel Learning (MKL) framework where each component classifier is a discriminative MKL. (Althloothi et al., 2014) also proposed an approach that employs MKL algorithms based on depth shape features, extracted using spherical harmonics representation and used to describe the 3D silhouette structure, and motion features, based on the estimated 3D joint positions. Finally, in (Zhu et al., 2014) an approach based on STIPs derived from the depth images is proposed. In this method, the skeleton data are used to create a bounding box exploited to remove irrelevant interested points (an example is shown in

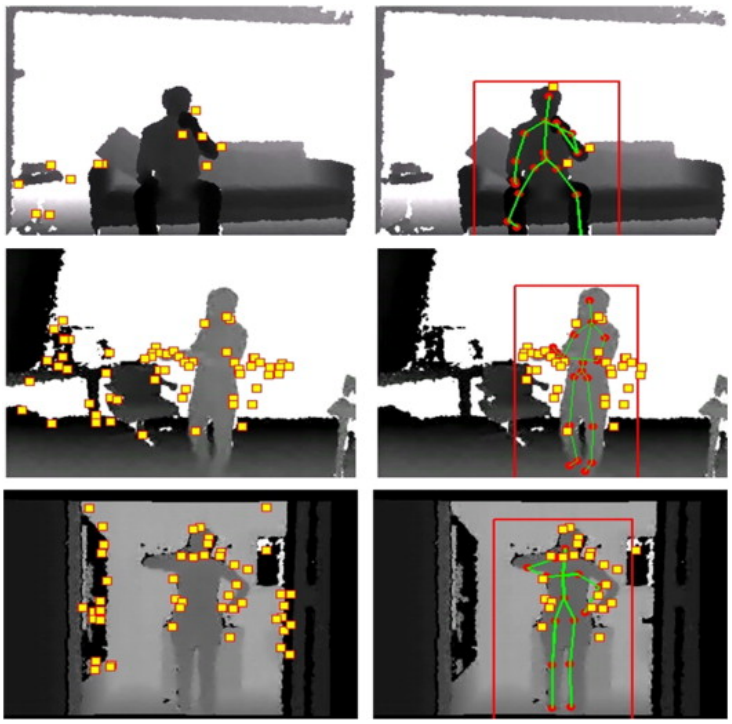


Figure 5.3: Depth-based STIPs refinement through skeleton data (Zhu et al., 2014).

Figure 5.3).

The potential of CNNs is also exploited in (Wang et al., 2016) where the authors propose an architecture based on three distinct CNNs with final classification obtained through a late score fusion. Each CNN accepts as an input a sequence of different depth maps which are constructed by projecting the 3D points to the three orthogonal planes.

5.3 Action recognition from skeleton data

Most of the approaches based on RGB-D data perform a skeleton analysis, adopting different representations of the set of joints such as the simple joint coordinates, normalised according to some body reference measure. Such as an example, (Gaglio et al., 2015) proposed an approach based on normalised

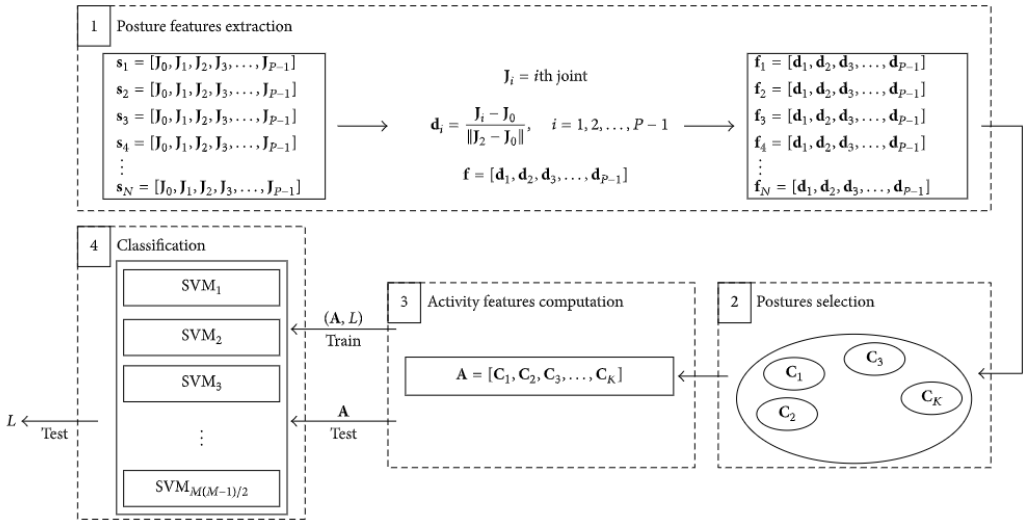


Figure 5.4: HAR algorithm constituted by four steps: posture extraction, posture selection, activity features and classification (Cippitelli et al., 2016).

feature vectors of 3D joints coordinates. The normalisation process uses two reference joints: torso and neck. The scale factor is defined as the ratio of the distance between these two joints and the same distance concerning a reference skeleton (detected offline). A translation matrix is finally applied to set the origin of the coordinate system to the torso. Instead, (Shan and Akella, 2014) normalised the 3D skeleton data using the skeleton height and shoulder width, in order to reduce the influence of different heights and limbs length.

Another approach is to use joints distances instead of normalised 3D joints coordinates, see for instance (Cippitelli et al., 2016) in which sets of distance vectors – with respect to the torso joint – are used. Also in this work, summarised in Figure 5.4, the feature normalisation is obtained exploiting the distance between the neck and torso joints. The posture selection phase (step 2 in Figure 5.4) consists of applying a k-means algorithm to the distance vectors that characterise a specific action. The purpose is the definition of k-centroids (i.e., key poses) that allow the subsequent creation of a particular

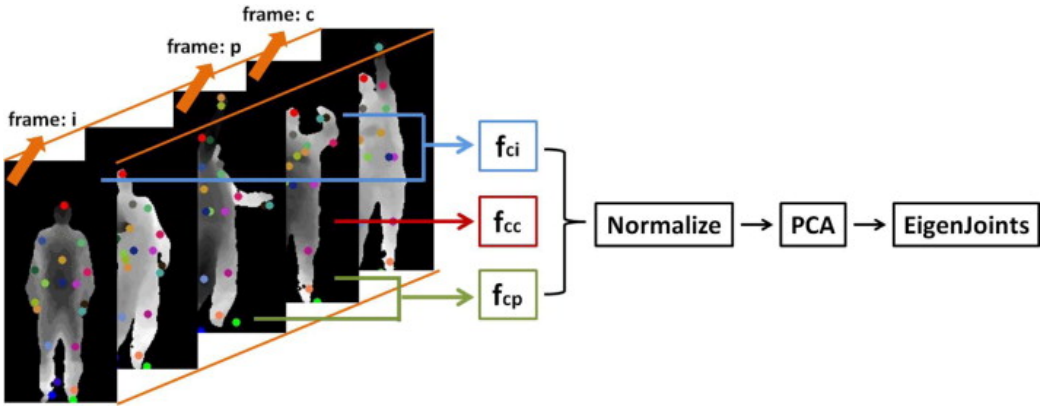


Figure 5.5: The framework for representing EigenJoints. In each frame, are computed three feature channels: f_{ci}, f_{cc} and f_{cp} (Yang and Tian, 2014b).

activity feature vector, that is sorted according to the order in which the different poses are assumed. This allows to preserve the temporal information of a specific sequence.

In (Yang and Tian, 2014b) is introduce a framework to represent EigenJoints (shown in Figure 5.5). This approach uses three different features channels: one for static feature postures (represented by the configuration of joints in a given frame, described by pair-wise joints differences), one for consecutive motion features (represented by the difference in configuration compared to the previous frame) and finally, one for overall dynamics feature (represented by the difference in configuration compared to the initial frame). Normalisation and Principal Component Analysis (PCA) are then applied to create an EigenJoints-based motion model.

In Xia et al. (2012) is defined the Histograms of 3D joints locations (HOJ3D). This histogram is obtained by partitioning the 3D space into various bin. Using the spherical coordinates defined by the authors, a specific joint can be located in a single bin. In Zhang and Tian (2012) kinematic features, obtained observing the angles between couples of joints, are used. Also in (Theodorakopoulos et al., 2014) angles are exploited. A human action is in fact represented as a sequence of poses over time, in which each pose is

described by 8 pairs of angles. Other representation include, for example, Gaussian Mixture Models representing the 3D positions of skeleton joints in (Piyathilaka and Kodagoda, 2013) or Dynamic Bayesian Mixture Model of 3D skeleton features in (Faria et al., 2014). Another common approach is to adopt a hierarchical representation where an activity is composed of a set of sub-activities, also called *actionlets* (Wang et al., 2012, 2014; Koppula et al., 2012; Ding et al., 2015). In the recent work by (Qi et al., 2018) is proposed an automatic joint configuration learning method, based on dictionary learning and sparse representation.

The interaction of humans with objects is analysed in a few works. Such as an example, the authors of Koppula et al. (2012) adopt a Markov Random Field where the nodes represent objects and sub-activities and the edges represent the relationships between object affordances, their relations with sub-activities, and their evolution over time; in (Koppula and Saxena, 2013) the authors propose a graph-based representation.

Also for skeletal data classification some recent techniques based on deep learning have been proposed. Long Short-Term Networks (LSTMs) are well suited to this aim for their capabilities of processing changes across time (Shahroudy et al., 2016; Cui et al., 2018; Battistone and Petrosino, 2019).

5.3.1 Joint orientations

Most of the skeleton-based approaches use the position of joints as primary information and, as we have seen, this involves different normalisation procedures. Our approach, that will be presented in the next chapter and exploited in the following ones, uses another information modality for human representation construction: the joint orientations. The main advantage of adopting orientation-based features is that they are invariant to human position, body size and sensor orientation (Han et al., 2017). It is possible to differentiate between two macro-categories of approaches: *i)* spatial orientation of pairwise joints, *ii)* temporal joint orientation.

In the former case, the orientation of displacement vectors of a pair of joints acquired at the same time step is computed. An example, although of gesture recognition, is reported in (Gu et al., 2012), in which the orientation of each joint is described by angles to a 3D centroid, represented by the joint torso. Similarly, in (Sung et al., 2012) the orientation matrix of each joint is computed concerning the RGB-D sensor, then each joint's rotation matrix is transformed with respect to the person's torso. Otherwise, in (Jin and Choi, 2012) are computed 19 first-order vectors, that are the orientation vectors from one joint to another one, and 14 second-order vectors, defined on the basis of a neighbourhood strategy to connect adjacent vectors. In a recent work by (Khokhlova et al., 2019) joint orientations are exploited to calculate kinematic gait parameters. In particular, orientations are used to calculate angles between lines connecting relevant pairs of joints (e.g., the angle between a line connecting left and right hip joints).

In the latter case, these approaches usually rely on the difference over time of the same set of joints. A very first example can be found in the work of (Campbell and Bobick, 1995) in which torso orientation trajectories are evaluated. Finally, (Boubou and Suzuki, 2015) introduced the Histogram of Oriented Velocity Vectors (HOVV) descriptor. The skeleton sequence is represented using a 2D spatial histogram that captures the distribution of the orientations of velocity vectors.

Chapter 6

Joint Orientations for HAR

In this chapter is presented an action recognition approach where angle information is used to encode the human body posture, i.e. the relative position of its different parts; such information is extracted from skeleton data (joint orientations), acquired by the cost-effective Kinect sensor. The system is evaluated on the well-known dataset CAD-60 for comparison with the state of the art; moreover, due to the lack of datasets including skeleton orientations, a new benchmark named Office Activity Dataset v.1.0 (OAD v.1.0) has been internally acquired. The tests confirm the efficacy of the proposed model and its feasibility for scenarios of varying complexity.

6.1 Introduction

As stressed in Section 3.2, automated high-level human action analysis and recognition play a fundamental role in many relevant and heterogeneous application fields such as video-surveillance, AAL, automatic video annotation or human-computer interfaces. Of course, different applications need specific approaches to be designed and implemented; general-purpose solutions, though highly desirable, are tough to implement due to, for example, the differences in the source of information, the requirements in terms of efficiency, the environmental factors which have a significant impact on per-

formance. This approach focuses on human action recognition in indoor environments which has typical applications in AAL, abnormal human behaviour recognition or, eventually, human computer interfaces. We have underlined unobtrusiveness as one of the most important and interesting features of AmI applications; to meet this requirement, the proposal of this chapter is a vision-based technique where simple cameras are used as input devices and the users are not require to wear neither to actively interact with sensors of different nature.

With respect to the other previously mentioned application scenarios such as video-surveillance, indoor environments offer several advantages: the input data are somehow more "controlled" and easier to process (e.g. to segment the subjects in the scene), the number of possible users is generally limited and input devices, such as RGB-D cameras, can be successfully adopted for data acquisition. The problem of action recognition is however still complex if we consider that the users are not cooperative and a real-time processing is needed to produce timely and useful information.

We have seen that many works in the action recognition literature are based on skeleton data. However, almost all only exploit 3D joint positions to describe human postures; since Kinect provides for each joint also the estimated orientation, we decided to explore the robustness of this information. We therefore derived a posture representation based exclusively on angle information, derived from both the joint position and orientation. Again, the great advantage of angle features derived from skeletons is that they are intrinsically normalised and independent from the user's physical structure. A good degree of invariance with respect to pose and view changes is also achieved since all the angles are computed with respect to the subject's coordinate system.

Aim of this proposal is hence to evaluate the reliability of the joint orientation estimates provided by Kinect and to verify their effectiveness for action recognition.

6.2 Proposed approach

The fundamental idea behind the proposed approach is to encode each frame of a video sequence as a set of angles derived from the human skeleton, which summarise the relative positions of the different body parts. This proposal presents some advantages: the use of skeleton data ensures a higher level of privacy for the user with respect to RGB sequences, and the angle information derived from skeletons is intrinsically normalised and independent from the user's physical build.

The skeleton information extracted by the Kinect (Shotton et al., 2013) consists of a set of n joints $J = \{j_1, j_2, ..., j_n\}$ where the number n of joints depends on the software used for the skeleton tracking (i.e. typical configurations include 15, 20 or 25 joints). Each joint $j_i = \left(\mathbf{p_i}, \overrightarrow{\mathbf{o_i}}\right)$ is described by its 3D position $\mathbf{p_i}$ and its orientation $\overrightarrow{\mathbf{o_i}}$ with respect to "the world". Our approach exploits the information given by joint orientations to compute relevant angles whose spatio-temporal evolution characterises an action. We consider three different families of angles (see Figure 6.1 and Figure 6.2):

- θ_{ab}: angle between the orientations $\overrightarrow{\mathbf{o}_a}$ and $\overrightarrow{\mathbf{o}_b}$ of joints j_a and j_b. Angles θ_{ab} are computed for the following set of couples of joints:

$$A_\theta = \{(j_1, j_3), (j_1, j_5), (j_3, j_4), (j_5, j_6), (j_0, j_{11}), (j_0, j_{12}), (j_7, j_8), (j_9, j_{10})\}$$

- φ_{ab}: angle between the orientation $\overrightarrow{\mathbf{o}_a}$ of j_a and the segment $\overrightarrow{j_a j_b}$ connecting j_a to j_b. We can consider the segment as the bone that interconnects the two joints. Angles φ_{ab} are computed for the following set of couples of joints:

$$A_\varphi = \{(j_3, j_1), (j_3, j_4), (j_4, j_3), (j_4, j_{11}), (j_{11}, j_4), (j_5, j_1), (j_5, j_6), (j_6, j_5),$$

$$(j_6, j_{12}), (j_{12}, j_6), (j_2, j_7), (j_7, j_2), (j_7, j_8), (j_2, j_9), (j_9, j_2), (j_9, j_{10})\}$$

- α_{bac}: angle between the segment $\overrightarrow{j_a j_b}$ connecting j_a to j_b and $\overrightarrow{j_a j_c}$ that connects j_a to j_c. Angles α_{abc} are computed for the following triplets

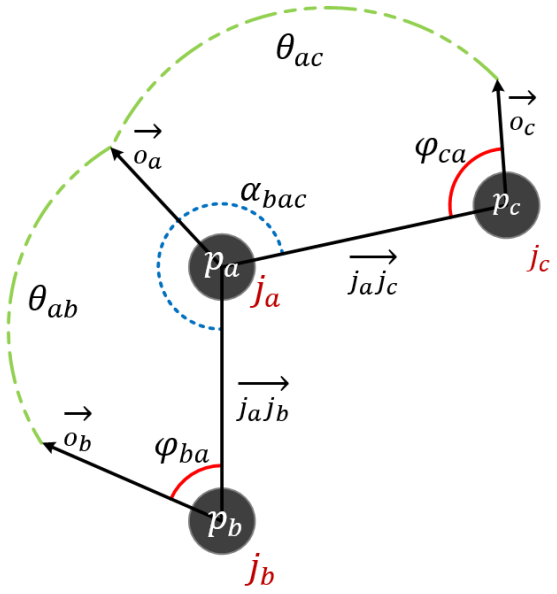


Figure 6.1: Representation of a subset of joints $j_a = (p_a, \overrightarrow{o_a})$, $j_b = (p_b, \overrightarrow{o_b})$ and $j_c = (p_c, \overrightarrow{o_c})$ and related angles θ, φ and α.

of joints:

$$A_\alpha = \{(j_2, j_7, j_8), (j_7, j_8, j_{13}), (j_2, j_9, j_{10}), (j_9, j_{10}, j_{14})\}$$

We consider only subset of the possible angles, mainly obtained from the joints of the upper part of the body, because not all the angles are really informative: for example, the angles between head and neck are almost constant over time and does not provide useful information for action discrimination. Different configurations of angles have been evaluated and compared in Section 6.3. Therefore, each frame f_i of the video sequence $S_i, i = 1, .., l$ is represented by a vector obtained as the ordered concatenation of the values of $\theta_i \mid i \in A_\theta$, $\varphi_j \mid j \in A_\varphi$, $\alpha_k \mid k \in A_\alpha$

$$\mathbf{v}_i = (\theta_1, ..., \theta_m, \varphi_1, ...\varphi_n, \alpha_1, ..., \alpha_s)$$

of size $(m + n + s)$.

It is worth noting that the number of frames for each video sequence can be extremely high and certainly not all the resulting feature vectors are

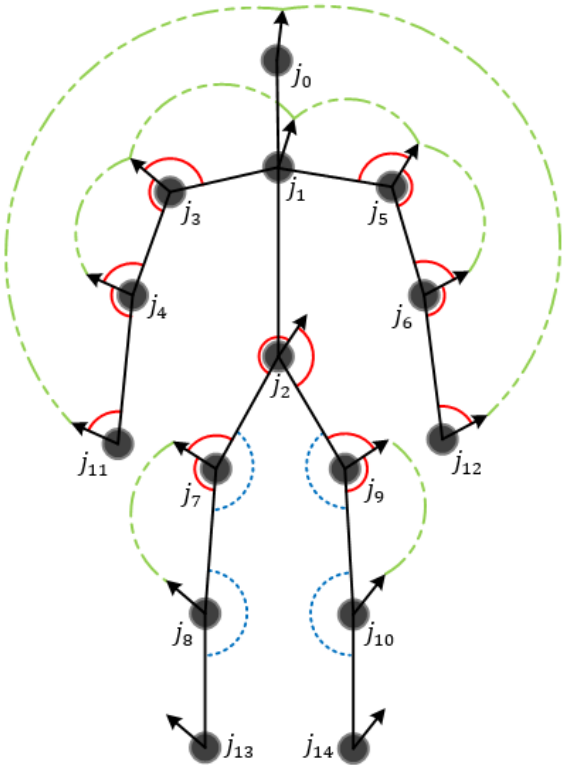


Figure 6.2: The 28 angles used in our experiments computed from a skeleton configuration with 15 joints.

significant: the variation of the angles between two subsequent frames is minimal and usually unnoticeable. We decided therefore to adopt a Bag of Word (BoW) model (Wang et al., 2009) with a two-fold objective: minimising the representation of each sequence keeping only the relevant information and producing fixed-length descriptor which can be used to train an action classifier. The idea is to represent each activity as an histogram of occurrences of some reference postures (see Figure 6.3 for a visual representation), derived from the analysis of the training set. A reference dictionary is first built by applying the well-known K-means clustering algorithm (Fukunaga, 1990) to the set of posture features extracted from the training sequences. Since some subjects could be left-handed, all the angle features are mirrored with respect to the x-axis. We denote with k the number of clusters determined (i.e. the

dictionary size). The dictionary should encode the basic postures assumed during the different actions in the training set and will be used to represent each sequence as an histogram of occurrences of such basic elements. Given a set of training sequences $TS = \{S_i, i = 1, .., d\}$, representative of the different actions, the k-means clustering algorithm is applied to the associated set of feature vectors $FV = \{\mathbf{v_i}, i = 1, .., d\}$ to obtain a set of k clusters: the cluster centroids are used as words of the reference dictionary $W = \{w_i, i = 1, .., k\}$. The number of clusters k determines the size of the dictionary and is one of the most relevant parameters of the proposed approach. Each sequence is then encoded as a normalised histogram of occurrences of the words in W. Of course the angle features are continuous values and a precise correspondence between the words in the dictionary and the descriptors is very unlikely; therefore when computing the histogram each feature vector f_i is associated to the closest word w_j^* in the dictionary: $j^* = \text{argmin}_j ||f_i - w_j||$. A Random Forest Classifier (Breiman, 2001) is trained to discriminate the different actions represented in the training set; the classifier consists of an ensemble of decision trees, each trained on a subset of the patterns and a subset of the features and the final classification is obtained combining the decisions of the single sub-trees.

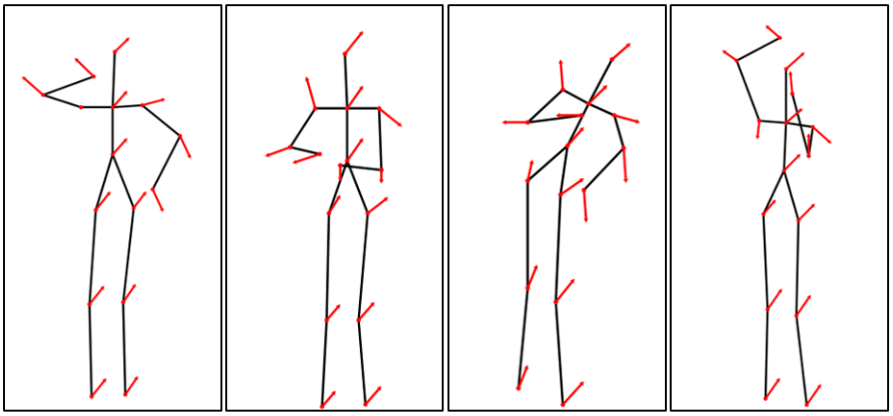


Figure 6.3: Visual representation of a subset of key poses corresponding to some cluster centroids of the dictionary W.

6.3 Experiments

Several experiments have been conducted to evaluate the sensitivity of the proposed approach to its main parameters (i.e. the set of angles selected and the dictionary size). Despite of the large number of existing benchmarks for action/activity recognition from skeleton information, joint orientations are generally not available. We used for testing the well-known CAD-60 (Sung et al., 2012, 2011), released by the Cornell University, and a newly acquired dataset. CAD-60 contains 60 RGB-D videos where 4 different subjects (two male and two female, one left-handed) perform 12 ADL in 5 environments (office, kitchen, bedroom, bathroom and living room). The authors of the benchmark propose two settings named *new person*, where a leave-one-out cross-validation is adopted, and *have seen* where the training set includes data from all the subjects. We adopted the *new person* testing protocol, in accordance with all the related works in the literature, to allow for a comparison of the results. Moreover, analogously to other works, the recognition accuracy is measured separately for the different rooms.

6.3.1 Office Activity Dataset v.1.0

Due to the lack of datasets including information on joint orientations, we decided to acquire a new database of human actions to perform further tests. Data acquisition was carried out in a single environment (office) from several perspectives based on the action being performed. From this point of view the benchmark is more complex than CAD-60 because all the activities need to be compared for recognition and the higher number of subjects increases the variability of each action. It contains 14 different actions: *drinking, getting up, grabbing an object from the ground, pour a drink, scrolling book pages, sitting, stacking items, take objects from a shelf, talking on the phone, throwing something in the bin, waving hand, wearing coat, working on computer, writing on paper*. Data was collected from 10 different subjects (five

Figure 6.4: Example frames of some activities carried out by the 10 subjects in the OAD v.1.0. Specifically: *drinking*, *talking on the phone*, *throwing something in the bin*, *sitting/getting up*, *grabbing an object from the ground*, *wearing coat*, *pouring a drink*, *scrolling book pages*, *waving hand* and *working on computer*

males and five females) aged between 20 and 35, one subject left-handed. The volunteers received only basic information (e.g. *"pour yourself a drink"*) in order to be as natural as possible while performing actions. Each subject performs each action twice, therefore we have collected overall 280 sequences. Some examples RGB frames are shown in Figure 6.4.

The device used for data acquisition is the Microsoft Kinect V2 whose SDK allows to track 25 different joints (19 of which have their own orientation). For testing, we adopted the same *"new person"* setting of the CAD-60 dataset: a leave-one-out cross-validation with rotation of the test subject. The set of angles used for testing the proposed approach is however the same used for CAD-60. At this moment, only skeletal data are available in the Smart City Lab web site[1]. As will be described in the next chapter, we have collected an extension of this dataset and we will release the final version of

[1] `http://smartcity.csr.unibo.it`

Office Activity Dataset (OAD).

6.3.2 Results

CAD-60. Performance evaluation starts from the analysis of the confusion matrix M where a generic element $M(i, j)$ represents the percentage of patterns of class i classified by the system as belonging to class j. Further synthetic indicators can be derived from the confusion matrix; in particular, we computed precision P and recall R as follows:

$$P = \frac{TP}{TP + FP}, R = \frac{TP}{TP + FN}$$

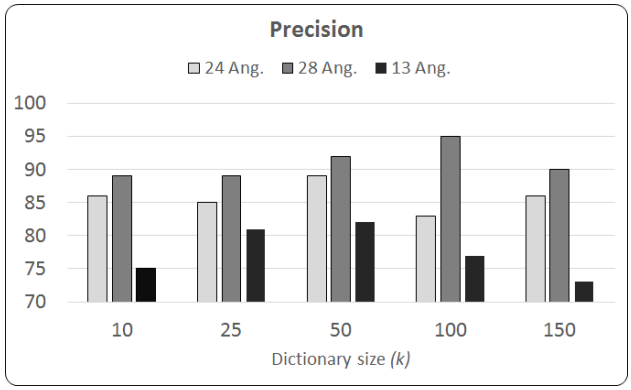
where *TP, FP* and *FN* represent respectively the True Positives, False Positives and False Negatives which can be easily derived from the extra-diagonal elements of the confusion matrix. In analogy to the proposal in (Cippitelli et al., 2016), each video sequence is partitioned into three sub-sequences which are used independently in the tests.

The results obtained are summarised in Figure 6.5 where the Precision (P) and Recall (R) values are reported for different experimental settings, i.e. variable dictionary size (k) and three subsets of angles considered for skeleton representation. In particular, the efficacy of the joint orientations is assessed by comparing the results of two different settings – 24 angles, (α angles omitted) and 28 angles – with those obtained using only A_α angles, computed between all the existing pairs of neighbouring segments (13 angles, no joint orientation is used in this case).

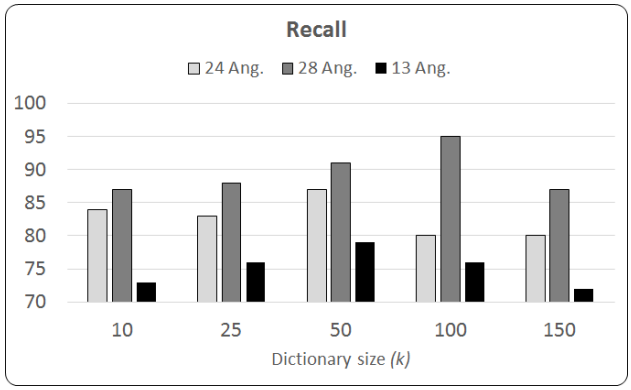
The results show that, overall, the accuracy of the proposed technique is good. As expected the dictionary size has a significant impact on the performance; it is worth noting that different actions have often very similar postures (e.g. drinking and talking on the phone) and a value of k excessively low probably determines the reference posture of such activities to collapse in a single word, thus making difficult to correctly distinguish them. On the other hand, a high value of k produces very sparse feature vectors, more sensitive to the presence of noise. The best results have been reached with

Table 6.1: Precision and Recall for the different rooms of CAD-60 with leave-one-out cross validation, $k = 100$ and the 28 angles configuration.

Room	Activity	Precision (%)	Recall(%)
Bathroom	brushing teeth	80.0	100
	rinsing mouth	100	75.0
	wearing contact lenses	100	100
	Average	**93.33**	**94.86**
Bedroom	talking on phone	100	91.67
	drinking water	80.0	100
	opening pill container	100	100
	Average	**97.43**	**97.22**
Kitchen	cooking (chopping)	86.0	100
	cooking (stirring)	100	83.0
	drinking water	100	100
	opening pill container	100	100
	Average	**96.43**	**95.83**
Living room	drinking water	92.31	100
	relaxing on couch	100	92.0
	talking on couch	92.0	100
	talking on phone	100	91.67
	Average	**96.15**	**95.83**
Office	drinking water	80.0	100
	talking on phone	100	75.0
	working on computer	100	100
	writing on whiteboard	100	100
	Average	**95.0**	**93.75**



(a)



(b)

Figure 6.5: Precision (a) and recall (b) values on CAD-60 with different configurations of angles, as a function of the dictionary size (k).

a value of $k = 100$ which also allows to efficiently perform the classification task. Also the angle configuration is important; the use of 28 angles produces better results both in terms of precision and recall with respect to the version with 24 angles.

The limited accuracy of the configuration with 13 angles, where the orientation is not exploited, confirm the effectiveness of joint orientation for accurate posture representation. These results also show that the significance of the angles varies greatly and a few strategical angles can greatly improve the recognition performance. As to the computational complexity, the proposed approach is very efficient, and all the angle configuration are

suitable for a real time processing.

Table 6.2: Confusion matrix using $k = 100$ words and a configuration of 28 angles on CAD-60.

	Talking on the phone	Writing on whiteboard	Drinking water	Rinsing mouth with water	Brushing teeth	Wearing contact lenses	Talking on couch	Relaxing on couch	Cooking (chopping)	Cooking (stirring)	Opening pill container	Working on computer
Talking on the phone	0.86		0.14									
Writing on whiteboard		1.0										
Drinking water			1.0									
Rinsing mouth with water				0.75	0.25							
Brushing teeth					1.0							
Wearing contact lenses						1.0						
Talking on couch							1.0					
Relaxing on couch							0.08	0.92				
Cooking (chopping)									1.0			
Cooking (stirring)									0.17	0.83		
Opening pill container											1.0	
Working on computer												1.0

Results for each activity, considering the different rooms, have been reported in Table 6.1. The confusion matrix, reported in Table 6.2, allows to analyse the main causes of errors. The mismatch occurred are all rather comprehensible since they are related to very similar actions (e.g. cooking-chopping, cooking-stirring). In these cases the skeleton information is probably too synthetic to discriminate the two actions which are very similar in terms of posture. A comparison with the state of the art is provided in Table 6.3 which summarises the results published in the benchmark website. Despite of the very good accuracy reached by different approaches in recent years, the proposed approach outperforms existing methods, both in terms of precision and recall.

OAD v.1.0. The results on the Office Activity Dataset v.1.0 are reported in Table 6.4 and Table 6.5 for the *standard* configuration with 28 angles and $k = 100$. The overall results confirm that this benchmark is more difficult for

Table 6.3: Precision (P) and recall (R) of the proposed approach on CAD-60, compared to the results published in the benchmark website. *"*"* indicates that a different protocol was used.

Algorithm	P	R
Proposed approach	**95.0**	**95.0**
(Sung et al., 2012, 2011)	67.9	55.5
(Koppula et al., 2012)	80.8	71.4
(Zhang and Tian, 2012)	86.0	84.0
(Ni et al., 2012)	Accur: 65.32	-
(Yang and Tian, 2014b)	71.9	66.6
(Piyathilaka and Kodagoda, 2013)	70*	78*
(Ni et al., 2013)	75.9	69.5
(Gupta et al., 2013)	78.1	75.4
(Wang et al., 2014)	Accur: 74.70	-
(Zhu et al., 2014)	93.2	84.6
(Faria et al., 2014)	91.1	91.9
(Shan and Akella, 2014)	93.8	94.5
(Gaglio et al., 2015)	77.3	76.7
(Parisi et al., 2015)	91.9	90.2
(Cippitelli et al., 2016)	93.9	93.5
(Urbano Nunes and Peixoto, 2017)	81.83	80.02
(Qi et al., 2018)	90.18	92.9
(Khaire et al., 2018)	93.06	90.0
(Battistone and Petrosino, 2019)	94.4	93.7

several reasons: i) the activities are not partitioned according to the room where they are performed and the probability of misclassification increases; ii) the number of subjects is higher and the variability in executing the actions increases proportionally. For instance, the worst results have been measured for the action "throwing something in bin" that the several subjects executed very differently.

Other mismatches occur between the actions "sitting" and "getting up"; in principle the reference postures of the two actions are similar, but their temporal ordering in the execution is different and probably the BoW representation adopted is not able to capture this aspect. However, in general,

Table 6.4: Precision (P) and Recall (R) values of the proposed approach for each action on OAD v.1.0.

Action	P	R
Drinking	60.87	77.78
Getting up	81.25	72.22
Grabbing object from ground	83.33	83.33
Pouring a drink	75.00	83.33
Scrolling book pages	80.95	94.44
Sitting	59.09	72.22
Stacking items	90.00	100.00
Taking objects from shelf	100.00	94.44
Talking on phone	86.67	72.22
Throwing something in bin	75.00	33.33
Waving	66.67	66.67
Wearing coat	100.00	100.00
Working on computer	94.12	88.89
Writing on paper	78.95	83.33
Overall	**80.85**	**80.16**

Table 6.5: Confusion matrix using $k = 100$ words and a configuration of 28 angles on OAD v.1.0.

	Drinking	Getting up	Grabbing obj.	Pouring a drink	Scrolling book	Sitting	Stacking items	Taking objects	Talking on phone	Throwing something	Waving	Wearing coat	Working on computer	Writing on paper
Drinking	0.78			0.06					0.11		0.06			
Getting up		0.72				0.28								
Grabbing obj.			0.83	0.06		0.06					0.06			
Pour a drink				0.83	0.17									
Scrolling book					0.94						0.06			
Sitting		0.17	0.06			0.72				0.06				
Stacking items							1.0							
Taking objects								0.94		0.06				
Talking on phone	0.17			0.06					0.72		0.06			
Throwing something	0.11		0.11		0.06	0.17	0.11			0.33	0.06			0.06
Waving	0.17			0.11							0.67			0.06
Wearing coat												1.0		
Working on computer													0.89	0.11
Writing on paper	0.06										0.06		0.06	0.83

the good performance of the proposed approach is confirmed on this dataset as well.

6.4 Final Remarks

A human action recognition technique based on skeleton information has been proposed in this chapter. In particular, the effectiveness of joint orientations, typically neglected by the majority of the action recognition works, has been evaluated on different benchmarks. The efficacy of the proposal have been confirmed; the results obtained overcome the state-of-the-art in the well-known CAD-60 benchmark and good accuracy levels can be reached also on the newly acquired OAD v.1.0 dataset.

Chapter 7

A multimodal approach for HAR

In this chapter the potentialities of the Kinect sensor are fully exploited to design a robust approach for action recognition combining the analysis of skeleton previously proposed and RGB data streams. The skeleton representation is designed to capture the most representative body postures, while the temporal evolution of actions is better highlighted by the representation obtained from RGB images. The experimental results confirm that the combination of these two data sources allow to capture highly discriminative features resulting in an approach able to achieve state-of-the-art performance on public benchmarks.

7.1 A multimodal system for action recognition

As we have seen, the Kinect sensor provides parallel access to different data streams; in this chapter we are interested in coupling information from both skeleton and RGB images. We will define an action as a sequence S of L data frames, $S_t, t = 1, .., L$; each element $S_t = (F_t, SK_t)$ includes F_t, the RGB frame acquired at time t (of size $W \times H$), and SK_t which is the corresponding skeleton. In practice the two data streams could be slightly misaligned, mainly due to the serialisation procedure which is not always able to work at the same frame rate the data are provided (a few frames could

thus be skipped some times). This misalignment was observed in several databases available for research purposes, but its impact on our approach is negligible since the contribution of the two information is combined at decision-level.

7.1.1 Skeleton

For the sake of the reader, the main aspects and categories of angles used by the approach proposed in the previous chapter are briefly reported. Each frame of a video sequence is represented by a set of angles derived from the human skeleton, which summarise the positions of the different body parts. The Kinect SDK represents the human skeleton as a set of d joints $J = \{j_1, j_2, ..., j_d\}$; each joint $j_i = \left(\mathbf{p_i}, \overrightarrow{\mathbf{o_i}}\right)$ is described by its 3D position $\mathbf{p_i}$ and its orientation $\overrightarrow{\mathbf{o_i}}$ with respect to "the world". To encode the user posture, we defined three types of angles:

- θ_{ab}: angle between the orientations $\overrightarrow{\mathbf{o}_a}$ and $\overrightarrow{\mathbf{o}_b}$ of joints j_a and j_b. θ_{ab} angles are computed from a set of m couples of joints A_θ ($m = 8$ in our work).

- φ_{ab}: angle between the orientation $\overrightarrow{\mathbf{o}_a}$ of j_a and the segment $\overrightarrow{j_a j_b}$ connecting j_a to j_b. φ_{ab} angles are computed from a set of n couples of joints A_φ ($n = 16$ in our work).

- α_{bac}: angle between the segment $\overrightarrow{j_a j_b}$ connecting j_a to j_b and $\overrightarrow{j_a j_c}$ that connects j_a to j_c. α_{bac} angles are computed from a set of s triplets of joints A_α ($s = 4$ in our work).

Unfortunately, the skeleton estimation provided by Kinect is not always accurate. The reliability is generally good for the joints of the upper part of the body, which contains most of the information needed for action recognition. Legs are generally quite unreliable, but in many cases they are occluded or almost static and do not provide significant contribution for action recognition. For this reason only a subset of the possible angles is considered,

mainly obtained from the joints of the upper part of the body. Each skeleton SK_t of the video sequence is represented by a vector obtained as the ordered concatenation of the values of $\theta_i \mid i \in A_\theta$, $\varphi_j \mid j \in A_\varphi$, $\alpha_k \mid k \in A_\alpha$

$$\mathbf{v}_i = (\theta_1, ..., \theta_m, \varphi_1, ...\varphi_n, \alpha_1, ..., \alpha_s)$$

of size $(m + n + s)$ where $m = |A_\theta|$, $n = |A\varphi|$ and $s = |A_\alpha|$.
The complete video sequence S is finally encoded using a BoW model where each action is represented as an histogram of occurrences of some reference postures. The skeleton BoW representation allows to effectively represent the main postures assumed by the human body during actions, but, as stressed before, the final representation does not capture the temporal evolution of the body movement (due to the global nature of the histogram representation). The temporal images described in the following subsection allow to better represent this aspect and provide a complementary representation with respect to the skeleton information.

7.1.2 HOG features from temporal images

In order to improve the recognition capabilities of the previously described approach, we developed a technique based on the analysis of RGB images with a two-fold objective: i) better encoding the temporal evolution of the action, needed to discriminate between actions characterised by similar postures but presented in a different order (e.g., sit down and get up); ii) capture to some extent the user interaction with objects which could help to classify the action. The feature extraction approach can be summarised into three main steps: *construction of the temporal images*, *gradient computation*, and *HOG encoding*.

7.1.2.1 Construction of the temporal images

We can represent a sequence of frames F_t with $t = 1, .., L$ as a volume image V (see Figure 7.1a), i.e. a parallelepiped in a 3D space (x, y, t), where the first two coordinates refer to the spatial coordinates of the frame pixels

and the third one represents time. To achieve independence from the body position in the images, each frame is cropped to a fixed-size window (25% of the frame width) centred on the spine mid joint. The width of the region of interest has been empirically determined, based on a rough analysis of the training set, and is not always accurate for the test sequences; however, it represents a good trade-off between computational complexity and accuracy. Our representation is obtained by a slicing operation of the volume V at predefined values of the y-coordinate (see Figure 7.1), properly selected to capture the body motion during the action. In particular, a set $T = \{T_{y_1}, T_{y_2}, ..., T_{y_M}\}$ of M temporal images of size $W \times L$ will be computed from V; the generic element of T is defined as: $T_{y_i}(r, c) = V(r, y_i, c)$ with $r \in [1, .., W]$ and $c \in [1, .., L]$. Examples of temporal images at fixed values of y are given in Figure 7.1b. As clearly visible in the example, the temporal image highlights the specific movement of a body region during time; the slice at the level of hands will show a very typical periodic movement originated by the steering action performed. Other temporal images, for instance from the leg region, will be more static for this specific action. As expected, the selection of the sections to analyse (y values) has an important impact on the accuracy of representation. We evaluated two strategies: i) y value of the main skeleton joints; ii) uniform sampling along the skeleton. The two approaches will be compared in the experiment section.

7.1.2.2 Temporal image gradient computation

Looking at the temporal images, it is easy to observe that the relevant information for action recognition is represented by the dynamic elements, the variations observed across time; the constant regions of the image are not interesting and must not be encoded. For this reason we convert each temporal image $T_{y_i} \in T$ in a grayscale image and we compute the gradient moduli G_{y_i} using the Sobel operator (see Figure 7.1c). Even if the RGB frames look quite defined, an analysis of the gradient images reveals the presence of a significant noise component that must be removed for reliable

feature extraction. A denoising operation is therefore applied both before and after gradient computation to reduce the effects of inter-frame variations due to the sensor, thus obtaining a regularised image $\tilde{G_{y_i}}$; the technique used for denoising is non-local means denoising (Buades et al., 2011).

7.1.2.3 HOG encoding of gradient images

Each regularised gradient image $\tilde{G_{y_i}}$ is finally encoded by HOG descriptors proposed in (Dalal and Triggs, 2005). The length of the different sequences could be different of course, thus determining temporal images of different size. We need however a fixed-length descriptor to train a classifier, so each image is partitioned into a fixed number of overlapping blocks and the final descriptor is obtained by the concatenation of the block descriptors. The OpenCV implementation of HOG descriptor computation has been used here; in particular, best results were achieved with a window of 4x8 cells. The size of the cells for a specific sequence obviously depends on the size of the input temporal image. A L2 normalisation is carried out on blocks made of 4x4 cells. The adoption of a histogram-based representation allows to further reduce the influence of noise.

7.1.3 Action classification

The two techniques discussed in the previous sections are quite complementary and their fusion can be useful to achieve good recognition accuracy. As shown in Figure 7.2, two classifiers are trained using the features extracted from skeleton and RGB images respectively. As for the skeleton data, we used the same configuration described in the previous chapter with the training of a Random Forest classifier. The second classifier consists of a set of M linear Support Vector Machines where each SVM represents a T_y slice, i.e. each model is trained on a specific volume slice; the classification of a particular action is carried out by the fusion of results obtained from the individual SVMs.

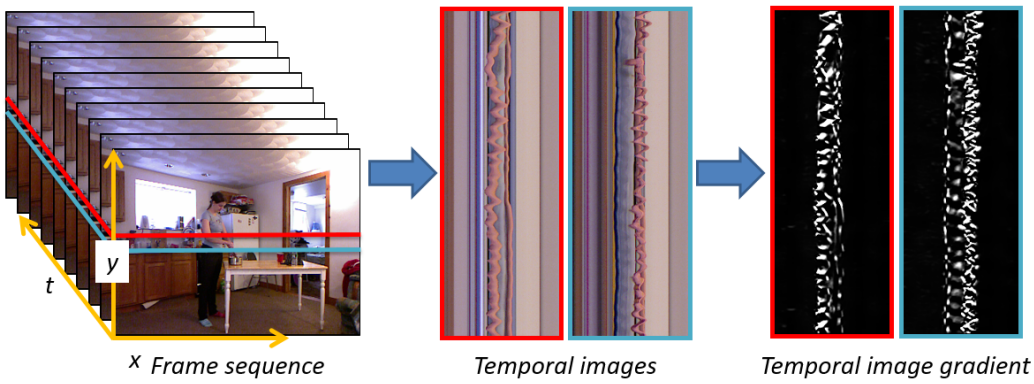


Figure 7.1: Representation of the feature extraction approach from RGB images. The temporal image (b) is a "slice" of the 3D volume representing the frame sequence (a). Relevant changes in time are well highlighted in the gradient image (c) extracted from (b).

The outputs of the two classifiers are then combined for the final result; among the existing combination strategies, the decision-level fusion is the most suited in this case due to the possible misalignment of the RGB and the skeleton streams which makes difficult a fusion at feature level. The two classifiers are equally weighted for the computation of the combined score, obtained by a simple sum rule. In our internal experiments, the typical fusion rules (max, sum, prod) have been evaluated. The sum rule provided the best results; in the next section, all the results concerning the score-fusion of the two classifiers use this rule. In the final part of the section, and in particular in Table 7.8, the results obtained with the different fusion rules will be compared and discussed.

7.2 Experiments and results

The proposed approach has been evaluated over three different public benchmarks, each of them including different sets of activities. In order to evaluate the effectiveness of the proposed approaches we had to focus on

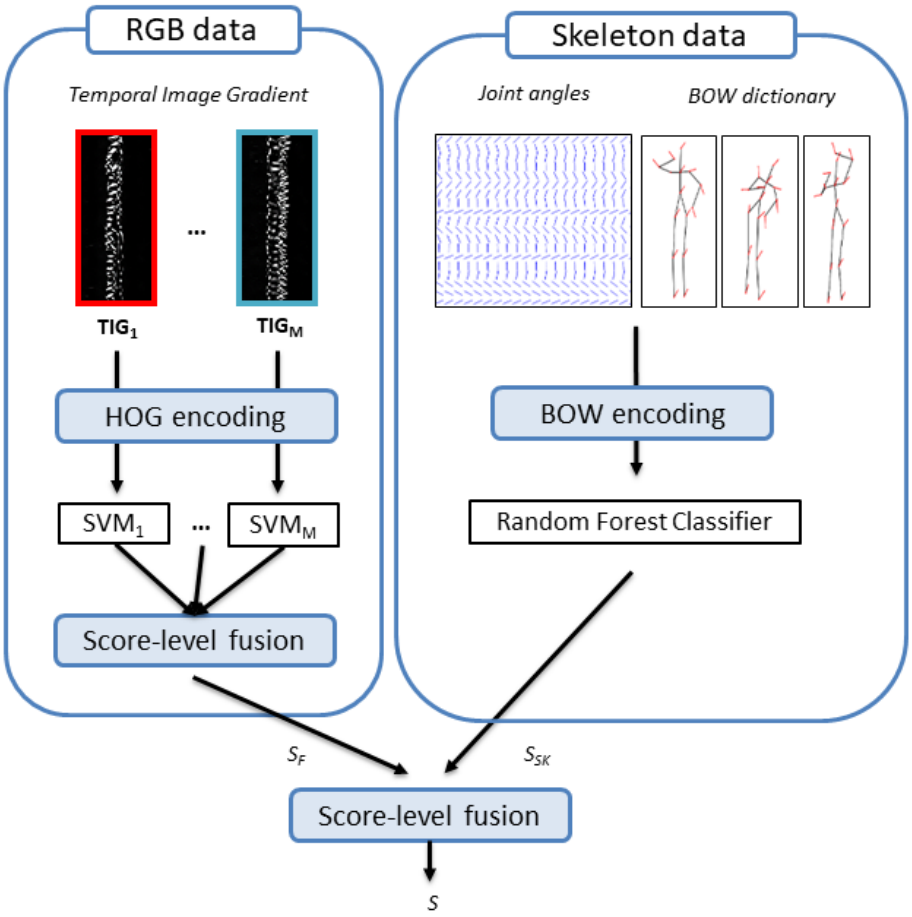


Figure 7.2: Schema of the proposed Multi-modal HAR approach. The final score is obtained by a score-level fusion of the output of the two modules based on RGB and Skeleton data respectively.

datasets providing both RGB frames as well as information about joint orientations. As underlined in Chapter 6, most of the existing HAR datasets provide only one of the two categories of data, so we finally selected two well-known public benchmarks, the already described CAD-60 and the more challenging CAD-120. We extended the experiments to an incremented version of the previously described OAD v.1.0.

7.2.1 Results on CAD-60

As mentioned in Section 7.1.2, we evaluated two different strategies for the selection of y-values, both based on skeleton information. In the first one, volume slices are extracted in correspondence of the position of the 15 joints describing the skeleton *(RGB - joint-based selection)*; the second one simply applies a uniform slice sampling along the whole skeleton *(RGB - uniform selection)*. A comparison between the two strategies on the CAD-60 dataset is given in Table 7.1 which also reports the results of other methods in the literature. Besides precision and recall, for each method an indication about the Kinect data exploited is given (*Sk*: skeleton, *RGB*: color frames, *De*: depth frames). It is worth noting that the skeleton information is derived by Kinect SDK from depth data, but we checked the *De* column only when the approach directly exploits depth images for feature extraction (different from skeleton).

The results show that the uniform sampling is more effective, probably because the initial join position in some cases (e.g. hands) is not significant. Moreover, the information related to specular joints (e.g., shoulders, pelvis, knees, elbows) is redundant and not informative, thus making us lean towards uniform sampling along the entire skeleton. The confusion matrix shown in Table 7.2, allows to analyse the results obtained with uniform sampling with 20 different slices.

Table 7.3 reports the confusion matrix obtained by the combination of RGB and skeletal representations; excellent results are observed, compared to existing approaches, both in terms of precision and recall.

7.2.2 Results on CAD-120

We have extended our evaluation to include the CAD-120. As the name suggests, CAD-120 consists of 120 videos of human actions where the subjects perform 10 high-level activities (*making cereal, taking medicine, stacking objects, unstacking objects, microwaving food, picking objects, cleaning objects,*

Table 7.1: Precision (P) and recall (R) of the proposed approaches on CAD-60, compared to the state-of-art results. For each method, the indication about the Kinect data exploited is also given: *Sk*: skeleton, *RGB*: color frames, *De*: depth frames."*" indicates that a different evaluation protocol was used.

Algorithm	*Sk*	*RGB*	*De*	*P*	*R*
(Sung et al., 2012, 2011)	✓	✓	✓	67.9	55.5
(Koppula et al., 2012)	✓	✓	✓	80.8	71.4
(Zhang and Tian, 2012)	✓			86.0	84.0
(Ni et al., 2012)	✓	✓	✓	Acc. 65.3	-
(Piyathilaka and Kodagoda, 2013)	✓			70.0*	78.0*
(Ni et al., 2013)		✓	✓	75.9	69.5
(Gupta et al., 2013)			✓	78.1	75.4
(Wang et al., 2014)	✓			Acc. 74.70	-
(Yang and Tian, 2014b)	✓		✓	71.9	66.6
(Zhu et al., 2014)	✓	✓	✓	93.2	84.6
(Faria et al., 2014)	✓			91.1	91.9
(Shan and Akella, 2014)	✓			93.8	94.5
(Gaglio et al., 2015)	✓			77.3	76.7
(Parisi et al., 2015)	✓		✓	91.9	90.2
(Cippitelli et al., 2016)	✓			93.9	93.5
(Urbano Nunes and Peixoto, 2017)	✓			81.83	80.02
Joint orientations approach	✓			**95.0**	**95.0**
(Qi et al., 2018)	✓			90.18	92.9
(Khaire et al., 2018)	✓	✓	✓	93.06	90.0
(Battistone and Petrosino, 2019)	✓	✓		94.4	93.7
RGB - joint based selection	✓	✓		**87.4**	**86.3**
RGB - uniform selection	✓	✓		**92.5**	**89.4**
Proposed multi-modal approach	✓	✓		**98.8**	**98.3**

taking food, arranging objects, having a meal). As for CAD-60, four subjects were considered, each of which performs each action three times. Several elements make CAD-120 a more challenging dataset: in particular, almost all activities exhibit relevant occlusions and the point of view varies depending on the actor. Besides, unlike the CAD-60, there is no distinctions between

Table 7.2: Confusion matrix of the RGB-based approach (using 20 uniform slices) on CAD-60.

	Talking on the phone	Writing on whiteboard	Drinking water	Rinsing mouth with water	Brushing teeth	Wearing contact lenses	Talking on couch	Relaxing on couch	Cooking (chopping)	Cooking (stirring)	Opening pill container	Working on computer
Talking on the phone	0.89		0.11									
Writing on whiteboard		1.0										
Drinking water			1.0									
Rinsing mouth with water				0.92	0.08							
Brushing teeth					1.0							
Wearing contact lenses						1.0						
Talking on couch							1.0					
Relaxing on couch							0.75	0.25				
Cooking (chopping)									0.75	0.25		
Cooking (stirring)									0.08	0.92		
Opening pill container											1.0	
Working on computer												1.0

rooms (i.e., all the actions take place in the same environment). As for CAD-60, even CAD-120 provides the 3D positions of 15 joints and the orientations of 11 of them. Different protocols are available for this benchmark; the most feasible for our evaluation is referred to as *Activity classification without ground-truth segmentation*[1].

The results on CAD-120 for the proposed approach are shown in Table 7.4 and 7.7. It is possible to observe in Table 7.7 that temporal images alone do not provide satisfactory results on CAD-120. This is probably due to the complexity of the dataset and in particular the frequent occlusion of subjects (typically through motionless objects that hinder the production of temporal images). The results obtained from skeletal information are consistently

[1]Cornell Activity Datasets: CAD-60 & CAD-120

Table 7.3: Confusion matrix using the score-level fusion approach on CAD-60.

	Talking on the phone	Writing on whiteboard	Drinking water	Rinsing mouth with water	Brushing teeth	Wearing contact lenses	Talking on couch	Relaxing on couch	Cooking (chopping)	Cooking (stirring)	Opening pill container	Working on computer
Talking on the phone	0.89		0.11									
Writing on whiteboard		1.0										
Drinking water			1.0									
Rinsing mouth with water				0.92	0.08							
Brushing teeth					1.0							
Wearing contact lenses						1.0						
Talking on couch							1.0					
Relaxing on couch								1.0				
Cooking (chopping)									1.0			
Cooking (stirring)										1.0		
Opening pill container											1.0	
Working on computer												1.0

better; however it clearly emerges that the two approaches are quite independent and their score-level fusion allows to significantly increase precision and recall (see Table 7.7). The confusion matrix describing the results obtained by merging the two techniques is shown in Table 7.5. Overall the results are encouraging, even if the method by (Koppula and Saxena, 2013) slightly outperforms our approach on this database. In our opinion, there are two main reasons for this behaviour. First, they perform a hierarchical analysis, identifying both high-level and low-level activities, and the information from low-level analysis can be very useful to improve recognition. Second, their graph-based representation explicitly models objects and interactions with objects, while in our approach these aspects are only indirectly represented by observing their effects of this interaction on the subject's movements in

RGB frames. The explicit knowledge about the objects in the scene allows to better deal with activities where the interaction with objects is a fundamental aspect (e.g., stacking or unstacking objects). Based on these considerations, we plan to explore possible improvements in our future research focusing on a better analysis of the context.

Table 7.4: Precision (P) and recall (R) of the proposed approaches on CAD-120, compared to the state-of-art results.

Algorithm	P	R
(Koppula et al., 2012)	81.8	80.0
(Koppula and Saxena, 2013)	**87.0**	82.7
Prop. appr. (RGB and skeleton fusion)	*85.4*	***83.3***

Table 7.5: Confusion matrix using the score-level fusion between the two classifiers on CAD-120.

	Arranging objects	Cleaning objects	Having meal	Making cereal	Microwaving food	Picking objects	Stacking objects	Taking food	Taking medicine	Unstacking objects
Arranging objects	0.83					0.17				
Cleaning objects		1.0								
Having meal			0.92			0.08				
Making cereal				1.0						
Microwaving food		0.08			0.83			0.08		
Picking objects						1.0				
Stacking objects				0.08			0.67			0.25
Taking food					0.08	0.08	0.08	0.67		0.08
Taking medicine				0.08			0.08		0.83	
Unstacking objects				0.25			0.17			0.58

Figure 7.3: Example frames of some of the activities carried out by the new 10 subjects in the OAD v.2.0.. Specifically: *stacking objects*, *taking objects from shelf*, *writing*, *drinking*, *getting up/sitting*, *grabbing an object from the ground*, *pouring a drink*, *scrolling book pages*, *talking on the phone* and *throwing something in the bin.*

7.2.3 Results on Office Activity Dataset v.2.0

Finally, the third dataset used for testing is the extended version of the OAD presented in the previous chapter. In order to offer the scientific community a more complex dataset with more significant variability, we have doubled the number of subjects (from 10 to 20) concerning the previous version. In addition, the 14 activities listed in the previous chapter are carried out in a totally different environment from several perspectives based on the action being performed. Some examples RGB frames are shown in Figure 7.3.

Of course, even in this extended version, each of the 14 actions is performed twice; hence, OAD v.2.0 includes a total of 560 video sequences. The skeletal data provided by the dataset are the same as in v.1.0 and are composed of the 3D positions of 25 tracked joints and the orientations of 19

Table 7.6: Confusion matrix using the score-level fusion between the two classifiers on OAD v.2.0.

	Drinking	Getting up	Grabbing obj.	Pouring a drink	Scrolling book	Sitting	Stacking items	Taking objects	Talking on phone	Throwing something	Waving	Wearing coat	Working on computer	Writing on paper
Drinking	0.88				0.03			0.02	0.07					
Getting up		0.88				0.10		0.02						
Grabbing obj.		0.06	0.82							0.12				
Pour a drink	0.09			0.88	0.03									
Scrolling book				0.03	0.97									
Sitting						1.0								
Stacking items							1.0							
Taking objects								1.0						
Talking on phone	0.05				0.02				0.90		0.03			
Throwing something			0.16		0.07		0.05			0.70	0.02			
Waving	0.20			0.05						0.03	0.72			
Wearing coat										0.07		0.93		
Working on computer													1.0	
Writing on paper							0.03							0.97

of them. RGB and depth images will be released with permission and in accordance with the General Data Protection Regulation (GDPR, EU no. 2016/679).

It is worth stress again that the execution of the different actions was loosely supervised, just giving to the subjects a generic definition of the action without specific indications on how it should be carried out. This results in a significant intra-class variability. From the confusion matrix shown in Table 7.6, it can be seen that the most critical activity remains *"throw something in the bin"*.

As underlined in the previous chapter, this action has been interpreted by the volunteers with great fantasy, in very different fashions (see Figure 7.4 for two samples of this activity). Many of them interpreted this particular activity as a sequence comprising the approach to bin, bending down and finally the release of the object. Others preferred a literal interpretation

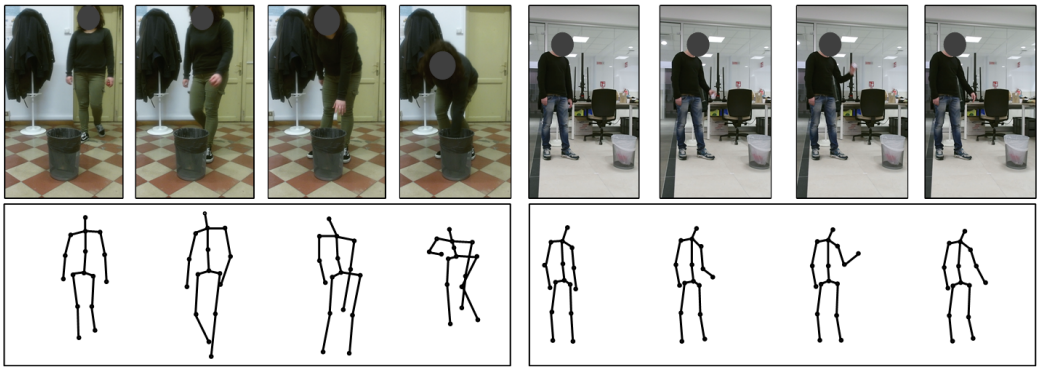


Figure 7.4: Sample frames (RGB and skeleton) for the *"throw something in bin"* action.

of the label name and performed the action by throwing the object from a distance. This explains some of the errors due to the misclassification with *"grab object from the ground"*.

The misclassification of *"waving"* and *"drinking"* is purely due to an intrinsic inter-class variation, mainly due to the similar configuration of a significant portion of the angles between these two activities. Despite of some errors in specific activities, the good behaviour of the proposed approach is confirmed in this test as well.

Finally, as mentioned in Section 7.1.3, we evaluated the typical fusion rules (max, sum, prod); the results over the three datasets are given in the Table 7.8. Overall, the common sum rule provides better results, probably because the two approaches are quite complementary and their sum results in a more robust estimation. The max rule provides the worst results meaning that, in some cases, one of the two methods provides the wrong class with a high confidence value and this problem is amplified by the max selection rule.

Table 7.7: Summary of the performance obtained on the three testing datasets.

Dataset	Approach	Precision	Recall
CAD-60	Skeleton	95.0	95.0
	RGB (20 sectors)	92.5	89.4
	Score-level fusion	98.8	98.3
CAD-120	Skeleton	77.6	73.1
	RGB (20 sectors)	61.1	59.3
	Score-level fusion	85.4	83.3
OAD v.2.0	Skeleton	80.6	80.5
	RGB (20 sectors)	85.8	85.9
	Score-level fusion	90.6	90.4

Table 7.8: Results obtained over the three datasets with different fusion rules.

Dataset	Fusion rule	Precision	Recall
CAD-60	Max	97.1	96.9
	Sum	**98.8**	**98.3**
	Prod	97.2	96.3
CAD-120	Max	78.0	74.6
	Sum	**85.4**	**83.3**
	Prod	82.4	80.2
OAD v.2.0	Max	84.1	83.6
	Sum	90.6	90.4
	Prod	**90.7**	**90.5**

7.3 Final Remarks

In this chapter, Human action recognition has been addressed by a multimodal approach based on the combination of skeletal information and tem-

poral images encoding obtained from RGB frames. Of course, combining different modalities increments the computational effort, in particular when dealing with RGB images. The cost of processing skeleton information is, in fact, negligible, i.e. a few milliseconds to process a whole activity; processing RGB frames for gradient, denoising and HOG features extraction is quite expensive, but overall the system is able to operate in real time since the recognition of a sequence (including RGB and skeleton data processing and their fusion) requires about 0.5 seconds using non optimised Python and C# code on an Intel Core i7-2600. We believe that the increment of computational effort is fully justified by the considerable improvement in recognition accuracy, in particular on the most difficult datasets. Of course, the deployment of this approach on embedded systems with reduced computational resources would require ad-hoc optimisations. The results on public benchmarks confirm the complementarity of the two information, leading to a significant improvement of classification performance with respect to the single techniques.

Chapter 8

Template co-updating in multi-modal HAR systems

In the specific context of the home environment, the acquisition of a large amount of training data is quite difficult and unlikely. The home environment is usually characterised by a very limited number of users, and also most of the reference benchmarks for activity recognition reproduce a "small-size" scenario, with few users and few activity samples per user. We are confident that in this scenario "traditional" computer vision techniques can achieve good results and real time processing capabilities even with limited computational power, while techniques based on deep learning are more difficult to apply. On the other hand, if we think at an home environment where the hypothetical monitoring system is continuously checking the ambience to detect possible anomalies and/or to understand human actions, it is clear that a huge amount of unlabelled data can be easily collected. On the contrary, labelled training data are often scarce, with a few video samples for each activity to be recognised; this is in our opinion the realistic scenario where huge amount of training data is very unlikely to be available. We believe therefore that the implementation of incremental updating techniques is mandatory in advanced recognition systems to fully exploit the richness of data that the specific scenario naturally provides. Moreover, as seen in

the Chapter 7, many works in the literature are multi-modal in nature.The template updating procedure could thus rely in many cases on different data sources whose combined use can in principle improve the effectiveness of the updating procedure, reducing at the same time the probability of selecting wrong data that could deteriorate the initial templates. Finding a good trade off between the need of adding new information to the initial templates and avoiding updating errors that could compromise them is, in fact, the main challenge in this problem.
In this chapter, we propose an incremental co-updating technique, based on the joint analysis of RGB images and human skeleton information acquired with the Kinect sensor. The proposed approach is semi-supervised, i.e. we suppose to have a small initial training set for the creation of the base templates which are subsequently updated in a totally unsupervised way. To the best of our knowledge, only a few works in the literature propose template updating techniques for human action recognition (see next section) based exclusively on RGB images while the possibility of co-updating based on multiple data sources has not been investigated so far.
The rest of the chapter is organised as follows: in Section 8.1 a specific discussion of the template updating techniques proposed for human action recognition is reported, Section 8.2 describes the proposed co-updating approach for a generic multi-modal system and presents a specific implementation based on RGB images and skeleton data, the experimental results are reported and discussed in Section 8.3 and finally Section 8.4 draws some concluding remarks and provides ideas for future research.

8.1 Related works

Most of the existing HAR approaches focus on static activity models, where all the training samples are supposed to be labelled and available at the time the model is first computed.
Only a few works address the problem of dynamic template updating. In

(Reddy et al., 2009) an incremental approach based on a feature tree is proposed; the feature tree grows as new data become available, but this requires maintaining all the training and updating samples and this aspect limits its practical applicability. (Minhas et al., 2012) describe an updating technique based on human tracking in video sequences. In this case a manual annotation of the human body is needed at the beginning of the action clip, so all the updating process is supervised to some extent and unfeasible for our purposes. An active learning technique based on the idea of adding new weak classifiers for new incoming instances is presented in (Hasan and Roy-Chowdhury, 2014). The whole method is based on STIP features (Laptev, 2005) extracted from RGB images. (Rosa et al., 2017) propose a general framework for active incremental recognition of human activities where the feature space used to represent information is gradually covered with balls centred on samples selected from the stream. Finally, (Hasan and Roy-Chowdhury, 2015) propose a framework for continuous learning based on deep hybrid feature models. In particular, the approach is aimed at automatically learning the optimal feature models for activity recognition exploiting a deep auto-encoder, and at continuously updating the templates; to this last purpose, a selection criteria is defined to identify, among the accumulated samples, the best subset for updating. Some of the approaches in the literature are very interesting and exhibit promising performance however, to the best of our knowledge, all of them focus on a single data source (RGB images in most cases).

Multiple data sources have been successfully exploited in other contexts, in particular for multi-modal biometric systems (Roli et al., 2007); the idea is that systems based on different characteristics provide complementary performance, since each recogniser is expected to assign correct labels to certain input data which are difficult for the other and vice-versa. We will explore the applicability of this principle to the problem of action recognition where different data sources (e.g. RGB, skeleton, depth data), possibly independent, are likely to be available when common acquisition devices such as Kinect are used.

8.2 Proposed approach

The aim of this work is to propose a general framework for template co-updating based on the analysis of multiple data sources. The algorithm will be described in the next section without any assumption about the specific features used; then a possible implementation based on the combination of information from RGB images and human skeleton will be described and used for the experimental validation of our proposal.

8.2.1 The general template co-updating algorithm

The template co-updating procedure exploits n types of information derived from different data sources. For each source, a specific classifier $Cl^k, k = 1, .., n$ is pre-trained on a set of a activities; the resulting templates are: $T^k = \{T_1^k, ..., T_a^k\}$. The basic idea of our co-updating algorithm is that when the prediction of an input sequence operated by a specific classifier (at least one) fulfils a series of reliability criteria, then that sequence will be used to update all classifiers (of course each with the specific data source). This way we think that it will be possible to accept for the other classifiers also data rather far from the existing templates thus increasing their representativeness. With the aim of determining the robustness and reliability of a prediction, we considered the following criteria.

8.2.1.1 Reliability of each classifier

A classifier may exhibit poor reliability in predicting specific classes of activities, and provide very good results on others. The reasons may be different, including the possible and repeated occlusion of body parts or the poor representativeness of the templates used. This can obviously lead to rapid degeneration of the quality of predictions concerning a specific classifier. Such a phenomenon becomes critical in a co-updating context, where, if not adequately addressed, one classifier may corrupt the templates of others. Based on the reliability of the various data sources in relation to the initial

Algorithm 1: Template Co-Updating

Initialize variables;
foreach *new sequence* s_j **do**
  $\mathsf{F} \leftarrow$ `ExtractFeatures(`s_j`)`;
  $\mathsf{y} \leftarrow$ `AssignClassLabel(`F`)`;
  if $\mathsf{y} = -1$ **then** `// sequence not labeled`
    $U^f \leftarrow U^f \cup f \; \forall f \in F$;
  else `// sequence labeled`
    $B^f \leftarrow B^f \cup f \; \forall f \in F$;
    $Y^f \leftarrow B^f \cup \mathsf{y} \; \forall f \in F$;
    $newTemplate \leftarrow 1$
  end
  if $B^f \geq$ bufferMax **then**
    $B^f \leftarrow$ `UpdateBuffer(`B^f,Y^f`)` $\forall f \in F$;
  end
  if $newTemplate = 1$ **then** `// classifiers update`
    $Cl^f \leftarrow$ `UpdateClassifier(`B^f,Y^f`)`$\forall f \in F$;
    Retry recognizing all unlabeled elements $\in U$ and update classifiers if needed;
    $newTemplate \leftarrow 0$
  end
end

templates, our approach assigns to each classifier an activity-specif weight, proportional to its ability to correctly identify activity sequences of each class. To pursue this goal, we determine, for each classifier Cl^k, a set of weights $w_i^k = \{w_1^k, ..., w_a^k\}$ for the different activities based on the classifier *precision* (as defined in Section 8.3). Among the others, we adopted this metric because it minimises the likelihood of accepting false positives for a specific class, a fundamental characteristic in this scenario, where accepting a false positive is certainly the most critical type of error.

8.2.1.2 Degree of certainty of a prediction

The decision of exploiting an incoming sequence s_j for template updating relies on the robustness of its classification by the different classifiers. To this end, let's suppose that the classifier Cl^k produces a distribution of probabilities $\mathbf{p}^k = [p_1^k, .., p_a^k]$ for the sequence s_j over the a activity classes, and let's suppose that the two most probable classes are, respectively, c_1^k and c_2^k. We define the *degree of certainty* of the prediction (s_j belongs to class c_1^k) such as:

$$doc(c_1^k) = 1 - \frac{\mathbf{p}^k[c_2^k]}{\mathbf{p}^k[c_1^k]}$$

where $p^k[c_2^k]$ and $p^k[c_1^k]$ are the two highest probability estimates offered by the classifier. The rationale behind this choice is that when the two highest probabilities are both high and similar, the prediction is very uncertain. Such a simple metric can be particularly useful in cases of strong similarities between classes (*e.g., drink/talking on phone*) and allows to exclude potential risky updates that could affect the robustness of the approach. The degree of certainty of the prediction is then weighed for the previously determined set of weights and define the *credibility* as:

$$cre(c_1^k) = doc(c_1^k) * w_{c_1^k}^k$$

8.2.1.3 Restrictive sequence acceptance rules

In order to define the criteria of acceptance of a new sequence, several aspects need to be taken into account. The first distinction is whether or not all classifiers agree on the predicted class (line 8 Algorithm 2). In the first case, to avoid a common misclassification, the algorithm exploits two different parameters of acceptability: δ_{cre} defines a threshold for the credibility (0.35 in our experiments) while δ_{close} defines a closeness threshold (0.2). On the one hand, δ_{cre} certifies that all predictions are considered robust, on the other, δ_{close} defines a common closeness of the consensus. Indeed, a classifier may have a degree of credibility higher than the relative threshold, but distant

Algorithm 2: AssignClassLabel(F)

input : $\mathsf{F} = \{f^1, ..., f^n\}$

output: A predicted class or -1

1 *Initialize variables*;
2 **foreach** *feature channel* f^k *in* F **do**
3 $\quad \mathbf{p}^k \leftarrow \texttt{ProbabilisticPrediction}(f^k)$;
4 $\quad c_1^k,\, c_2^k \leftarrow \texttt{MostProbableClasses}(\mathbf{p}^k)$;
5 $\quad doc(c_1^k) \leftarrow 1 - \mathbf{p}^k[c_2^k] \setminus \mathbf{p}^k[c_1^k]$;
6 $\quad cre(c_1^k) \leftarrow doc(c_1^k) * w^k_{c_1^k}$;
7 **end**
8 **if** $(c_1^i = c_1^j = c\, \forall (Cl^i, Cl^j)) \wedge (cre(c_1^i) \geq \delta_{cre}\, \forall i) \wedge$
$(|cre(c_1^i) - cre(c_1^j)| < \delta_{close} \forall\, (Cl^i, Cl^j))$
then return c ;
9 **if** $\exists\, Cl^k |\, (cre(c_1^k) \geq \delta_{cre}) \wedge$
$(|cre(c_1^k) - cre(c_1^i)| \geq \delta_{diff}\, \forall i)$ **then**
10 $\quad$ **return** c_1^k
11 **else**
12 $\quad$ **return** -1 ;
13 **end**

from the others; this implies a partial dissent in the common choice. Of course, both the thresholds could be weighed by the number of classifiers, relaxing these constraints. On the other hand, if classifiers do not agree, the prediction with the highest credibility (which must necessarily be higher than δ_{cre}) will be considered (line 9 Algorithm 2). In fact, the degree of credibility must be sufficiently higher than the one presented by others. For this purpose, δ_{diff}, which parameterises the supremacy of one prediction over the others, is exploited. For our experiments we fixed this parameter to 0.2. These constraints are defined in Algorithm 2. If the algorithm is able to assign a reliable class label to the new sequence observed, according to

the rules described above, the new sample will be included in the buffer of labelled samples B and used for template updating (for all the classifiers). Otherwise, the sample will be queued to the buffers of unlabelled samples U. After each incremental update of the classifier, our framework will attempt to re-assign a label to the queued unlabelled samples (line 17 Algorithm 1); after the updating, in fact, they could be better recognised.

8.2.1.4 Template preservation

Finally, it is realistic to assume that buffers are limited, so that when they reach their maximum capacity, some samples have to be removed. The strategy adopted is to preserve the most expressive samples for a given class. Therefore, even if it may seem counter-intuitive, the underlying assumption is to search for those buffered samples for which the classifiers show the highest uncertainty. Indeed, if the templates with the highest degree of confidence were preserved, it would be legitimate to assume that the classifier is reducing the expressiveness of a specific activity representation; the most useful templates could potentially be excluded from an adequate representation of intra-class variations. Clearly, the search policy is based on the class of the element that is causing the buffer overflow. In addition, to avoid over-representation of a specific class, the places in the buffer are evenly distributed among the different classes; therefore, adding a template is bound to an additional parameter that defines the maximum number of buffer elements for a given class. The maximum buffer size has been fixed in our tests to 170, and the buffer is initialised with the samples of the training set, in order to avoid any forgetting effect.

8.2.2 An implementation based on RGB and skeleton data

In Chapter 7, is presented an approach based on temporal images defined through the RGB information. This approach, however, uses a set of SVM

Figure 8.1: Improved Dense Trajectories features in a frame of the *drink* action. Red dots describe the position of interest points in the current frame. In green are represented the IDTs over $L = 8$ frames.

classifiers, whose cardinality is equal to the number of sectors exploited. In evaluating the template co-updating approach, we initially prefer to adopt a single classifier solution, reserving the evaluation of other multi-classifier-based approaches for future development.

Among the possible alternatives to represent RGB information, we adopted Improved Dense Trajectories (IDTs) (Wang and Schmid, 2013), well-known for their excellent performance in action recognition tasks. These trajectories are a composition of different features extracted from each video sequence, specifically: HOG, HOF and Motion Boundary Histogram (MBH). The features have been extracted from each video using the code published on the INRIA website[1]. Similar to (Rosa et al., 2017), we kept the default parameters and only reduced the length L of the trajectory (from 15 to 8 frames). An example of some IDTs extracted from the OAD dataset, is show in Figure 8.1.

[1]IDTs documentation – INRIA website

The extracted trajectories are accumulated and each of the four features used (HOG, HOF, MBHx and MBHy) has been encoded using a BoW model (Wang et al., 2009) with $K = 500$. Therefore, each video is described by a histogram of 2000 elements obtained by concatenating the individual descriptors. The classifier adopted is a linear incremental SVM with stochastic gradient descent (SGD) learning[2].

As for the Chapters 6 and 7, we adopted the joint orientations representation. The substantial difference concerning previous proposals is the adoption of a different classifier instead of the aforementioned Random Forest. Indeed, the same classifier used for RGB information has been used here as well.

8.3 Experiments

The proposed approach has been validated with extensive experiments where the proposed co-updating technique is compared with a batch template creation and a fully supervised incremental updating.

8.3.1 Database and protocol

The independence of the data sources used for co-updating is a key factor for the effectiveness of the process. The size of the two previously described CAD datasets is unfortunately very limited, with data taken from only 4 subjects, and we think that a validation on this data would not be so meaningful. For this reason we performed all the experiments on the OAD v.2.0. For testing the proposed updating technique, we followed the common "new person" protocol, meaning that disjoint subjects are used for training, updating and testing. In particular, the available subjects are randomly partitioned as follows:*training set* (TR) 20% (4 subjects, 112 sequences), *updating set* (UPD) 50% (10 subjects, 280 sequences), *testing set* (TE) 30% (6 subjects, 168 se-

[2]Scikit-Learn SGD Classifier

Exp. Setup ID	TR	UPD	TEST
Set_1	{0,1,10,11}	{2,3,4,5,6,12,13,14,15,16}	{7,8,9,17,18,19}
Set_2	{2,3,12,13}	{4,5,6,8,9,14,15,16,18,19}	{0,1,7,10,11,17}
Set_3	{7,8,17,18}	{2,3,4,5,6,12,13,14,15,16}	{0,1,9,10,11,19}
Set_4	{4,9,14,19}	{0,1,6,7,8,10,11,16,17,18}	{2,3,5,12,13,15}
Set_5	{5,6,15,16}	{0,1,7,8,9,10,11,17,18,19}	{2,3,4,12,13,14}
Set_6	{0,7,10,17}	{1,2,3,8,9,11,12,13,18,19}	{4,5,6,14,15,16}
Set_7	{1,9,11,19}	{2,3,4,5,7,12,13,14,15,17}	{0,6,8,10,16,18}

Table 8.1: Partition configurations used to validate the co-updating approach on OAD v.2.0.

quences). The experiments are repeated seven times with different subject partitions (see Table 8.1) and the average results are finally computed.

The performance is reported in the form of overall recognition accuracy (percentage of activities correctly classified), precision and recall values. For evaluation purposes, the unsupervised co-updating approach is compared to two supervised approaches. In particular, the performance are reported for:

- *Proposed co-updating* (**Co-Updating**): the initial templates are created from the subjects in the train set TR and subsequently incrementally updated with the set UPD based on the approach proposed in this work (see Section 8.2).

- *Supervised template updating* (**Supervised Updating**): the initial templates are created from the subjects in TS, and then are updated sequentially using the subjects in UPD exploiting the real activity labels; all the incoming information is used in this case correctly, so this approach gives an upper bound to the performance that can be achieved with an incremental learning strategy.

- *Batch template creation* (**Batch**): all the subjects of training and updating sets ($TR \cup UPD$) are exploited for the initial template creation;

this approach gives an idea of the top-performance that could be obtained if a huge amount of training data were available at the time of initial template creation. The system is static, no updating is carried out in this case.

In all cases, the performance are measured on the testing subjects in TS.

8.3.2 Results

Table 8.3 summarises the results in terms of precision and recall measured for the proposed approach, as well as for the two supervised techniques taken as reference systems. Of course the supervised approaches provide an upper bound to the performance that can be achieved; in particular the batch approach represents the most favourable case where a huge amount of data is available from the beginning. The performance indicators are given for the separate modalities (RGB and skeleton) as well as for their fusion. The results measured with the initial template are compared to those obtained after updating to appreciate the effects of template updating.

RGB features seem to be generally more stable and reliable than skeleton data, even if the difference is overall quite limited. The independence between the two modalities is very useful during co-updating; in fact, new incoming samples quite far from the existing template for a given feature can often be accepted due to the high confidence of the other; this is indeed the ultimate objective of template updating: adding new, different samples to the template to increase its representativeness. The accuracy trend during the different updating steps in one of the different runs of experiments is given in Figure 8.2. The trend is positive for the single modalities and, as expected, also for their fusion. The skeleton templates greatly benefit from the updating procedure, thanks to the support of RGB templates.

When the performance of the unsupervised approach are compared to the batch of the supervised one, we can observe very similar values of precision

Table 8.2: Confusion matrix using only the training set (i.e., before the application of the template co-updating algorithm).

	Drinking	Getting up	Grabbing obj.	Pouring a drink	Scrolling book	Sitting	Stacking items	Taking objects	Talking on phone	Throwing something	Waving	Wearing coat	Working on computer	Writing on paper
Drinking	0.68			0.10				0.03	0.12			0.07		
Getting up		0.64	0.06			0.26	0.01			0.01				0.01
Grabbing obj.			0.81			0.08	0.01	0.01		0.07		0.01		
Pour a drink	0.03			0.78	0.14							0.04		0.01
Scrolling book				0.10	0.88				0.01			0.01		
Sitting		0.07	0.01			0.92								
Stacking items		0.01					0.97	0.01						
Taking objects							0.01	0.97	0.01					
Talking on phone	0.03		0.03	0.01					0.69	0.03	0.06	0.15		
Throwing something	0.01	0.04	0.11	0.03		0.06	0.04	0.04	0.07	0.43	0.07	0.07		
Waving	0.12		0.03	0.04	0.03				0.08	0.04	0.49	0.17		
Wearing coat											0.01	0.99		
Working on computer												0.01	0.99	
Writing on paper								0.01					0.10	0.89

and recall. This very positive result confirms that the unsupervised approach exploits at best the updating set, i.e. accepts a high number of unknown sequences and correctly uses them for template updating.

The effects of our co-updating procedure can be better analysed looking at the confusion matrices of Figure 8.2 and Figure 8.4 referred, respectively, to the initial templates and to the final templates obtained with unsupervised updating. Some activities were already recognised with a good level of accuracy even with the initial templates (e.g. stack items, take objects from shelf, wear coat, work on computer) and such performance is preserved by the updating procedure; we can conclude that wrong updates are very rare, totally absent for most activities, and the initial templates are not corrupted. For other activities, probably characterised by a higher degree of variability between different subjects, the initial templates provide poor performance; template updating greatly improves the results in several cases (e.g. drink,

		Initial templates		**Final templates**	
		Precision	*Recall*	*Precision*	*Recall*
Skeleton	Co-Upd.	0.701	0.697	0.758	0.749
	Sup. Upd.	0.701	0.697	0.767	0.752
	Batch	0.785	0.776	//	//
RGB	Co-Upd.	0.798	0.788	0.887	0.872
	Sup. Upd.	0.798	0.788	0.907	0.891
	Batch	0.925	0.917	//	//
Fusion	Co-Upd.	0.799	0.794	0.893	0.887
	Sup. Upd.	0.799	0.794	0.923	0.922
	Batch	0.944	0.943	//	//

Table 8.3: Comparison between the proposed co-updating procedure, the supervised updating (where real activity labels are exploited) and the batch updating (where all the training samples are available for initial template creation)

.

get up, waving or talking on phone). Particularly interesting is the improvement observed for "get up", very similar from the point of view of body posture to "sit". Several of the surviving errors are quite comprehensible if we consider that the mistaken activities often share common body positions and movements. Even for this evaluation, the most particular case is represented by the "throw something in bin" action. The already mentioned intra-class variability is the reason why the performance increment is limited in this case, and the final accuracy is still sub-optimal.

8.4 Final Remarks

In this chapter a general framework for template co-updating in multi-modal activity recognition systems has been proposed. The validity of the

Table 8.4: Confusion matrix after the application of the template co-updating algorithm.

	Drinking	Getting up	Grabbing obj.	Pouring a drink	Scrolling book	Sitting	Stacking items	Taking objects	Talking on phone	Throwing something	Waving	Wearing coat	Working on computer	Writing on paper
Drinking	0.88			0.07				0.03	0.01			0.01		
Getting up		0.89				0.10				0.01				
Grabbing obj.			0.88			0.04	0.03	0.04						
Pour a drink	0.04			0.81	0.14							0.01		
Scrolling book				0.03	0.97									
Sitting		0.08				0.92								
Stacking items							0.99	0.01						
Taking objects								0.99	0.01					
Talking on phone	0.08			0.03	0.04				0.82			0.03		
Throwing something	0.07	0.06	0.01	0.06		0.01	0.10	0.06		0.54	0.01	0.03		0.06
Waving	0.03				0.04				0.08		0.79	0.06		
Wearing coat												1.0		
Working on computer													1.0	
Writing on paper					0.01								0.03	0.96

proposal has been assessed with a specific implementation based on RGB and skeleton data extracted from video sequences acquired with the Kinect sensor. The results show that jointly exploiting different data modalities allows to greatly improve the initial performance, thanks to the inclusion of new data, previously not adequately represented by the initial templates.
The results obtained are fully satisfactory, however several further improvements are possible: for instance the weights used in the algorithm for the different classifiers are now static while a dynamic updating could better exploit the effects of template updating. The main extension of the approach will be the definition of a strategy for discovering new activity classes and including them in the set of known behaviours.

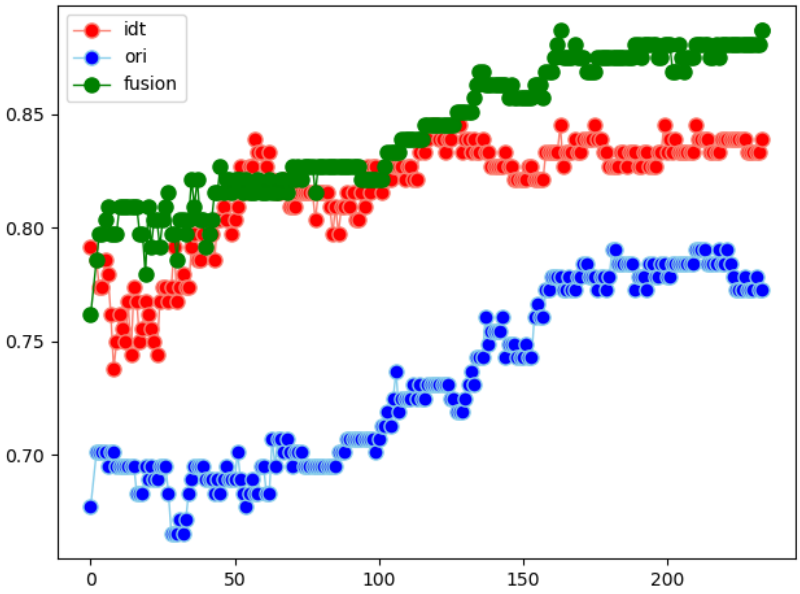


Figure 8.2: Activity recognition accuracy trend during unsupervised template co-updating (as a function of the number of updates performed) for the Set_1 configuration. The three curves represent the RGB templates (red), the skeleton templates (blue) and their fusion (green).

Part III

A Monitoring Framework for IoT

Chapter 9

Related works

The goal of this part of the thesis is to discuss some of the most promising IoT platforms while proposing a completely home made solution relying on open source technologies. This approach allows us to discuss design and implementation details at each layer of the stack our platform is built upon, enabling researchers and practitioners to fully understand what lies behind an IoT solution. Other academic institutions have felt the need to propose IoT platforms that could offer an under-the-hood view, for example (Castellani et al., 2010) have proposed a solution explicitly focused on indoor environments. Conversely, our proposal is designed for both outdoor – as sensor networks distributed in urban and suburban areas – and indoor environments. Therefore, particular importance will be posed on the integration and interoperability between the different networks. In this context, we want to offer the possibility to monitor information from the various sensors currently considered (see Section 10.1.1). As proposed in (Latré et al., 2016; Chan et al., 2018), we want to offer an IoT testbed that is useful for both academic teaching and research activities. Above all, we want to propose a platform that could be adopted in two potential scenarios: *i)* where a citizen can act as an active component, for example by adding one or more nodes to the network in an agile way and possibly monitoring specific areas of interest (Gubbi et al., 2013); *ii)* where it is possible to remotely monitor multiple

smart homes or AAL environments through a dedicated client application.

9.1 IoT Commercial Platforms Comparison

Since the term IoT was coined in 1999 by Kevin Ashton during a presentation at Procter & Gamble (Ashton, 2009), the basic idea behind IoT solutions has been widely explored by both the academic world and the ICT community. The IoT domain can be intuitively discussed as follows: let us consider a number of distributed sensors or gadgets (i.e., *"things"*) lying in an unpredictable vast environment (a house, a large urban area or a greater region). These things are able to gather a massive amount of raw data and translate them into relevant information. This ecosystem could reacts proactively, minimising (or at least trying to minimise) human involvement, complementing the AmI vision.

Although straightforward this scenario may appear, it hides a number of open questions. Which kind of architecture should be adopted? Which requirements are the most meaningful among others? Which communication standards should be adopted in order to enable device interoperability? What kind of API should be implemented to easily allow a sensor (or a sensor network as a whole) to join the ecosystem?

In (Guth et al., 2016), the authors propose an interesting comparison aimed at highlighting common architectural aspects of several IoT platforms and infer a reference architecture. Conversely, a comprehensive description and comparison of the main requirements (both functional and non-functional) of a IoT platform is discussed in (Razzaque et al., 2016).

Many platforms and solutions were proposed within the IoT domain. Each of them was designed with a business model in mind and thus holds specific features: in this thesis we adopt the taxonomy proposed in (da Cruz et al., 2018), where IoT platforms are discussed in relation of the corresponding application area.

Device Management Platforms, as defined by the Open Mobile Al-

liance Device Management, must guarantee the provisioning and onboarding of the devices, including remote parameterisation and real-time configuration. Again, they should allow remote firmware updates as well as a real time monitoring concerning devices faults and errors (Open Mobile Alliance, 2012). Therefore, these platforms enable a quick deployment of individual or entire groups of devices, and allow to define taxonomies and hierarchies upon them. They also allow to define access policies for different types of devices. One of the key aspects this kind of solution tend stress is the optimization of network resources. Device Management Platforms are becoming increasingly important and have consequently drawned the attention of many companies, such as Amazon, which at the end of 2017 has released the new *Amazon IoT Device Management* platform[1].

Application Development Platforms are aimed at fastening the implementation process of ICT services addressing the IoT domain. End-user applications are developed through automatic code generation and combined with a number of predefined API. One of the best-known toolkits is Temboo[2], which allows parametrisation, events management and automatic code generation for a number of heterogeneous devices.

Application Enablement Platforms, as the name suggests, allow IoT architectures to integrate with pre-existing external services and applications. Therefore, these solutions operate between the hardware layer consisting of sensors and actuators, and the end-user application layer. They often act as an integration middleware: devices communicate directly with the platform through transport protocols such as HTTP/S or MQTT and encapsulate data using classical data-exchange formats (XML, JSON). The integration middleware rearrange this information and delivers it to end-user applications.

The solution we discuss in the following falls into the latter category. For ease of reading, we point out that Application Enablement Platforms are

[1]AWS IoT Device Management website

[2]Temboo website

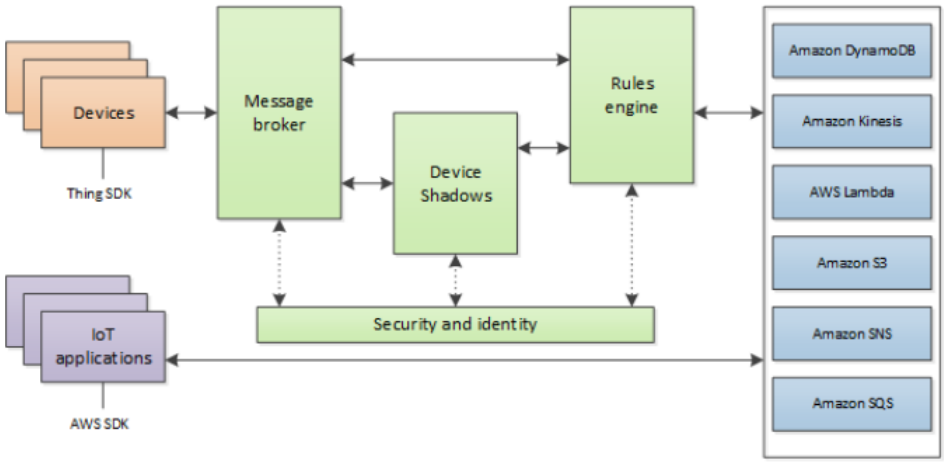


Figure 9.1: AWS IoT Core architecture.

often referred to as *IoT middleware*, *middleware* or *IoT middleware platform*: we use these definitions interchangeably. Before introducing our solution, we discuss some of the most prominent platforms belonging to this latter category.

9.1.1 Amazon Web Services IoT Core

Amazon Web Services (AWS) IoT Core is the middleware proposed by Amazon. It consists in a cloud solution relying on a Platform as a Service (PaaS) business model. Scalability and interoperability are the most relevant features of this solution: Amazon ensures that a single IoT Core instance can support billions of devices, allowing the exchange of tens of billions of messages between AWS endpoints. The main role of AWS IoT Core is therefore to provide a reliable connection between "things" and the AWS cloud. In order to achieve this, the well-known HTTP, MQTT and WebSockets protocols are used and all communications are secured through TLS and X.509 certificates.

The platform architecture (see Figure 9.1) consists of four leading modules (*message broker, device shadows, rules engine, security and identity*) plus a fifth component (the *device gateway*) which is not represented in figure. This

latter module connects devices to the message broker. Specifically, it exposes an incoming interface implementing the aforementioned protocols and acts as an intermediary to the message broker. The *message broker* is a publish/-subscribe service that allows all devices to receive or send messages related to a specific topic they have previously registered to (e.g., Sensor/RGBD-Sensor/LivingRoom). A device communicates its own status to the platform publishing a message under a proper topic. The *device shadow* service enable virtualisation and persistence of each device in the cloud, allowing maintenance of the last known device state even when it is no longer online. When an object is properly connected, the status of its shadow can be updated consistently with respect to the physical device. Conversely, when the communication fails, it is still possible to interact with the device relying on its shadow. The *rules engine* module implements the business logic of the platform, making it possible to collect and process raw data. As the name suggests, the user is allowed to define rules that orchestrate the distribution of messages among other objects or AWS services. Finally, all these components interact with the *security and identity* module which is responsible of providing reciprocal authentication and encryption at all communication levels of the stack. Therefore, a two-way communication without identity assessment will never occur. This middleware holds all the benefits provided by Amazon Web Services, but, as predictable, several implementation details remain unknown.

9.1.2 Microsoft Azure IoT Suite

Azure IoT Suite is the cloud platform developed by Microsoft. As per the AWS IoT Core, the business model is PaaS. One of the main advantages offered by this platform is the ability for users to install preconfigured solutions designed to fit common IoT scenarios. These solutions are released for free. As an example, Azure IoT Suite is equipped with a weather forecasts setting which enables data collection as well as information transmission to the middleware and its analysis through the Azure Machine Learning mod-

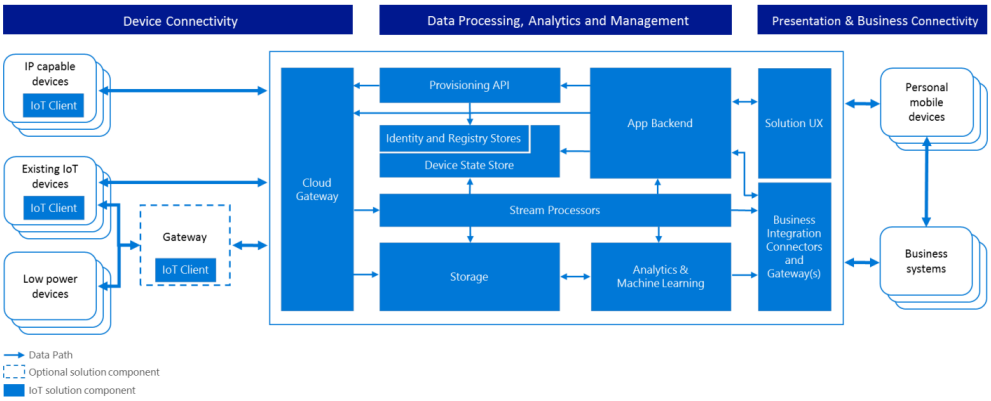


Figure 9.2: Microsoft Azure IoT reference architecture.

ule. Each of these preconfigured solutions involves different devices and rely on several modules among those offered as a service by Azure and Azure IoT Hub, which are indeed the real middleware. Figure 9.2 shows the reference architecture of an IoT system according to Microsoft's vision: within the blue rectangle it is represented an ensemble of cloud components needed to support an IoT solution. Azure IoT Hub plays the leading role as *Cloud Gateway* technology.

Azure IoT Hub enables connection between millions of devices and a cloud based back-end, supporting bi-directional communication for AMQP, MQTT and HTTPS protocols. Features of this hub include *twin devices*, a similar solution to the AWS Device Shadow. A *twin device* consists in a JSON document in which information concerning the status of the paired device is stored. For each connected device, Azure maintains a twin whose information can be used by the device itself or by other applications, in order to perform device configurations or to query it for data. This feature is very helpful for batch operations. Regarding communication security, Azure Hub IoT grant access to each hub endpoint through a token-based authorization mechanism or through X.509 certificates. Such authorisations may restrict access to the hub and to some specific functionality.

This platform will have to be carefully examined in the near future. In fact, the possibility of quickly pairing the new Azure Kinect device will be of

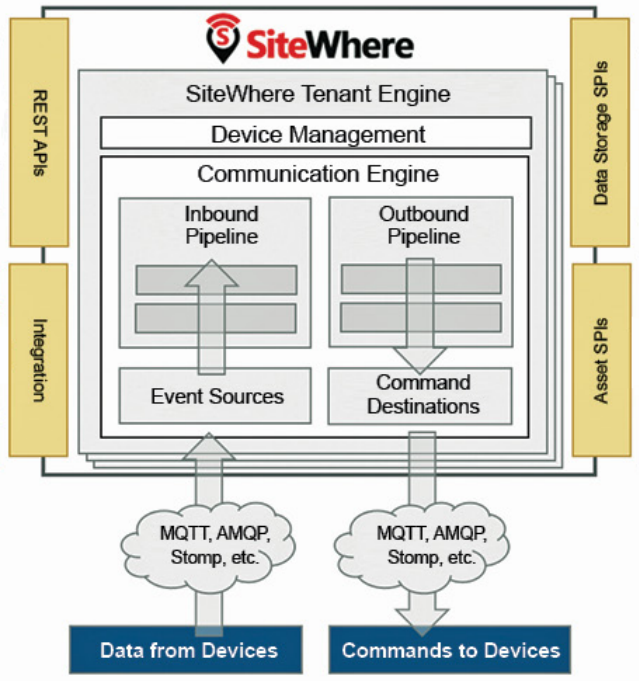


Figure 9.3: Sitewhere architecture.

particular interest. This will allow a rapid deployment of IoT RGB-D based solutions.

9.1.3 SiteWhere

Sitewhere differs from previously discussed middlewares primarily for its business model. It is indeed an open-source IoT platform, developed and maintained by SiteWhere. This solution is licensed under CPAL-1.0 (Common Public Attribution License Version 1.0). To be more accurate, two variants of this platform were released: a free for use *Community Edition*, and an *Enterprise Edition*, which consists in an extended paid-for version of the first. The latter solution need to be purchased directly from SiteWhere. Several are the requirements to deploy a SiteWhere instance: an Apache Tomcat web server should be configured as well as a MongoDB repository. Java and HiveMQ (a MQTT broker) are also required.

The SiteWhere server represents the central node through which it is

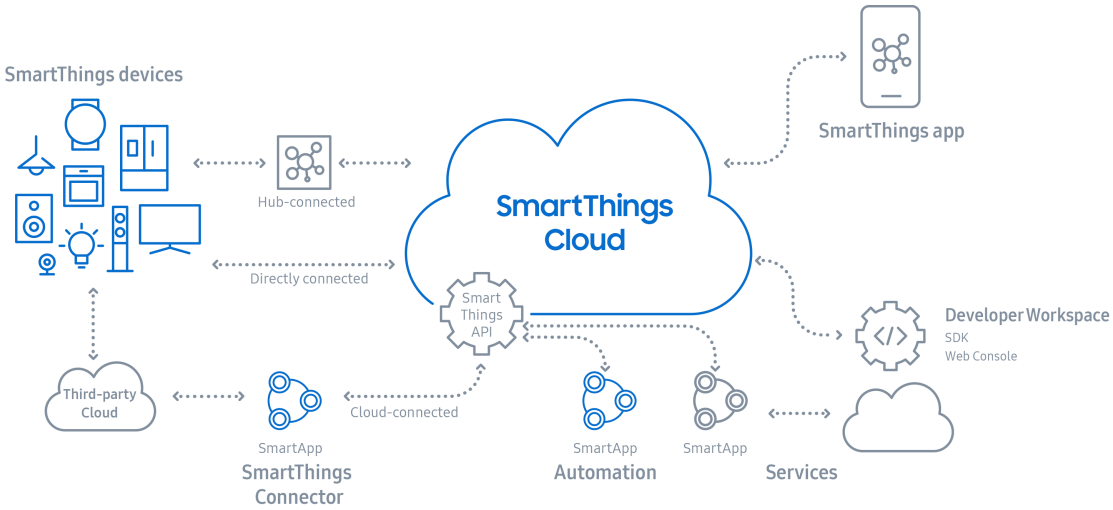


Figure 9.4: Samsung SmartThings architecture.

allowed to manage both components and REST services. This solution is designed as a multi-tenant system in which tenants are responsible for most of the processing logic. Within each server, one or more tenant engines are bootstrapped, each running as a different IoT application. In order to keep the information separate, each tenant is coupled with its own data store. As depicted in Figure 9.3, every tenant also features a processing pipeline that can be customised without affecting other pipelines. Sensors send data through a gateway which operates between tenants and devices. SiteWhere supports MQTT, AMQP and REST communications.

9.1.4 Samsung SmartThings

Samsung SmartThings is an IoT applications ecosystem. SmartThings project started in 2012 through a Kickstarter campaign. The basic idea was to realize a solution for smart domestic environments through a hub connected to a set of "things" (e.g., temperature and humidity sensors, smoke and CO alarms). As the project was started, it was coupled with a smartphone app able to communicate with the remote hub. 2014 represents a milestone for SmartThings as it was acquired by Samsung Electronics. The

initial architecture has evolved considerably to become a genuinely cloud-centric platform. Indeed, as depicted in Figure 9.4, it is now possible to connect devices to the cloud back-end following three different strategies, even without the aid of a *direct hub connection*. Another type of connection is the *cloud-connected* one, that makes possible to implement an indirect communication channel between (cloud-based) third-party devices and the SmartThings cloud. While the presence of a hub that acts as a gateway between devices and the cloud is recommended, several operations could be performed locally, without the need to query the back-end. In this specific case, SmartThings refers to these devices as *hub-connected* and they rely on ZigBee or Z-Wave communication protocols. In SmartThings applications, objects are usually organized and grouped according to the room they are in. The *room* concept is therefore a key aspect for SmartThings clients.

SmartThings includes the concept of *automation* which allows the user to interact with the ecosystem without any manual intervention. With respect to automation, two are the possible strategies to adopt: the first relies on WebHook, the second on AWS Lambda functions. For instance, it is possible to define an automation strategy designed to adjust light intensity within a particular room according to weather changes.

This cloud solution also supports encrypted communications between all connected components through the SSL/TLS protocol (Ammar et al., 2018). Although the architecture offered by SmartThings is solidly aimed at the domestic environment or, more generally, at the *smart building* concept, its features make it possible to adopt it in broader contexts. In particular, thanks to *cloud-connections*, it would be possible to hook up a sensor network.

Chapter 10

IoT Manager

As we are witnessing to the convergence of the IoT and the cloud computing paradigms, sensor networks are being deployed everywhere and grow both in number and significance. One of the main concerns is thus to provide the community with versatile and resilient frameworks capable to store and rearrange data collected by these sensors. However, the world largest information technology companies tend to release products in a as a service fashion, avoiding to reveal the know-how concerning design and implementation details. As a consequence, a common trend for academic institutions is to use these mainstream IoT platforms as *'black boxes'*. In this chapter, is presented IoT Manager, a general framework designed for sensor networks monitoring which was entirely developed within the University of Bologna. Through this case study, we provide the scientific community with a detailed implementation strategy concerning our specific IoT solution. Our results are supported from a LGPL release of the IoT Manager client in order to serve as a test bed both for research and teaching purposes.

10.1 Architecture

(Calderoni et al., 2014) proposed a general ICT architecture designed to manage several subsystems in urban contexts. *IoT Manager* represents an

evolution of this model and implements its main features. In this section, we want to clearly explain how our platform was designed and implemented, in order to provide the reader with a tangible example of a fully open IoT stack. As pointed out, an IoT Player does not frequently reveal details about its solution. In addition to this, it is increasingly common to see platforms that are not supported by exhaustive details about the connection of sensors. Discussion typically focuses on the IoT middleware layer, its infrastructure and the services offered. However, this leads to neglect some relevant details concerning on the one hand the physical component that has to communicate with the middleware, and on the other hand the possible application component. In this section we want to face this discussion in its entirety: through a top-down approach we will analyse *IoT Manager* not only discussing the role of integration middleware but also describing the physical and application layers of the stack. Indeed, this will allow us to describe the sensors used in a real case study (see Section 10.2) and in a hypothetical scenario (see Section 10.3), make a comparison between our proposed middleware and those introduced in Section 9.1 and illustrate a client application connected to the platform.

From a high-level architectural point of view (see Figure 10.1 for reference), this system is composed of three layers. The networking layer, introduced in Section 4.1, is natively embedded in our proposal as it represents wireless communication technologies and techniques that enable the transmission and reception of data from/to sensing layer.

The **sensing layer** consists of a number of heterogeneous sensor networks. These networks can be distributed anywhere in the globe and their purpose is to collect raw data. In addition, the platform is fully geo-referenced, allowing application-level filtering based on sensors effective location. Range queries may also be addressed in relation with the user's current position, as examples consider a user who wants to check the air quality in a specific area or who wants to monitor some critical information gathered from the sensors deployed in the serviced smart homes in his surroundings. The geo-referencing

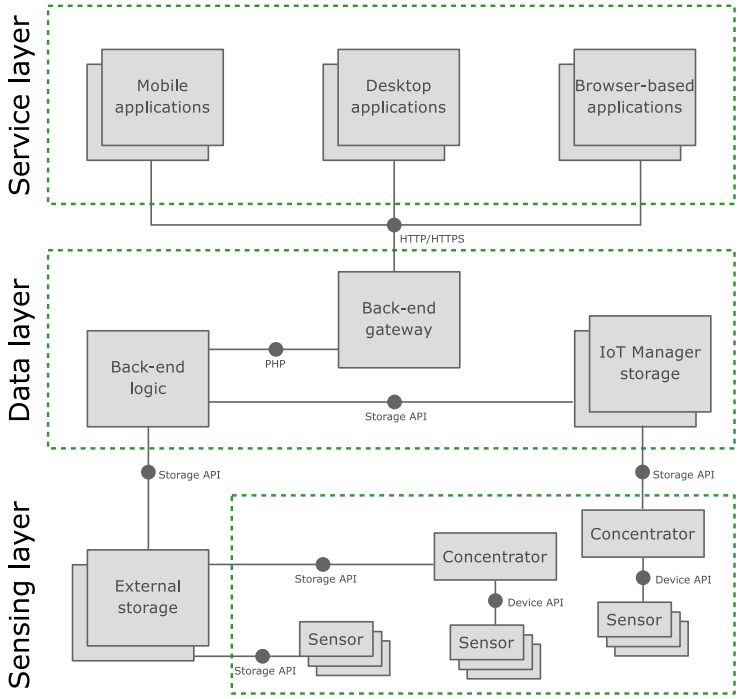


Figure 10.1: A high-level diagram showing IoT Manager architecture.

of the sensors does not offer guarantees regarding the logical division of the sensors of interest. Therefore, IoT Manager is designed to natively support sensors with a multi-tier taxonomy. In detail, in the sensing layer a network node can be treated either as a *simple sensor* or as a *concentrator*. In this second case, the purpose is to represent a logical set of different simple sensors. Thanks to the multi-level taxonomy, the *back-end gateway* allows for requests which only address the set of simple sensors connected to a given concentrator. Raw data from sensors and concentrators are sent to the middleware via APIs that depend on the storage engine adopted. IoT Manager currently has its own internal storage, but through a set of predefined APIs it is possible to integrate data from sensor networks whose storage is external to the back-end. This is another key aspect of our solution: it is possible for third parties – such as a citizen, health-care professionals or a caregiving company operator – to connect a specific sensor or sensor network. Within Section 10.1.1 we describe some sensors which are already handled by IoT

Manager and we also detail the procedure used to build one of this devices from scratch.

The **data layer** represents the back-end of the system and is responsible for two main features: on the one hand, it serves as a repository for all of the sensed information, on the other hand, it provides several API which may be called by client applications in order to query those data and retrieve them in a properly arranged format. As we have seen, this level plays the key role of maintaining compatibility between the various subsystems. It also provides the application layer with an efficient and transparent way to access data. The role of integration middleware is covered by the *back-end logic* module (see Figure 10.1) that represents the more sophisticated component of the system.

Specifically, this component is able to retrieve raw data produced by sensors and concentrators using a set of predefined APIs which allow it to query different storage engines. Furthermore, information recovery is empowered both for internal and external storages. Albeit raw data might be retrieved from a wide range of different repositories, the back-end logic can revise these records in order with the goal for them to conform to a particular format, in accordance with the *back-end gateway* dispositions. The back-end gateway is another key component of this layer. It exposes HTTP/HTTPS APIs to enable communication with client applications. It is also responsible for requests translation (in a set of jobs handled by the back-end logic component) and for final response formatting (JSON). An in-depth discussion about the back-end gateway and the back-end logic is provided in Section 10.1.2.

The **service layer** (also known as application layer) offers users a wide range of possible client applications that communicate with the back-end gateway through appropriate APIs. These APIs are currently based on HTTP and HTTPs protocols, which makes integration with desired user application quite simple. Within Section 10.1.3 we provide a detailed design of one of these client applications, which has been developed for Android mobile devices. Clients are subject to a specific access policy and handle

geo-referenced data.

10.1.1 Sensing layer: some examples

Our solution deals with different types of sensors, one of which consists in a low-cost weather station. This prototype relies on a UDOO Neo Extended board. This board is equipped with a NXP i.MX 6SoloX processor with two different core: an ARM Cortex-A9 and a Cortex-M4 (an Arduino UNO-compatible platform). In addition, it is provided with 1GB RAM, a Bluetooth 4.0 receiver, a Wi-Fi module and 9 integrated sensors (3-Axis accelerometer, magnetometer, gyroscope) which were not considered in our case study. Finally, there is an I^2C (Inter Integrated Circuit) connector used to plug sensor modules (UDOO bricks). One of the main features concerning UDOO bricks is the ability to work through a cascade configuration: it is allowed to connect several sensor modules using the sole I^2C interface on the board. Of course, it is also allowed to connect sensors directly to the Arduino socket provided by the board (Borrello et al., 2015).

In our experiment, we used three different sensor modules: a *Barometer brick* (based on MPL3115) that is able to sense pressure (hPa) and temperature ($°C$), a *Light brick* (based on TSL2561T) that returns illuminance (Lux) ambient values and a *Humidity brick* (based on SI7006-A20) providing relative humidity percentage. Similarly to (da Silva et al., 2015), this weather station can also be used for monitoring air quality in indoor or AAL environments.

We have developed a simple bash script which allowed us to read data from the barometer. Conversely, as part of our implementation relies on external libraries, other bricks were handled through an Arduino sketch. Data received from each sensor are collected by the UDOO operating system and then sent to an external storage via HTTP/S API. In order to comply with IoT Manager specifications, the payload also includes some mandatory information (*sensor identifier*, *sensor name*, *subsystem identifier*, *status*, *latitude* and *longitude*). These fields are introduced in Section 10.1.2.

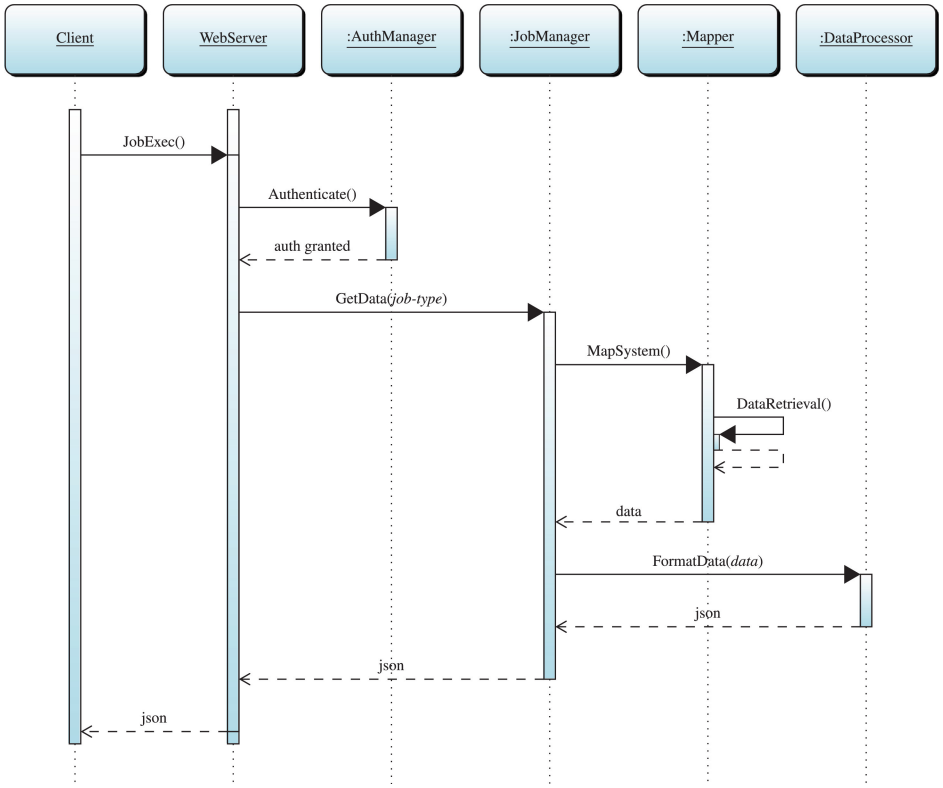


Figure 10.2: IoT Manager requests processing from the back-end perspective.

Besides weather stations, the sensing layer is currently composed of a number of other urban devices such as *ArLu* and *Lamps*. An ArLu, representing a lighting cabinet, acts as a *concentrator* and is logically connected to a set of simple sensors (Lamps) allowing a full lighting system management. This two-level taxonomy enables a logical partition even when ArLu and Lamps are not physically connected one each other.

10.1.2 Data layer: the back-end logic

When a client application queries the back-end for data, the data layer acts as outlined in Figure 10.2.

The client application delivers a request over a HTTP/HTTPS post channel. The web server, implementing the back-end gateway, handles this request and, first of all, checks for user authentication. This operation is per-

Table 10.1: IoT Manager input parameters derived from the HTTP service contract exposed by the back-end gateway.

Parameter	Description
user, pwd	Username and password for authentication.
filter	Susbsystem identifier (0: all subsystems).
id	Single sensor or single city/zone identifier, depending on the job.
minLon, maxLon	Longitude bounding values.
minLat, maxLat	Latitude bounding values.
job	Job identifier, as outlined below.

formed by the *AuthManager* class, a specific software component which addresses authentication queries to the central IoT Manager storage. Thanks to a complete integration with prepared statements, this module preserves the framework from being affected by SQL injection. On authentication granted, the back-end gateway instantiates a *JobManager*: this module checks for the type of the handled request and instantiates in turn a *Mapper* object in order to retrieve data. The set of request parameters and the job types supported by our framework are reported in Table 10.1 and in Table 10.2.

While jobs 3, 4, 5 and 7 depend on meta data and affect IoT Manager storage only, jobs 1, 2 and 6 may also affect a number of external storages. In fact, as previously discussed (see Figure 10.1 for reference), our framework is able to retrieve raw data both from its own storage and from a number of external sources. As we may notice, each of these jobs is completely transparent with respect to the calling application concerning real data location. Thanks to a set of back-end APIs, the Mapper object connects to each subsystem and retrieves each relevant record. Desired records are thus collected by the JobManager object and prepared for being returned to the calling application by the *DataProcessor* (see Figure 10.2 for reference). The latter class is responsible for data formatting in compliance with the service contract through JSON notation.

Table 10.2: Job types derived from the HTTP service contract exposed by the back-end gateway.

Job	Description
1	Returns a list of sensors lying within a specific bounding box specified by the calling application. Depending on the *filter* parameter, it is possible to address this request to a specific subsystem (a specific set of sensors) or to each subsystem.
return	[*id, name, subsystem, longitude, latitude, status*]:list
2	Returns a single sensor and all of its related information in a key-value fashion. The identity of the sensor is provided in the request through the couple *subsystem, id*.
return	[*attribute name, value*]:list
3	Returns the list of subsystems handled by IoT Manager.
return	[*subsystem, name*]:list
4	Returns the list of known cities/zone in the back-end atlas.
return	[*city, name*]:list
5	Returns a single city/zone and all of its related information.
return	*city, nation, name, longitude, latitude, gmt*
6	Returns the list of sensors connected to a specific concentrator (uniquely identified by *subsystem, id*).
return	[*id, name, subsystem, longitude, latitude, status*]:list
7	Returns a key-value list exposing a semantic description of each attribute for each specific subsystem.
return	[*attribute name, description*]:list

Our back-end logic thus relies on several APIs for data retrieval. It is important here to point out that each retrieved record may belong to a sep-

arate subsystem, each holding specific features. As a consequence, data may contain a large number of heterogeneous attributes. This is the reason why we defined a restricted set of attributes which subsystems need to exhibit as a mandatory requirement for them to be connected to IoT Manager. Specifically, these attributes shall represent a *sensor identifier* (unique within its own subsystem), a *sensor name* (or description), the *identifier of the subsystem* they belong to, a *status* information and a couple of fields specifying the *longitude* and *latitude* coordinates of the sensor. It is meaningful to note that these data do not need to be stored under a single or predefined column name. For each external source, the Mapper queries IoT Manager meta data in order to know which column or columns contain each mandatory information and which names represent those columns within the external storage schema. This mapping feature provided by the back-end logic allows for a proper implementation of jobs 1 and 6 which, as should be noticed, produce a list of compliant information derived from heterogeneous subsystems. This allows client applications to easily handle sensor lists throughout each part of the user interface where sensor-specific details are not required. Conversely, when a calling application would require something specific about a single sensor, a different mapping principle applies. This is indeed the case of job 2. The back-end logic access the aforementioned meta data and search for column mapping concerning sensor and subsystem identifiers. Through the proper connection API, the Mapper queries target storage for each data related to the sensor and blindly collect them. Sensor-specific data are then JSON formatted and returned through the HTTP service in a key-value fashion. The calling application is thus responsible for data interpretation. In order to build a proper user interface and to correctly show meaningful data, end-user application developers may rely on job 7, which provide the client with a human readable description of each returned field. Finally, a couple of words about georeferencing. IoT Manager natively supports positional data. Mobile services built against the IoT Manager framework may use GPS coordinates to enrich their queries with bounding box information. However,

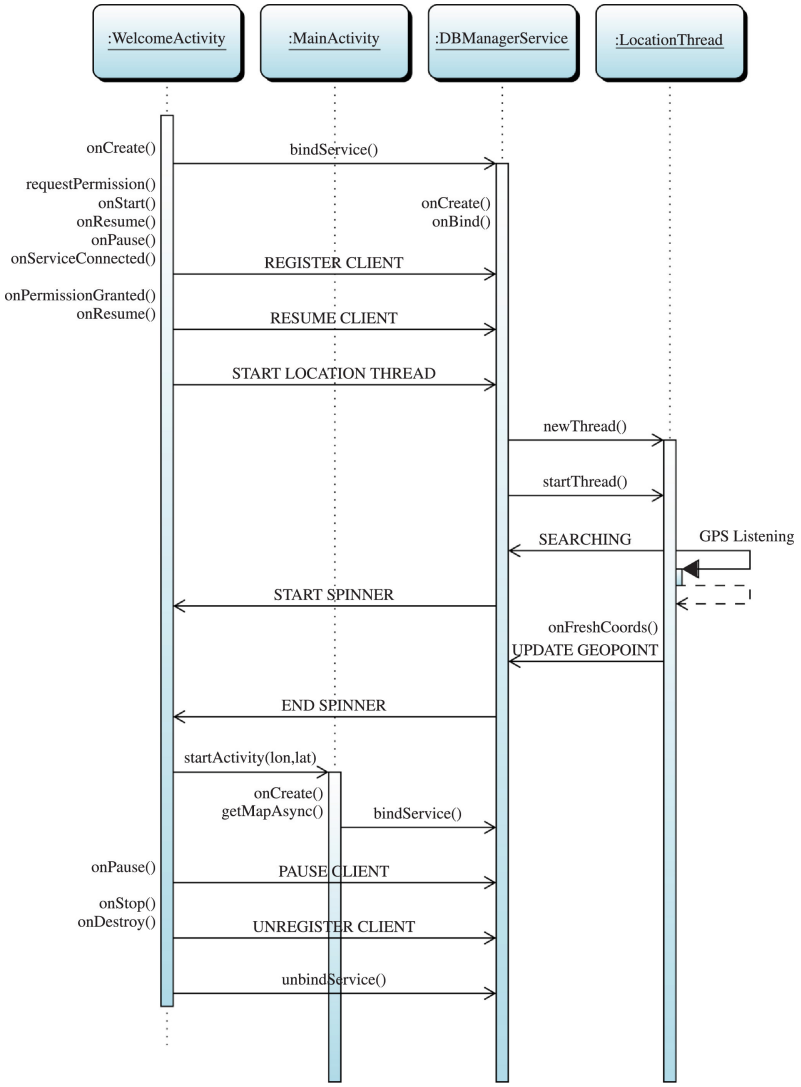


Figure 10.3: Service Layer: Launch sequence (Android API level $\geq$ 23).

when a client application is not aware of its location, or when the hardware it is executed on is not equipped with any form of location sensing device, job 4 and 5 may be used to simulate user's position as derived from the framework atlas.

10.1.3 Service layer: an example of client application

As previously discussed (see Figure 10.1 for reference), the service layer is an ensemble of applications designed to interact with IoT Manager data layer. Within this section we explore this layer through a real application which was designed and implemented by our research team. The service we are about to discuss consists of a mobile application built over Android OS.

The main aim of this application is to sense location information through GPS and network hardware and to display sensors which lie within a given distance with respect to the device itself. The user is also allowed to displace his position using one of those provided by the back-end atlas. As this application is intended to be used in the IoT domain, it is designed with multi-threading and asynchronism in mind. Each client *activity* relies on a shared Android service in order to obtain positional data as well as each kind of external data to be downloaded through IoT Manager HTTP API. The launch sequence of our application is described in Figure 10.3.

As we may see, a *welcome activity* is initially started along with a *service* running on a separate thread. The activity first checks for user permission concerning GPS and Network and, on permission granted, asks the service for location coordinates. The service then starts a dedicated thread which implements several primitives provided by Android OS able to deal with GPS and Network sensing. When a fresh position is sensed, the location thread sends a message to the service, which in turn sends these new coordinates to each connected activity. As the welcome activity receives coordinates, the program control passes to the *main activity* which immediately binds to the service. The main activity first checks for authentication information within the application preferences. Figure 10.4 shows its starting sequence assuming these credentials were already provided by the user.

While the Android service and the location service keep on running on their own threads, this sequence diagram shows a new type of thread which has been designed to handle back-end calls. During its starting sequence, main activity asks the service for a number of external data. For each

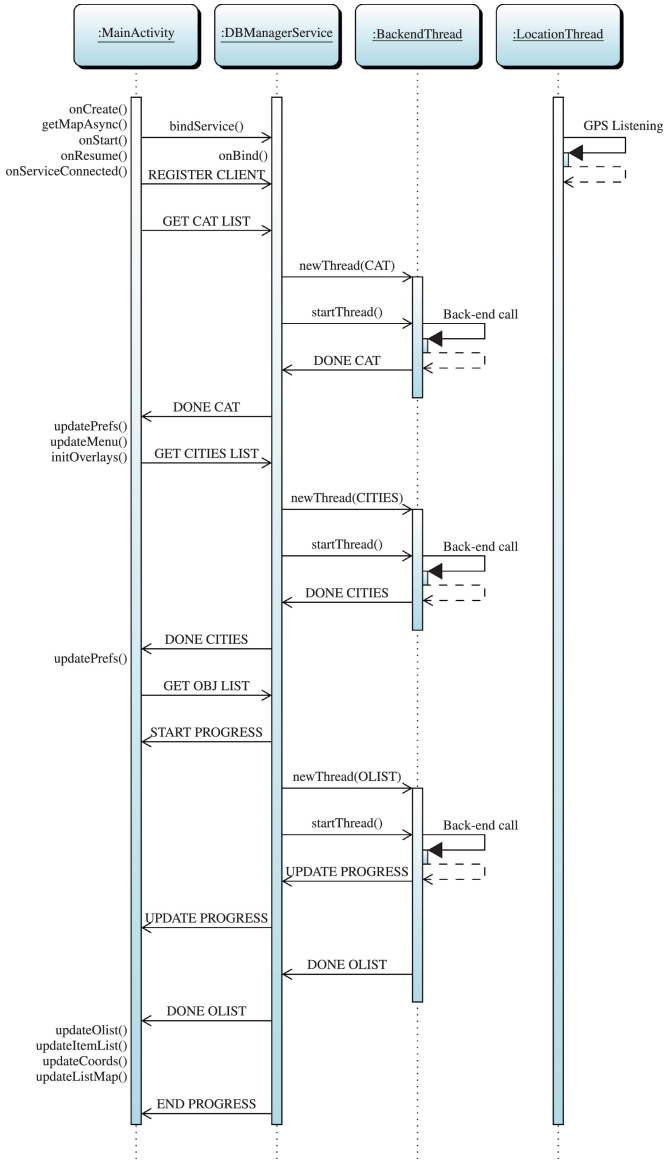


Figure 10.4: Service Layer: main activity starting sequence. Here we assume login information has been already filled in the application preferences and no specific location from the back-end atlas was selected instead.

task, the service instantiates a single thread implementing the IoT Manager communication service and propagates the request to the endpoint through HTTP/S. Specifically, it first asks for the complete list of subsystems handled

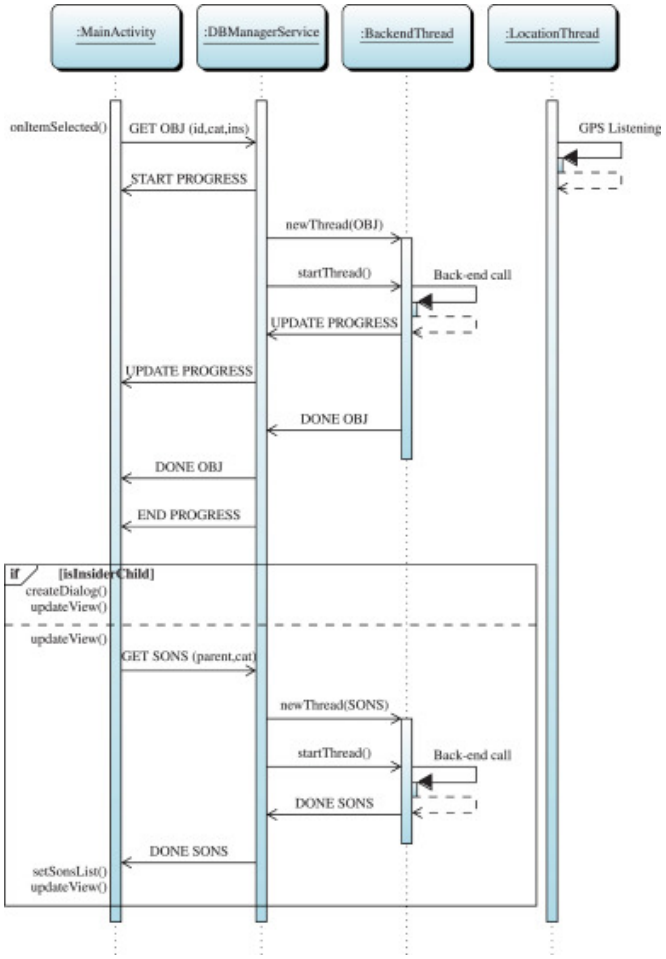


Figure 10.5: Service Layer: Sensor details request. The collected information replaces the map layout container or, when the request arises from that container itself (isInsiderChild = true), it is shown in a dedicated dialog.

by the back-end (Table 10.2, job n.3). Then it requests the list of cities/-zones stored in the back-end atlas, used to populate a specific combo box within application preferences (Table 10.2, job n.4). Finally, it asks for a list of sensors (belonging to any subsystem) which lie within a predefined range from the user (Table 10.2, job n.1).

As we may see, each back-end call is handled by a specific thread and does not affect the application responsiveness at all. Please note that each *back-*

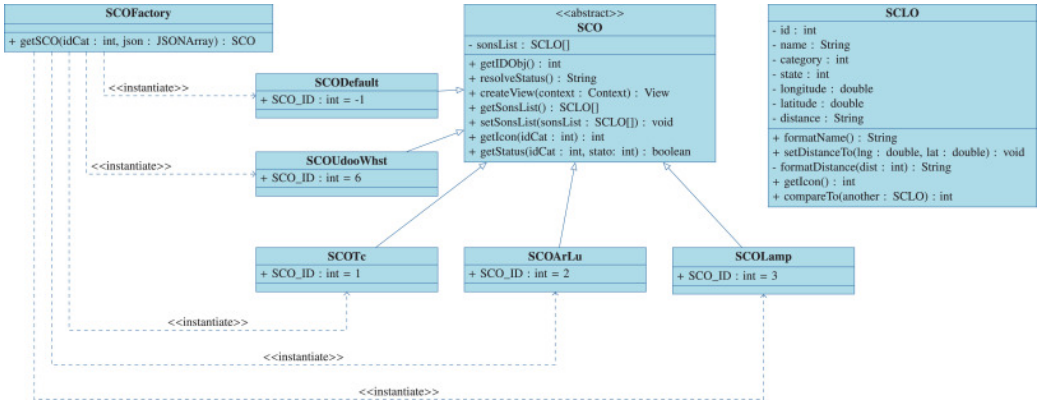


Figure 10.6: Service Layer: the Android client class factory. The abstract class representing a single sensor and some of its specializations.

end call depicted in Figure 10.4 may be exploded with the sequence diagram provided in Figure 10.2. When these calls are completed and information is returned to the calling activity, the application GUI is updated with sensors data. A sorted list of sensors (with respect to the distance to the user) is populated on the left, while a map showing an overlay icon for each sensor is proposed on the right. It is meaningful to point out that, at this stage, no detailed sensor information is required. In order to populate list and map it is enough to know few basic information as those returned by job 1 or 6 (see Table 10.2 for reference). Consequently, our GUI is subsystem-independent and is able to deal with heterogeneous sensors with no need for specific personalization. Processing of sensors list and map relies on a specific class called *SCLO* (see Figure 10.6). The role of this class is to store basic sensor information for those devices included in the current bounding box. Such information constitutes the instances objects of the *SCLO* class. As we may see in the class diagram depicted in Figure 10.6, the SCLO class exposes a number of methods. Among them, it is meaningful to underline those dealing with distance evaluation with respect to the user's position. These methods and fields are relevant as they enable location-based filtering and sorting. Again, it is important to note that the abstract class *SCO* includes the *sonsList* field, consisting of an array of *SCLO*. It also contains the related

overloads of the *getSonsList* method. As we may see, each instance of the *SCO* class keep all the information related to sensor's sons in a compact and interoperable fashion via the *SCLO* class.

When the user clicks or taps on a specific sensor, a request for sensor's details is propagated to the back-end, as depicted in Figure 10.5.

This sequence implements the call for job n.2 (see Table 10.2 for reference). When the download process terminates and the information is delivered to the main activity, the GUI is properly updated. Again, as the given sensor could be a concentrator, another back-end call (job n.6) is propagated in order to show the list of related sensors. Conversely as per the sequence proposed in Figure 10.4, the information to be shown is sensor-specific and, thus, a specific layout needs to be designed to arrange it. Our Android client is conveniently designed to this purpose and it is provided with a *class factory* which instantiates the proper object on a subsystem basis. A simple class diagram showing some specializations of the abstract class implementing a single sensor is provided in Figure 10.6.

Each sensor class need to specialize an abstract method *createView()*. This method should contain those instructions used to render a proper layout for the sensor. Consequently, when we need to show some sensor-specific detail within the GUI, it is sufficient for us to call this method on the object representing the given sensor, without any other knowledge about its features.

10.2 Case study: a Smart City scenario

The open source framework IoT Manager was firstly designed and introduced to reflect the needs of several partners of the University of Bologna in a smart city scenario. As each partner possessed a different, separate sensor network, the main goal was to allow these networks to join the middleware without any ad-hoc intervention. This challenge represented an excellent case study for both industrial and research purposes, and positively contributed to the platform implementation process.

Table 10.3: Geographical distribution and quantification of the various types of sensors currently involved in our case study.

Sensor	Quantity	Distribution
ArLu	$\simeq 50$	Europe
Lamp	$\simeq 500$	Europe
Traffic Controller	$\simeq 30$	Europe and Morocco
Weather Station	$\simeq 10$	Italy

Currently, IoT Manager involves four different types of sensors: in addition to the already mentioned ArLu, Lamp and Weather Station (see Section 10.1.1) a sensor called *Traffic Controller* (TC) is also handled. This sensor is based on a smart camera that continuously monitors a road section using some virtual spires placed on the lanes. The TC is responsible for counting, classifying and estimating the speed of vehicles crossing the virtual coils that are placed in strategic points of the roadway. Although the number of sensors is not very high, they are widely spread across the European continent (see Table 10.3 and Figure 10.7). Data collected by these sensors were derived from an agglomeration of corporate databases and research outcomes as the result of a number of collaborations between the University of Bologna and other institutions.

Each among the aforementioned sensors belongs to a different network involved in some kind of outdoor urban sensing. Specifically, the TCs sensor network provides traffic monitoring information upon several major arterial roads in different European countries. Conversely, the network comprising ArLus and Lamps is used for public lighting management and is mainly deployed in Italy, France, and Germany. Finally, weather stations are part of a prototype network deployed in Italy solely, and they are designed for air quality and weather conditions monitoring. As discussed above, these sensor networks were already operative, and belong to different companies. Therefore, they are part of different and separate systems and they store raw data on separate remote data bases. Thanks to IoT Manager, we were able to harmonise these networks transparently. While they still collect data in

each respective storage system, IoT Manager is able to retrieve each data on the middleware and offers a unified application for an easier sensor network management.

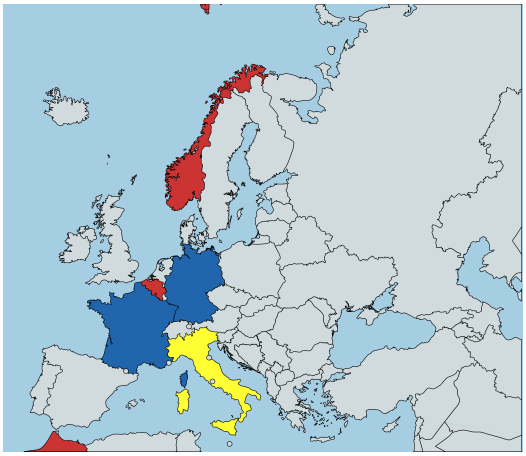


Figure 10.7: Distribution of the various types of sensors that are part of the IoT Manager sensing layer in a real Smart City scenario. In red, those countries in which TC sensors are deployed. In dark blue, those countries involving ArLu and Lamp. Italy (yellow) is the only nation where all of the currently handled sensors have been deployed.

It is finally important to stress that IoT Manager is designed with research and teaching purposes in mind. We released an open distribution of the client application introduced in Section 10.1 on *GitHub*[1]. This approach allows students and researchers to synchronize their IDE with IoT Manager's repository and to develop their own IoT solutions against the framework.

10.3 Case study: an AAL scenario

As stressed in the previous sections, IoT Manager currently includes the definition of four specific classes of sensors operating in Europe. In order to demonstrate the applicability of IoT Manager to a different application

[1]https://github.com/smartcitylabunibo

scenario, we have assumed the adoption of the framework and the Android-based application in a different context. Therefore, we have hypothesised the realisation of a monitoring system that concerns several AAL environments. As we have seen, this constitutes one of the most current and critical application scenarios in this domain. The starting hypothesis is that these environments are, of course, independent subsystems, realistically designed with different technologies and solutions specific to AAL (as seen in Section 4.1). The prototype realised exploits the functionalities previously described and the interoperability guaranteed by the framework, defining a first tool for the effective human behaviour monitoring. The two-level taxonomy allows the rapid definition of a generic AAL environment. In fact, we have identified two new instances of the abstract class SCO: the *SCOAAL*, whose purpose – similarly to what observed for the ArLu – is to act as a logical concentrator for the *SCORooms*, introduced to model the concept of room logically belonging to a specific environment. Both the SCOAAL and the SCORoom can include different sets of sensors. The primary goal in the realisation of this scenario was to define environments in which embedded systems, interconnected with RGB-D sensors, were deployed. To this end, we introduced another new class called *SCOEmbKinect*. Of course, these systems exploit the HAR and template co-updating techniques described in the Part II of this thesis. Currently, our simulation is based on the random selection of video sequences belonging to the datasets described in the previous chapters. For example, we exploited the CAD-60 room division to define different SCOAAL composed of five different SCORoom, one for each environment represented in the dataset. Of course this prototype is a simplification limited to recognition, so it does not take into account the necessary detection and segmentation operations. As a result, it is possible to create a lifelog of the actions carried out in the various environments: this allows simple remote monitoring of the user, with the possibility to analyse and highlight any changes in behaviour or wrong behaviour pattern in the short/long term. A concrete example can be the analysis of the subject's feeding habits and in

particular of the assimilation of liquids during the day. The work presented by (Gasparrini et al., 2015), for example, shows how monitoring these habits, especially in the elderly, can be decisive in defining a healthy lifestyle.

The natural evolution of this prototype is a system that provides the monitoring of different aspects, more or less critical. Among the most relevant scenarios, the supervision of the correct medicine intake, the detection of falls or, in general, abnormal behaviour, for example in subjects suffering from dementia, in a typical context of an Emergency Monitoring System (EMS). The recognition of critical events would generate an alarm for caregivers or health-care professionals, allowing for the fastest intervention based on their position or assigned zone. However, it must be specified that the prototype does not currently provide the transmission of any video sequence to the service layer. Indeed, it is realistic to assume that, in the case of alarm management, the sequences of interest must be analysed remotely, possibly restricting only to depth frames to preserve the users' privacy.

10.4 Discussion and future improvements

IoT Manager's goal is twofold: first, to provide researchers and practitioners with a full-stack platform that enables rapid deployment of prototype IoT solutions; second, to provide guidance at all architectural levels for the production of open-source IoT layers/platforms. Commercial solutions presented in Section 9.1 offer a typically partial or compartmentalised view. A rather evident lack is the absence of operational details concerning the application layer. A full-stack solution, as IoT Manager represents, could be useful for research groups to understand how to build an IoT platform from scratch and to quickly hook up sensor networks. IoT Manager also help the designer in the customisation of the client application which needs to be implemented according to the requirements of specific application contexts. This feature is usually provided by *Application Development Platform* (which fall outside of the scope of our evaluation) while it is rarely adopted by *Application En-*

ablement Platform, which focus is posed on the middleware (see Section 9.1 for details).

Therefore, as stressed before, the main features of IoT Manager are *(i)* its interoperability and *(ii)* its full-stack architecture. Concerning *(i)*, as we have seen in the Section 10.2, the proposed framework allows the rapid coupling of entire networks of sensors, even for those which already operate. The only constraint is the existence of the six mandatory information (*sensor identifier*, *sensor name*, *subsystem identifier*, *status*, *latitude*, and *longitude*), as discussed in Section 10.1.2. In specific application contexts, this feature makes IoT Manager's interoperability more agile than its commercial counterparts, which often require the creation of an ad-hoc digital twin (e.g., AWS IoT Core) for each connected device. In relation to *(ii)*, IoT Manager is combined with a complete end-user application framework which enables to quickly define the taxonomy of the different types of sensors involved in a project. The client application manages this taxonomy with a class factory design pattern. This feature allows the rapid rendering of customized graphical interfaces, potentially relying on those which are already provided on GitHub. Besides, although IoT Manager was initially designed for urban contexts, as we have seen in Section 10.3, the presence of hierarchies makes it possible to adopt it in several other scenarios, including the AAL scenario, home automation or, more generally, in smart buildings. Again, it is allowable to use sensor hierarchies to define groups of sensors belonging to the same place. For instance, the introduction of a hierarchical subdivision by rooms, as detailed in the previous section, may reflect the sensors partitioning provided by Samsung SmartThings.

Our research and teaching team is constantly working on IoT Manager platform. Several modules were implemented during the recent years in order to expand the data layer capabilities as well as to extend the set of subsystems handled by the framework. Several efforts have been also carried out in order to improve the service layer. As one of the main concern of IoT Manager is interoperability, we will devote our attention to the platform's APIs.

Two are the main challenges with respect to this subject: first, a wider set of communication protocols should be exposed by the back-end gateway. As an example, several IoT platforms accept connection from MQTT or WebSockets protocols, which are not handled by our middleware at the moment. Second, the back-end mapper should be provided with a wider set of external storage engine APIs. This condition would indeed lead to an easier connection of pre-existing subsystems. Specific attention should be posed on NoSQL databases and column-based storage engines. We are currently working on an additional module located between the *Sensing Layer* and the *Data Layer* (see Figure 10.1) in order to enhance our three-layered stack. The mission of this module is to act as a dispatcher between sensors and the back-end allowing a two-way message exchange. The dispatcher should be combined with appropriate APIs for sensors connection. Implementing this component would enable a publish/subscribe paradigm similarly as discussed for AWS IoT Core (see Section 9.1.1) and represents one of the most insightful challenges of the IoT Manager project. Moreover, this module would allow the realization of the scenario described in Section 10.3, allowing the real-time management of potential alarms.

10.5 Final Remarks

In this chapter we introduced IoT Manager, a full stack IoT platform relying on open source technologies. We discussed our platform in accordance with several mainstream IoT middlewares provided by well-known companies. In Chapter 9, we emphasised several common patterns which may be found in commercial platforms while, in this one, we discussed our own solution with respect to these reference architectures. As a lot of research and teaching projects within this field rely on hidden details which private companies do not tend to unveil, our main aim was to provide the scientific community with a tangible implementation of such a solution, along with a detailed description of our design strategies at each level of the stack.

Part IV

Conclusion

Chapter 11

Results achieved and future works

In this thesis, two relevant topics concerning AAL and smart environments have been explored. The former is about the design and implementation of different human action recognition algorithms while the latter deals with the presentation of a monitoring platform that could allow the analysis of human behaviour through the deployment of various smart devices and sensors. This chapter will summarise the most notable contributions of the thesis and future research directions.

11.1 Discussion and Contributions

World population ageing is set to be one of the 21st century's main challenges. The costs of providing care for an ageing population will grow significantly in conjunction with a decrease in the number of workers and an increase in the number of people with disabilities or chronic conditions. Such a trend is expected to be even more significant in the next decade and will have a considerable impact on the healthcare system and thus on GDP. Henceforth, it is mandatory to take sensible action and react to this inevitable and worrying phenomenon. AmI and in particular the area of AAL offer a feasible response, allowing the creation of human-centric smart environments that are sensitive, adaptive and responsive to the needs, habits and behaviours of

the user. These technologies and approaches aim to foster the self-conductive life of the patient in his or her preferred environment, reducing dependence on health-care facilities and intensive personal care. In this context, human activity recognition and monitoring play a fundamental role.

In the opening chapter of the thesis, we have outlined the vision of this work by defining the presentation of these two lines of research followed by an overview of the main elements that characterise them in Part I. On the one hand, specifically in Part II, we proposed several RGB-D based action recognition approaches. These contributions are accompanied by a brief literature review, focused on the different information modalities which RGB-D sensors are capable of using. A first algorithm, based on skeleton data and in particular on joint orientations, has been proposed in Chapter 6. The main intention of this approach was to determine the reliability and robustness of features based on joint orientations, often neglected in other action recognition works. This algorithm achieves state-of-the-art results in CAD-60 where, considering the cross-validation leave-one-actor-out protocol (i.e., "new person" setting), it achieves 95.0% of both precision and recall. Due to the lack of joints orientations data in the most common HAR datasets, we have internally acquired a new dataset called OAD. The accuracy is rather good also in this benchmark, reaching 80.85% precision and 80.16% recall. Moreover, through the analysis of the confusion matrices of this new dataset, it was possible to understand which were the main weaknesses of the algorithm, such as the temporal ordering of specific sequences not adequately represented by the approach (e.g., sitting/getting up). To this end, a multi-modal approach has been proposed in Chapter 7. This algorithm integrates the joint orientations, used to define the main body postures, with features extracted from RGB images. More precisely, a number of temporal images are exploited to allow a more accurate description of actions temporal evolution. In this way, the results of both the CAD-60 – which reaches 98.8% precision and 98.3% recall – and the extended version of the OAD (90.6% precision and 90.4% recall) have been further increased. This multimodal approach has also achieved

good results compared to the more complex CAD-120. The Chapter 8 aimed to offer a different approach to action recognition, focusing mainly on the introduction of the concept of template co-updating which, to the best of our knowledge, has not been investigated so far. We are confident that in an AAL scenario, where a monitoring system is continuously checking the ambience to understand human actions, a considerable amount of unlabelled data can be easily collected. Moreover, labelled training data are often scarce, with a few video samples for each activity to be recognised. We believe that the implementation of incremental (co-)updating techniques is mandatory to fully exploit the richness of (multiple) data that the specific scenario naturally provides. We have thus proposed a semi-supervised approach that allows the incremental learning of templates using multiple data sources (skeletal and RGB information). The proposed multimodal approach exploits, also in this case, the joint orientations features combined with the robust IDTs, used to represent the RGB data. The experimental phase, limited to OAD v.2.0, is promising, with interesting trends and offering good results (89.3% precision and 88.7% recall). Of course, these metrics are not comparable with the results previously described, having adopted an incremental approach and a different testing protocol.

On the other hand, in Part III, we presented a monitoring platform for generic IoT environments compared to some of the leading solutions on the market. We have emphasised two main aspects of this solution: i) its interoperability and ii) its full-stack architecture. This platform is in fact operative for the monitoring of heterogeneous sensors distributed throughout Europe. These sensor networks are managed by different subsystems that are using different technologies. In Chapter 10, we have thus offered the reader the possibility of understanding the design of such a platform. The proposed solution can be quickly adapted for the rapid prototyping of different IoT solutions; specifically, we have simulated its use for the monitoring of sensors distributed in several smart homes in an AAL context.

11.2 Future works

This section analyses the main open issues and possible research directions concerning the techniques of action recognition, template co-updating and the monitoring platform. In the different action recognition algorithms, we have strongly exploited the joint orientations. The features derived from them has proved to be effective in the benchmarks on which it has been validated. The generation of the key poses codebook is based on the use of the clustering algorithm k-means applied to all the feature vectors extracted from the whole training set. A possible extension could be the production of a *"smart codebook"* which is generated through a clustering algorithm that is aware of the class of a particular training sequence. In this way, the clusters of key poses would presumably be more significantly separated. This extension should be further explored, especially if the actions share common poses that could entail a certain degree of redundancy.

Another aspect that can be addressed is the adoption of techniques that allow preserving the temporal ordering of key poses, in order to avoid being bound to specific representations such as the proposed temporal images.

As for the co-updating template approach, it will be assessed against different skeletal features and with the introduction of depth data. Indeed, this would allow a more comprehensive view of the algorithm, highlighting in a more precise way the pros and cons and enabling a thorough evaluation of the parameters described. We stressed the lack of datasets which include joints orientations, and we have overcome by introducing OAD. However, it is worth investigating the template co-updating approach against more complex and comprehensive benchmarks such as NTU RGB-D. Nevertheless, it is clear that the scientific community needs a sufficiently large ADLs dataset from a real-world scenario. Unfortunately, many of the RGB-D datasets used so far are obtained in an over-controlled, far-fetched environment.

Finally, in this thesis we focused on the recognition of pre-segmented video sequences. In a real context, it is necessary to implement robust detection and segmentation approaches concerning the continuous video stream.

Moreover, it is possible that the activity templates do not include all the typical actions. In these cases, the system has to discriminate against actions that are unidentified to it, for example, by an automatic discovering of action patterns which will have to be validated by a human being. A simplified solution could be the adoption of the "unknown" label, which would also be followed by human evaluation.

Another essential aspect to investigate is the explicit modelling of user interaction with objects which could represent a valuable source of information for action/activity comprehension. In fact, an user-object interaction approach could greatly simplify the recognition of some actions, characterised by a similar sequence of poses (e.g., drinking, answering the phone).

On the other hand, concerning the monitoring platform, the main future works have been defined in Section 10.4. Of course, one of the priorities in the implementation of an effective AAL monitoring system will undoubtedly be the introduction of a publish/subscribe protocol that allows bidirectional communication between the different layers, sensors and actuators. Moreover, as far as the deployment of vision-based devices is concerned, some evaluations that were not covered in this thesis are necessary, such as the setup of multi-camera view environments. Besides, some domestic environments are more sensitive to privacy issues than others (e.g., bathroom). In a context of remote risk assessment, video streaming of sequences from those environments should include mechanisms to protect users' privacy. Such as an example, by using only skeletal video representations or depth data. Finally, a smart monitoring system should include a local reasoning module that, on the one hand, acts as a gateway to the proposed middleware and, on the other hand, allows to distinguish the occurrence of specific events and autonomously undertakes related intervention actions. The inclusion of this module is clearly domain-specific and will be investigated to further verticalize the platform towards practical implementation in the AAL domain.

Bibliography

Sami Abu-El-Haija, Nisarg Kothari, Joonseok Lee, Paul Natsev, George Toderici, Balakrishnan Varadarajan, and Sudheendra Vijayanarasimhan. Youtube-8m: A large-scale video classification benchmark. *CoRR*, abs/1609.08675, 2016.

Giovanni Acampora, Diane J. Cook, Parisa Rashidi, and Athanasios V. Vasilakos. A survey on ambient intelligence in healthcare. *Proceedings of the IEEE*, 101(12):2470–2494, 2013.

Jake K. Aggarwal and Quin Cai. Human motion analysis: A review. *Computer Vision and Image Understanding*, 73(3):428–440, 1999.

Jake K. Aggarwal and Lu Xia. Human activity recognition from 3d data: A review. *Pattern Recognition Letters*, 48:70–80, 2014.

JK Aggarwal. Understanding of human motion, actions and interactions. In *IEEE Conference on Advanced Video and Signal Based Surveillance, 2005.*, page 299. IEEE, 2005.

Aitor Almeida, Rubén Mulero, Piercosimo Rametta, Vladimir Urosevic, Marina Andric, and Luigi Patrono. A critical analysis of an iot—aware aal system for elderly monitoring. *Future Generation Computer Systems*, 97: 598 – 619, 2019.

Belal Alsinglawi, Quang Vinh Nguyen, Upul Gunawardana, Anthony Maeder, and Simeon J Simoff. Rfid systems in healthcare settings and

activity of daily living in smart homes: A review. *E-Health Telecommunication Systems and Networks*, 6:1–17, 2017.

Salah Althloothi, Mohammad H. Mahoor, Xiao Zhang, and Richard M. Voyles. Human activity recognition using multi-features and multiple kernel learning. *Pattern Recognition*, 47(5):1800–1812, 2014.

Mahmoud Ammar, Giovanni Russello, and Bruno Crispo. Internet of things: A survey on the security of iot frameworks. *J. Inf. Sec. Appl.*, 38:8–27, 2018.

Kevin Ashton. That 'internet of things' thing. *RFID journal*, 22(7):97–114, 2009.

Ferhat Attal, Samer Mohammed, Mariam Dedabrishvili, Faicel Chamroukhi, Latifa Oukhellou, and Yacine Amirat. Physical human activity recognition using wearable sensors. *Sensors*, 15(12):31314–31338, 2015.

Luigi Atzori, Antonio Iera, and Giacomo Morabito. The internet of things: A survey. *Computer Networks*, 54(15):2787–2805, 2010.

Juan C Augusto, Vic Callaghan, Diane Cook, Achilles Kameas, and Ichiro Satoh. Intelligent environments: a manifesto. *Human-Centric Computing and Information Sciences*, 3(1):12, 2013.

Juan Carlos Augusto. *Ambient Intelligence: The Confluence of Ubiquitous/Pervasive Computing and Artificial Intelligence*, pages 213–234. Springer London, London, 2007.

Juan Carlos Augusto, Hideyuki Nakashima, and Hamid K. Aghajan. Ambient intelligence and smart environments: A state of the art. In Hideyuki Nakashima, Hamid K. Aghajan, and Juan Carlos Augusto, editors, *Handbook of Ambient Intelligence and Smart Environments*, pages 3–31. Springer, 2010.

Stylianos Balampanis, Stelios Sotiriadis, and Euripides G. M. Petrakis. Internet of things architecture for enhanced living environments. *IEEE Cloud Computing*, 3(6):28–34, 2016.

Marco Bassoli, Valentina Bianchi, Ilaria De Munari, and Paolo Ciampolini. An iot approach for an AAL wi-fi-based monitoring system. *IEEE Trans. Instrumentation and Measurement*, 66(12):3200–3209, 2017.

Francesco Battistone and Alfredo Petrosino. TGLSTM: A time based graph deep learning approach to gait recognition. *Pattern Recognition Letters*, 126:132–138, 2019.

P. Bellavista, G. Cardone, A. Corradi, and L. Foschini. Convergence of manet and wsn in iot urban scenarios. *IEEE Sensors Journal*, 13(10):3558–3567, 2013.

Aaron F Bobick. Movement, activity and action: the role of knowledge in the perception of motion. *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, 352(1358):1257–1265, 1997.

Giulio Borrello, Erica Salvato, Giovanni Gugliandolo, Zlatica Marinkovic, and Nicola Donato. Udoo-based environmental monitoring system. In Alessandro De Gloria, editor, *Applications in Electronics Pervading Industry, Environment and Society - APPLEPIES 2015, Rome, Italy, May 5-6, 2015*, volume 409 of *Lecture Notes in Electrical Engineering*, pages 175–180. Springer, 2015.

Somar Boubou and Einoshin Suzuki. Classifying actions based on histogram of oriented velocity vectors. *J. Intell. Inf. Syst.*, 44(1):49–65, 2015.

Christian Braunagel, Enkelejda Kasneci, Wolfgang Stolzmann, and Wolfgang Rosenstiel. Driver-activity recognition in the context of conditionally autonomous driving. In *IEEE 18th International Conference on Intelligent Transportation Systems, ITSC 2015, Gran Canaria, Spain, September 15-18, 2015*, pages 1652–1657. IEEE, 2015.

Leo Breiman. Random forests. *Mach. Learn.*, 45(1), 2001.

Antoni Buades, Bartomeu Coll, and Jean-Michel Morel. Non-Local Means Denoising. *Image Processing On Line*, 1:208–212, 2011.

Luca Calderoni, Dario Maio, and Stefano Rovis. Deploying a network of smart cameras for traffic monitoring on a "city kernel". *Expert Syst. Appl.*, 41(2):502–507, 2014.

Lee W. Campbell and Aaron F. Bobick. Recognition of human body motion using phase space constraints. In *Procedings of the Fifth International Conference on Computer Vision (ICCV 95), Massachusetts Institute of Technology, Cambridge, Massachusetts, USA, June 20-23, 1995*, pages 624–630. IEEE Computer Society, 1995.

Massimo Camplani, Lucia Maddalena, Gabriel Moyà Alcover, Alfredo Petrosino, and Luis Salgado. A benchmarking framework for background subtraction in RGBD videos. In *Workshop @ International Conference on Image Analysis and Processing - ICIAP 2017*, pages 219–229, 2017.

João Carreira and Andrew Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 4724–4733. IEEE Computer Society, 2017. ISBN 978-1-5386-0457-1. URL `https://ieeexplore.ieee.org/xpl/conhome/8097368/proceeding`.

Angelo Paolo Castellani, Nicola Bui, Paolo Casari, Michele Rossi, Zach Shelby, and Michele Zorzi. Architecture and protocols for the internet of things: A case study. In *Eigth Annual IEEE International Conference on Pervasive Computing and Communications, PerCom 2010, March 29 - April 2, 2010, Mannheim, Germany, Workshop Proceedings*, pages 678–683. IEEE, 2010.

Alexandros André Chaaraoui, Pau Climent-Pérez, and Francisco Flórez-Revuelta. A review on vision techniques applied to human behaviour analysis for ambient-assisted living. *Expert Syst. Appl.*, 39(12):10873–10888, 2012.

Alexandros André Chaaraoui, Pau Climent-Pérez, and Francisco Flórez-Revuelta. Silhouette-based human action recognition using sequences of key poses. *Pattern Recognition Letters*, 34(15):1799–1807, 2013a.

Alexandros André Chaaraoui, José Ramón Padilla-López, and Francisco Flórez-Revuelta. Fusion of skeletal and silhouette-based features for human action recognition with RGB-D devices. In *2013 IEEE International Conference on Computer Vision Workshops, ICCV Workshops 2013, Sydney, Australia, December 1-8, 2013*, pages 91–97. IEEE Computer Society, 2013b.

Ang Loon Chan, Gim Guan Chua, Desmond Zhen Liang Chua, Shuqiao Guo, Paul Min Chim Lim, Mun-Thye Mak, and Wee Siong Ng. Practical experience with smart cities platform design. In *4th IEEE World Forum on Internet of Things, WF-IoT 2018, Singapore, February 5-8, 2018*, pages 470–475. IEEE, 2018.

Guang Chen, Daniel Clarke, Manuel Giuliani, Andre Gaschler, and Alois Knoll. Combining unsupervised learning and discrimination for 3d action recognition. *Signal Processing*, 110:67–81, 2015.

Liming Chen and Chris D. Nugent. *Human Activity Recognition and Behaviour Analysis - For Cyber-Physical Systems in Smart Environments*. Springer, 2019.

Enea Cippitelli, Samuele Gasparrini, Ennio Gambi, and Susanna Spinsante. A human activity recognition system using skeleton data from rgbd sensors. *Intell. Neuroscience*, 2016:21–, 2016.

Ian Cleland, Basel Kikhia, Chris D. Nugent, Andrey Boytsov, Josef Hallberg, Kåre Synnes, Sally I. McClean, and Dewar D. Finlay. Optimal placement of accelerometers for the detection of everyday activities. *Sensors*, 13(7): 9183–9200, 2013.

Mauro Conti, Ali Dehghantanha, Katrin Franke, and Steve Watson. Internet of things security and forensics: Challenges and opportunities. *Future Generation Comp. Syst.*, 78:544–546, 2018.

Diane J. Cook and Sajal K. Das. *Smart environments - technology, protocols and applications*. Wiley, 2005.

Diane J. Cook, Juan Carlos Augusto, and Vikramaditya R. Jakkula. Ambient intelligence: Technologies, applications, and opportunities. *Pervasive and Mobile Computing*, 5(4):277–298, 2009.

Javier Cubo, Adrián Nieto, and Ernesto Pimentel. A cloud-based internet of things platform for ambient assisted living. *Sensors*, 14(8):14070–14105, 2014.

Ran Cui, Aichun Zhu, Gang Hua, Hongsheng Yin, and Haiqiang Liu. Multisource learning for skeleton-based action recognition using deep LSTM and CNN. *J. Electronic Imaging*, 27(04):043050, 2018.

Mauro A. A. da Cruz, Joel José Puga Coelho Rodrigues, Jalal Al-Muhtadi, Valery Korotaev, and Victor Hugo C. de Albuquerque. A reference model for internet of things middleware. *IEEE Internet of Things Journal*, 5(2): 871–883, 2018.

Madalena Pereira da Silva, Alexandre L. Gonçalves, M. A. R. Dantas, Brunno Vanelli, Guilherme Manerichi, Stephan A. R. D. dos Santos, Mauri Ferrandim, and Alex R. Pinto. Implementation of iot for monitoring ambient air in ubiquitous AAL environments. In *2015 Brazilian Symposium on Computing Systems Engineering, SBESC 2015, Foz do Iguacu, Brazil, November 3-6, 2015*, pages 158–161. IEEE Computer Society, 2015.

Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *In CVPR*, pages 886–893, 2005.

L Minh Dang, Md Piran, Dongil Han, Kyungbok Min, Hyeonjoon Moon, et al. A survey on internet of things and cloud computing for healthcare. *Electronics*, 8(7):768, 2019.

Riccardo De Benedictis, Amedeo Cesta, Luca Coraci, Gabriella Cortellessa, and Andrea Orlandini. Adaptive reminders in an ambient assisted living environment. In *Ambient Assisted Living*, pages 219–230. Springer, 2015.

Itamir de Morais Barroca Filho and Gibeon Soares de Aquino Junior. Iot-based healthcare applications: A review. In *Computational Science and Its Applications - ICCSA 2017 - 17th International Conference, Trieste, Italy, July 3-6, 2017, Proceedings, Part VI*, volume 10409 of *Lecture Notes in Computer Science*, pages 47–62. Springer, 2017.

Christian Debes, Andreas Merentitis, Sergey Sukhanov, Maria E. Niessen, Nikolaos Frangiadakis, and Alexander Bauer. Monitoring activities of daily living in smart homes: Understanding human behavior. *IEEE Signal Process. Mag.*, 33(2):81–94, 2016.

George Demiris, Debra Parker Oliver, Jarod Giger, Marjorie Skubic, and Marilyn Rantz. Older adults' privacy considerations for vision based recognition methods of eldercare applications. *Technology and Health Care*, 17 (1):41–48, 2009.

Wenwen Ding, Kai Liu, Fei Cheng, and Jin Zhang. STFC: spatio-temporal feature chain for skeleton-based human action recognition. *J. Visual Communication and Image Representation*, 26:329–337, 2015.

Angelika Dohr, Robert Modre-Osprian, Mario Drobics, Dieter Hayn, and Günter Schreier. The internet of things for ambient assisted living. In Shahram Latifi, editor, *Seventh International Conference on Information*

Technology: New Generations, ITNG 2010, Las Vegas, Nevada, USA, 12-14 April 2010, pages 804–809. IEEE Computer Society, 2010.

Muhammad Ehatisham-ul-Haq, Ali Javed, Muhammad Awais Azam, Hafiz Malik, Aun Irtaza, Ikhyun Lee, and Muhammad Tariq Mahmood. Robust human activity recognition using multimodal feature-level fusion. *IEEE Access*, 7:60736–60751, 2019.

D. R. Faria, C. Premebida, and U. Nunes. A probabilistic approach for human everyday activities recognition using body motion from rgb-d images. In *The 23rd IEEE International Symposium on Robot and Human Interactive Communication*, pages 732–737, 2014.

Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. *CoRR*, abs/1604.06573, 2016.

F. Florez-Revuelta and A.A. Chaaraoui, editors. *Active and Assisted Living: Technologies and Applications*. Healthcare Technologies. Institution of Engineering and Technology, 2016.

Keinosuke Fukunaga. *Introduction to Statistical Pattern Recognition (2Nd Ed.)*. Academic Press Professional, Inc., 1990.

S. Gaglio, G. L. Re, and M. Morana. Human activity recognition process using 3-d posture data. *IEEE Transactions on Human-Machine Systems*, 45(5):586–597, 2015.

Ennio Gambi, Laura Montanini, Laura Raffaeli, Susanna Spinsante, and Lambros Lambrinos. Interoperability in iot infrastructures for enhanced living environments. In *2016 IEEE International Black Sea Conference on Communications and Networking, BlackSeaCom 2016, Varna, Bulgaria, June 6-9, 2016*, pages 1–5. IEEE, 2016.

Samuele Gasparrini, Enea Cippitelli, Ennio Gambi, Susanna Spinsante, and Francisco Flórez-Revuelta. Performance analysis of self-organising neural networks tracking algorithms for intake monitoring using kinect. 2015.

Rohit Girdhar, Deva Ramanan, Abhinav Gupta, Josef Sivic, and Bryan C. Russell. Actionvlad: Learning spatio-temporal aggregation for action classification. *CoRR*, abs/1704.02895, 2017.

Berto de Tácio Pereira Gomes, Luiz Carlos Melo Muniz, Francisco Jose da Silva e Silva, Luis Eduardo Talavera Ríos, and Markus Endler. A comprehensive and scalable middleware for ambient assisted living based on cloud computing and internet of things. *Concurrency and Computation: Practice and Experience*, 29(11):e4043, 2017.

Lena Gorelick, Moshe Blank, Eli Shechtman, Michal Irani, and Ronen Basri. Actions as space-time shapes. *Transactions on Pattern Analysis and Machine Intelligence*, 29(12):2247–2253, December 2007.

Ye Gu, Ha Do, Yongsheng Ou, and Weihua Sheng. Human gesture recognition through a kinect sensor. In *2012 IEEE International Conference on Robotics and Biomimetics, ROBIO 2012, Guangzhou, China, December 11-14, 2012*, pages 1379–1384. IEEE, 2012.

Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. Internet of things (iot): A vision, architectural elements, and future directions. *Future Generation Comp. Syst.*, 29(7):1645–1660, 2013.

Dominique Guinard and Vlad Trifa. *Building the web of things: with examples in node. js and raspberry pi.* Manning Publications Co., 2016.

Raj Gupta, Alex Yong-Sang Chia, and Deepu Rajan. Human activities recognition using depth images. In *Proceedings of the 21st ACM International Conference on Multimedia*, MM '13, pages 283–292. ACM, 2013.

Jasmin Guth, Uwe Breitenbücher, Michael Falkenthal, Frank Leymann, and Lukas Reinfurt. Comparison of iot platform architectures: A field study based on a reference architecture. In *2016 Cloudification of the Internet of Things, CIoT 2016, Paris, France, November 23-25, 2016*, pages 1–6. IEEE, 2016.

Mohamed Ahmed Hail and Stefan Fischer. Iot for AAL: an architecture via information-centric networking. In *2015 IEEE Globecom Workshops, San Diego, CA, USA, December 6-10, 2015*, pages 1–6. IEEE, 2015.

Fei Han, Brian Reily, William Hoff, and Hao Zhang. Space-time representation of people based on 3d skeletal data: A review. *Computer Vision and Image Understanding*, 158:85–105, 2017.

M. Hasan and A. K. Roy-Chowdhury. Incremental activity modeling and recognition in streaming videos. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 796–803, 2014.

M. Hasan and A. K. Roy-Chowdhury. A continuous learning framework for activity recognition using deep hybrid feature models. *IEEE Transactions on Multimedia*, 17(11):1909–1922, 2015.

Samitha Herath, Mehrtash Harandi, and Fatih Porikli. Going deeper into action recognition. *Image Vision Comput.*, 60(C):4–21, April 2017.

Microsoft Azure IoT. Azure Iot Suite.

Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35:221–231, 2010.

Sou-Young Jin and Ho-Jin Choi. Essential body-joint and atomic action detection for human activity recognition using longest common subsequence algorithm. In Jong-Il Park and Junmo Kim, editors, *Computer Vision - ACCV 2012 Workshops, ACCV 2012 International Workshops, Daejeon,*

Korea, November 5-6, 2012, Revised Selected Papers, Part II, volume 7729 of *Lecture Notes in Computer Science*, pages 148–159. Springer, 2012.

Gunnar Johansson. Visual perception of biological motion and a model for its analysis. *Perception & psychophysics*, 14(2):201–211, 1973.

Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Fei-Fei Li. Large-scale video classification with convolutional neural networks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, pages 1725–1732, 2014.

Pushpajit Khaire, Praveen Kumar, and Javed Imran. Combining cnn streams of rgb-d and skeletal data for human activity recognition. *Pattern Recognition Letters*, 115:107 – 116, 2018.

Margarita Khokhlova, Cyrille Migniot, Alexei A. Morozov, Olga S. Sushkova, and Albert Dipanda. Normal and pathological gait classification LSTM model. *Artificial Intelligence in Medicine*, 94:54–66, 2019.

Alexander Klaser, Marcin Marszalek, and Cordelia Schmid. A spatio-temporal descriptor based on 3d-gradients. In *In BMVC'08*.

Yu Kong and Yun Fu. Human action recognition and prediction: A survey. *CoRR*, abs/1806.11230, 2018.

Hema Koppula and Ashutosh Saxena. Learning spatio-temporal structure from rgb-d videos for human activity detection and anticipation. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, volume 28, pages 792–800. JMLR Workshop and Conference Proceedings, 2013.

Hema Swetha Koppula, Rudhir Gupta, and Ashutosh Saxena. Learning human activities and object affordances from RGB-D videos. *CoRR*, abs/1210.1207, 2012.

I. Laptev, M. Marszalek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

Ivan Laptev. On space-time interest points. *International Journal of Computer Vision*, 64(2):107–123, Sep 2005.

Steven Latré, Philip Leroux, Tanguy Coenen, Bart Braem, Pieter Ballon, and Piet Demeester. City of things: An integrated and multi-technology testbed for iot smart city experiments. In *IEEE International Smart Cities Conference, ISC2 2016, Trento, Italy, September 12-15, 2016*, pages 1–8. IEEE, 2016.

Wanqing Li, Zhengyou Zhang, and Zicheng Liu. Action recognition based on a bag of 3d points. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2010, San Francisco, CA, USA, 13-18 June, 2010*, pages 9–14. IEEE Computer Society, 2010.

Dima Litvak, Yaniv Zigel, and Israel Gannot. Fall detection of elderly through floor vibrations and sound. In *2008 30th annual international conference of the IEEE engineering in medicine and biology society*, pages 4632–4635. IEEE, 2008.

Bangli Liu, Haibin Cai, Zhaojie Ju, and Honghai Liu. RGB-D sensing based human action and interaction analysis: A survey. *Pattern Recognition*, 94: 1–12, 2019.

Zaigham Mahmood. *Guide to Ambient Intelligence in the IoT Environment: Principles, Technologies and Applications*. Springer, 2019.

Paul J McCullagh and Juan Carlos Augusto. The internet of things: The potential to facilitate health and wellness. *CEPIS Upgrade*, 12(1):59–68, 2011.

R. Minhas, A. A. Mohammed, and Q. M. J. Wu. Incremental learning in human action recognition based on snippets. *IEEE Transactions on Circuits and Systems for Video Technology*, 22(11):1529–1541, 2012.

Thi-Hoa-Cuc Nguyen, Jean-Christophe Nebel, and Francisco Flórez-Revuelta. Recognition of activities of daily living with egocentric vision: A review. *Sensors*, 16(1):72, 2016.

B. Ni, Y. Pei, P. Moulin, and S. Yan. Multilevel depth and image fusion for human activity detection. *IEEE Transactions on Cybernetics*, 43(5): 1383–1394, 2013.

Bingbing Ni, Pierre Moulin, and Shuicheng Yan. *Order-Preserving Sparse Coding for Sequence Classification*, pages 173–187. Springer Berlin Heidelberg, 2012.

Yannis Nikoloudakis, Spyridon Panagiotakis, Evangelos Markakis, Evangelos Pallis, George Mastorakis, Constantinos X Mavromoustakis, and Ciprian Dobre. A fog-based emergency system for smart enhanced living environments. *IEEE Cloud Computing*, (6):54–62, 2016.

Open Mobile Alliance. Oma device management tree and description. Technical report, Open Mobile Alliance Ltd., 2012.

Omar Oreifej and Zicheng Liu. HON4D: histogram of oriented 4d normals for activity recognition from depth sequences. In *2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013*, pages 716–723, 2013.

Paolo Palmieri, Luca Calderoni, and Dario Maio. Private inter-network routing for wireless sensor networks and the internet of things. In *Proceedings of the Computing Frontiers Conference, CF'17, Siena, Italy, May 15-17, 2017*, pages 396–401. ACM, 2017.

German Ignacio Parisi, Cornelius Weber, and Stefan Wermter. Self-organizing neural integration of pose-motion features for human action recognition. In *Front. Neurorobot.*, 2015.

Dan Partynski and Simon G. M. Koo. Integration of smart sensor networks into internet of things: Challenges and applications. In *2013 IEEE International Conference on Green Computing and Communications (GreenCom) and IEEE Internet of Things (iThings) and IEEE Cyber, Physical and Social Computing (CPSCom), Beijing, China, August 20-23, 2013*, pages 1162–1167. IEEE, 2013.

Ashish Patel and Jigarkumar Shah. Sensor-based activity recognition in the context of ambient assisted living systems: A review. *JAISE*, 11(4):301–322, 2019.

Soledad Pellicer, Guadalupe Santa, Andrés L. Bleda, Rafael Maestre, Antonio J. Jara, and Antonio Fernandez Gómez-Skarmeta. A global perspective of smart cities: A survey. In *Seventh International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, IMIS 2013, Taichung, Taiwan, July 3-5, 2013*, pages 439–444. IEEE Computer Society, 2013.

Marta Pinto, Mário Pereira, Diana Raposo, Marco Simões, and Miguel Castelo-Branco. Gameaal-an aal solution based on gamification and machine learning techniques. In *2019 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pages 1–4. IEEE, 2019.

L. Piyathilaka and S. Kodagoda. Gaussian mixture based hmm for human daily activity recognition using 3d skeleton features. In *2013 IEEE 8th Conference on Industrial Electronics and Applications (ICIEA)*, pages 567–572, 2013.

Ronald Poppe. A survey on vision-based human action recognition. *Image Vision Comput.*, 28(6):976–990, 2010.

J. Qi, Z. Wang, X. Lin, and C. Li. Learning complex spatio-temporal configurations of body joints for online activity recognition. *IEEE Transactions on Human-Machine Systems*, 48(6):637–647, 2018.

Hossein Rahmani, Arif Mahmood, Du Q. Huynh, and Ajmal S. Mian. HOPC: histogram of oriented principal components of 3d pointclouds for action recognition. In David J. Fleet, Tomás Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part II*, volume 8690 of *Lecture Notes in Computer Science*, pages 742–757. Springer, 2014.

Roni Ram, Francesco Furfari, Michele Girolami, Gema Ibañez-Sánchez, Juan-Pablo Lázaro-Ramos, Christopher Mayer, Barbara Prazak-Aram, and Tom Zentek. Universaal: provisioning platform for aal services. In *Ambient Intelligence-Software and Applications*, pages 105–112. Springer, 2013.

Parisa Rashidi and Alex Mihailidis. A survey on ambient-assisted living tools for older adults. *IEEE J. Biomedical and Health Informatics*, 17(3): 579–590, 2013.

Partha Pratim Ray. A survey of iot cloud platforms. *Future Computing and Informatics Journal*, 1(1-2):35–46, 2016.

Mohammad Abdur Razzaque, Marija Milojevic-Jevric, Andrei Palade, and Siobhán Clarke. Middleware for internet of things: A survey. *IEEE Internet of Things Journal*, 3(1):70–95, 2016.

K. K. Reddy, J. Liu, and M. Shah. Incremental action recognition using feature-tree. In *2009 IEEE 12th International Conference on Computer Vision*, pages 1010–1017, 2009.

Sergio Ricciardi, José Roberto Amazonas, Francesco Palmieri, and María

Bermúdez-Edo. Ambient intelligence in the internet of things. *Mobile Information Systems*, 2017:2878146:1–2878146:3, 2017.

Fabio Roli, Luca Didaci, and Gian Marcialis. Template co-update in multimodal biometric systems. pages 1194–1202, 08 2007.

Rocco De Rosa, Ilaria Gori, Fabio Cuzzolin, and Nicolò Cesa-Bianchi. Active incremental recognition of human activities in a streaming context. *Pattern Recognition Letters*, 99:48–56, 2017.

Samsung. SmartThings - The SmartThings Ecosystem.

Paul Scovanner, Saad Ali, and Mubarak Shah. A 3-dimensional sift descriptor and its application to action recognition. In *Proceedings of the 15th ACM International Conference on Multimedia*, MM '07, pages 357–360. ACM, 2007.

Amazon Web Service. AWS IoT Core.

Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. NTU RGB+D: A large scale dataset for 3d human activity analysis. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, pages 1010–1019. IEEE Computer Society, 2016.

J. Shan and S. Akella. 3d human action segmentation and recognition using pose kinetic energy. In *2014 IEEE International Workshop on Advanced Robotics and its Social Impacts*, pages 69–75, 2014.

Jamie Shotton, Andrew W. Fitzgibbon, Mat Cook, Toby Sharp, Mark Finocchio, Richard Moore, Alex Kipman, and Andrew Blake. Real-time human pose recognition in parts from single depth images. In *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, Colorado Springs, CO, USA, 20-25 June 2011*, pages 1297–1304. IEEE Computer Society, 2011.

Jamie Shotton, Toby Sharp, Alex Kipman, Andrew Fitzgibbon, Mark Finocchio, Andrew Blake, Mat Cook, and Richard Moore. Real-time human pose recognition in parts from single depth images. *Communications of the ACM*, 56(1):116–124, 2013.

Jae Mun Sim, Yonnim Lee, and Ohbyung Kwon. Acoustic sensor based recognition of human activity in everyday life for smart home services. *IJDSN*, 11:679123:1–679123:11, 2015.

Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 568–576, 2014.

Eugene Siow, Thanassis Tiropanis, and Wendy Hall. Analytics for the internet of things: A survey. *ACM Comput. Surv.*, 51(4):74:1–74:36, 2018.

SiteWhere. SiteWhere - The Open Platform for the Internet of Things.

Jaeyong Sung, Colin Ponce, Bart Selman, and Ashutosh Saxena. Human activity detection from rgbd images. In *Proceedings of the 16th AAAI Conference on Plan, Activity, and Intent Recognition*, AAAIWS'11-16, pages 47–55. AAAI Press, 2011.

Jaeyong Sung, C. Ponce, B. Selman, and A. Saxena. Unstructured human activity detection from rgbd images. In *2012 IEEE International Conference on Robotics and Automation*, pages 842–849, 2012.

Ilias Theodorakopoulos, Dimitris Kastaniotis, George Economou, and Spiros Fotopoulos. Pose-based human action recognition via sparse representation in dissimilarity space. *J. Visual Communication and Image Representation*, 25(1):12–23, 2014.

Can Tunca, Hande Özgür Alemdar, Halil Ertan, Özlem Durmaz Incel, and Cem Ersoy. Multimodal wireless sensor network-based ambient assisted

living in real homes with multiple residents. *Sensors*, 14(6):9692–9719, 2014.

Pavan K. Turaga, Rama Chellappa, V. S. Subrahmanian, and Octavian Udrea. Machine recognition of human activities: A survey. *IEEE Trans. Circuits Syst. Video Techn.*, 18(11):1473–1488, 2008.

DESA United Nations. World Population Ageing 2017, a.

DESA United Nations. World Population Prospects 2019, b.

Diego R. Faria Urbano Nunes and Paulo Peixoto. A human activity recognition framework using max-min features and key poses with differential evolution random forests classifier. *Pattern Recognition Letters*, 99:21–31, 2017.

Praneeth Vepakomma, Debraj De, Sajal K. Das, and Shekhar Bhansali. A-wristocracy: Deep learning on wrist-worn sensing for recognition of user complex activities. In *12th IEEE International Conference on Wearable and Implantable Body Sensor Networks, BSN 2015, Cambridge, MA, USA, June 9-12, 2015*, pages 1–6. IEEE, 2015.

Jie Wan, Xiang Gu, Liang Chen, and Jin Wang. Internet of things for ambient assisted living: Challenges and future opportunities. In *2017 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery, CyberC 2017, Nanjing, China, October 12-14, 2017*, pages 354–357. IEEE, 2017.

Heng Wang and Cordelia Schmid. Action recognition with improved trajectories. In *IEEE International Conference on Computer Vision*, Sydney, Australia, 2013.

Heng Wang, Muhammad Muneeb Ullah, Alexander Klaser, Ivan Laptev, and Cordelia Schmid. Evaluation of local spatio-temporal features for action recognition. In *BMVC 2009 - British Machine Vision Conference*, pages 124.1–124.11. BMVA Press, 2009.

J. Wang, Z. Liu, Y. Wu, and J. Yuan. Mining actionlet ensemble for action recognition with depth cameras. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1290–1297, 2012.

Jiang Wang, Zicheng Liu, and Ying Wu. *Learning Actionlet Ensemble for 3D Human Action Recognition*, pages 11–40. Springer International Publishing, 2014.

Jindong Wang, Yiqiang Chen, Shuji Hao, Xiaohui Peng, and Lisha Hu. Deep learning for sensor-based activity recognition: A survey. *Pattern Recognition Letters*, 119:3–11, 2019a.

Pichao Wang, Wanqing Li, Zhimin Gao, Jing Zhang, Chang Tang, and Philip O. Ogunbona. Action recognition from depth maps using deep convolutional neural networks. *IEEE Trans. Human-Machine Systems*, 46 (4):498–509, 2016.

Y. Wang and G. Mori. Hidden part models for human action recognition: Probabilistic versus max margin. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(7):1310–1323, 2011.

Yan Wang, Shuang Cang, and Hongnian Yu. A survey on wearable sensor modality centred human activity recognition in health care. *Expert Systems with Applications*, 137:167 – 190, 2019b.

Mark Weiser. The computer for the 21st century. *Mobile Computing and Communications Review*, 3(3):3–11, 1999.

Lu Xia and J. K. Aggarwal. Spatio-temporal depth cuboid similarity feature for activity recognition using depth camera. In *2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013*, pages 2834–2841, 2013.

Lu Xia, Chia-Chih Chen, and J. K. Aggarwal. View invariant human action recognition using histograms of 3d joints. In *CVPR Workshops*, pages 20–27. IEEE Computer Society, 2012.

Xiaodong Yang and Yingli Tian. Super normal vector for activity recognition using depth sequences. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, pages 804–811, 2014a.

Xiaodong Yang and YingLi Tian. Effective 3d action recognition using eigenjoints. *Journal of Visual Communication and Image Representation*, 25(1): 2 – 11, 2014b.

Koudai Yano, Yusuke Manabe, Masatsugu Hirano, Kohei Ishii, Mikio Deguchi, Takashi Yoshikawa, Takuro Sakiyama, and Katsuhito Yamasaki. Video-surveillance system for fall detection in the elderly. In *International Conference on Human-Computer Interaction*, pages 328–333. Springer, 2019.

L. Yeffet and L. Wolf. Local trinary patterns for human action recognition. In *2009 IEEE 12th International Conference on Computer Vision*, pages 492–497, 2009.

Kiat Seng Yeo, Mojy Curtis Chian, Tony Chon Wee Ng, and Anh-Tuan Do. Internet of things: Trends, challenges and applications. In *2014 International Symposium on Integrated Circuits (ISIC), Singapore, December 10-12, 2014*, pages 568–571. IEEE, 2014.

Alper Yilmaz and Mubarak Shah. Actions sketch: a novel action representation. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 984–989 vol. 1, 2005.

Chenyang Zhang and Yingli Tian. Rgb-d camera-based daily living activity recognition. *Journal of Computer Vision and Image Processing*, 2(4):12, 2012.

Shugang Zhang, Zhiqiang Wei, Jie Nie, Lei Huang, Shuang Wang, and Zhen Li. A review on human activity recognition using vision-based method. *Journal of healthcare engineering*, 2017, 2017.

Zhong Zhang, Christopher Conly, and Vassilis Athitsos. A survey on vision-based fall detection. In Fillia Makedon, editor, *Proceedings of the 8th ACM International Conference on PErvasive Technologies Related to Assistive Environments, PETRA 2015, Corfu, Greece, July 1-3, 2015*, pages 46:1–46:7. ACM, 2015.

Yu Zhu, Wenbin Chen, and Guodong Guo. Evaluating spatio-temporal interest point features for depth-based action recognition. *Image and Vision Computing*, 32(8):453 – 464, 2014.