Paul Lukowicz
Kai Kunze
Gerd Kortuem (Eds.)

LNCS 6446

Smart Sensing and Context

5th European Conference, EuroSSC 2010
Passau, Germany, November 2010
Proceedings

Springer

Lecture Notes in Computer Science 6446

Paul Lukowicz Kai Kunze Gerd Kortuem (Eds.)

Smart Sensing and Context

5th European Conference, EuroSSC 2010
Passau, Germany, November 14-16, 2010
Proceedings

Springer

Volume Editors

Paul Lukowicz
Universität Passau, IT-Zentrum/International House, Eingebettete Systeme
Innstraße 43, 94032, Passau, Germany
E-mail: paul.lukowicz@uni-passau.de

Kai Kunze
Universität Passau, IT-Zentrum/International House, Eingebettete Systeme
Innstraße 43, 94032, Passau, Germany
E-mail: kai.kunze@gmail.com

Gerd Kortuem
Lancaster University, InfoLab 21, Computing Department
Lancaster, LA1 4WA, UK
E-mail: kortuem@comp.lancs.ac.uk

Library of Congress Control Number: 2010938381

CR Subject Classification (1998): C.2, H.4, D.2, H.5, I.2, H.3

LNCS Sublibrary: SL 5 – Computer Communication Networks
and Telecommunications

ISSN 0302-9743
ISBN-10 3-642-16981-3 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-16981-6 Springer Berlin Heidelberg New York

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper 06/3180

Preface

Welcome to the 5th European Conference on Smart Sensing and Context. EuroSSC 2010 is a venue for high-quality papers that describe both original and unpublished research advancing the state of the art in smart surroundings, context-awareness and networked embedded sensor and actuator systems. The conference brings researchers of a variety of disciplines, perspectives and geographical areas together. It aims to explore the implications of computing as it starts to surround us transparently, interwoven into the fabric of everyday life.

Our Program Committee—composed of 28 leading international researchers in the fields of context recognition and smart sensing—accepted 13 papers to be published in these proceedings.

We want to take the opportunity to express our gratitude to the diligent efforts of all members of the EuroSSC Program Committee and the team for the local arrangements. We also want to thank the over 50 (co-)authors from over 13 countries for their contributions, shaping the EuroSSC into a true international venue.

In addition to the official conference proceedings, we are pleased to also include a short report about the Workshop on Ambient Assisted Living (AAL) platforms. We also want to express our thanks to the MonAMI coordinators, as without them the shared workshop would not have been possible. We found that the workshop sparked interesting discussions between the conference participants about the fascination topic of aging in a pervasive computing age.

Last but not least, we want to thank all the people who attended this year's EuroSSC conference, as the opportunity to meet and interact with interesting people in the field makes the planning of such events a rewarding endeavor for everybody involved in the organizational aspects of it.

November 2010

Paul Lukowicz
Kai Kunze
Gerd Kortuem

Organization

General Chair

Paul Lukowicz	Uni Passau, Germany

Program Chairs

Kai Kunze	Uni Passau, Germany
Gerd Kortuem	Lancaster University, UK

Local Arrangement Chair

Tobias Franke	Uni Passau, Germany

Publicity Chair

Clemens Lombriser	IBM Research, Switzerland

Programming Committee

DK Arvind	University of Edinburgh, UK
Martin Bauer	NEC Laboratories Europe, Germany
Xiang Chen	Institute for Infocomm Research, Singapore
Klaus David	University of Kassel, Germany
Elisabetta Farella	Bologna University, Italy
Ling Feng	Tsinghua University, China
Kaori Fujinami	Tokyo University of Agriculture and Technology, Japan
Hans Gellersen	Lancaster University, UK
Sandeep K.S. Gupa	Arizona State University, USA
Manfred Hauswirth	Digital Enterprise Research Institute (DERI), Ireland
Clemens Lombriser	IBM Research, Switzerland
Nirvana Meratnia	University of Twente, The Netherlands
Tatsuo Nakajima	Waseda University, Japan
Oscar Mayora Ibarra	CREATE-NET, Italy
Julia Kantorovitch	VTT, Technical Research Center of Finland, Finland
Marc Langheinrich	University of Lugano (USI), Switzerland

Santosh Pandey	Cisco Systems Inc., USA
Daniel Roggen	ETH Zurich, Switzerland
Kamran Sayrafian	National Institute of Standards & Technology, USA
Hong-Linh Truong	Vienna University of Technology, Austria
Kristof Van Laerhoven	TU Darmstadt, Germany

Table of Contents

Applications

Sensing

Systems Support

Higher Level Modeling

Report on the Workshop on ICT and Ageing: Announcing the European Initiative for an Ambient Assitant Living (AAL) Platform

FireGuide: A Context-Aware Fire Response Guide for the Building Occupants

Yuanping Li[1], Ling Feng[1], Lin Qiao[1], Yiping Li[1], Shoubin Kong[1], Yu Yi[1], Daqing Zhang[2], and Weijun Qin[3]

[1] Department of Computer Science & Technology, Tsinghua University, Beijing 10084, China
{liyp04@mails.,fengling@}.tsinghua.edu.cn
[2] Telecommunication Network and Services Department Institut TELECOM & Management, SudParis, France
Daqing.Zhang@it-sudparis.eu
[3] Institute of Software, Chinese Academy of Sciences, Beijing, China
qinweijun@is.iscas.ac.cn

Abstract. Context-awareness is a basic requirement of ubiquitous computing applications. While lots of good efforts have been made to deliver context-aware tour guide or museum guide, we ask ourselves: “can we build a context-aware fire guide to assist on-site fire victims to escape from a fire?” In reality, many people lose their lives in fire disasters due to bad judgment. Poor decisions are likely made in urgent situations. Timely and appropriate guidance is thus crucial to help people safely escape from the hazardous situations. So far, various pervasive computing techniques have been used to assist the firefighters in different aspects of their mission. However, not much research has been reported on assisting the occupants with personal and user-centric devices. In this paper, we present the idea of designing a context-aware fire response guide (FireGuide) for the building occupants from a technical perspective. By sensing the context of the building on fire and the occupants in the building, FireGuide advises either the fastest safe escape route or an action-list for “no-way-out” people. We evaluate the applicability of FireGuide through both user studies and experiments, which show that context-awareness in such a fire response guide can help improve the egress time. We also highlight the lessons we learn in designing such mission-critical context-aware applications in the paper.

Keywords: context-awareness, fire response guide, building occupants.

1 Introduction

1.1 Motivation

Fire hazards have been causing injuries and casualties. According to the statistics, there are 7-8 million fire disasters all over the world, which killed 70-80 thousand people and injured 500-800 thousand more annually [3]. The object of this study is to deliver a mission-critical context-aware fire response guide for the building occupants with the aim to calm and assist people to react properly when confronted with an fire disaster. Let us take a close look at two recent fire disasters, one in a college dormitory and the other in a club.

P. Lukowitz, G. Kortuem, and K. Kunze (Eds.): EuroSSC 2010, LNCS 6446, pp. 1–14, 2010.

[Case 1] On September 20, 2008, a fire occurred at Wuwang Club at Long Gang Street, Shenzhen, China. According to the police, the fire originated from the fireworks being set off on the stage. The disaster killed 44 people with 88 people being injured [19]. At the beginning of the fire, people in the club mistook it for a show. The fire quickly led to lots of smoke, and many people were smothered. As the customers were not familiar with the back door where the emergency exit is located, hundreds of people rushed to the front door through which they entered, causing many people being stumbled to the ground. In contrast, as the employees of the club knew the emergency exit, the majority of them succeeded in escaping from the fire via this exit gate. The death rate for the total 150 employees is far less than that for the club customers.

[Case 2] On November 14, 2008, at the Shanghai Business College in China, a dormitory room at the sixth floor caught fire at 6:10am in the morning. The fire expanded very quickly and gave off a lot of smoke. The door was incidentally closed. The four trapped girls fell back to try to keep away from the fire. Eventually, the fire flame spread to the balcony. The girls jumped from the balcony before the firefighters arrived. They may have thought it was safer to do so, yet none of them could survive. The police received the fire report at 6:12am. The firefighters soon arrived, and the fire was put out at 6:30am [10].

One harsh lesson we learn from these tragedies is that on-site occupants' assistance in an unexpected and urgent fire situation is critical. Suppose there had been a guide system, *which could direct nervous and chaotic people to the right emergency exit along with the proper escape path* (in case 1); or *which could adaptively advise the four trapped girls to use sheets tied to their body and climb down to the lower floors, e.g. the fifth floor, rather than directly jump down to the ground* (in case 2), their lives might have been saved.

1.2 Existing Solutions

Modern fire fighting systems. People have built a variety of excellent modern fire fighting systems, which can detect a fire, sound an alarm, and start corresponding activities to extinguish the fire. Such systematic facilities react quite well as a whole against a fire disaster. However, its assistance may not reach each individual on site. Take the above Wuwang club fire as an example (case 1). When the fire started, the fire fighting equipments, e.g., water showers, inside the club did begin to work. However, the turmoil still led to serious fire casualties. In fact, insufficient direction to individual occupants is a shortcoming of many modern fire fighting systems.

One recent work is based on dynamic identification lights, which were invented by Blohberger and Grundler [2]. The patent expressed the idea of dynamically activating the lights according to the situation, although the effectiveness of the system was not discussed.

Context-awareness in fire fighting. In the previous work on applying context-aware techniques to fire hazards, Jiang *et al.* presented a context-aware system called "Siren" to help professional firefighters. The system can support tacit communication between firefighters and support multi-channel communication [7]. Firefighters take PDAs both as a communication tool and a wireless sensing device. One firefighter can place the wireless-enabled sensors on the fly when he sizes up the situation. Wilson *et al.* proposed a new design of head-mounted displays for increased indoor firefighting safety and efficiency [20]. Luyten *et al.* designed a system to support fire brigades with a role-based focus+context user interface on PDAs and Tablet computers [11]. Klann described an approach of using game-like techniques to engage firemen from the Paris Fire Brigade into a participatory design process of wearable computing for emergency response [8]. Chen *et al.* described an application in helping emergency first responders in a fire exercise by the local police department and fire department at Lebanon airport (NH) in the spring of 2004 [5]. Landgren and Nulden analyzed the patterns of mobile interaction in the emergency response work [9]. Salam *et al.* evaluated a class of strategies to support efficient sensor energy workforce management used in mission-critical wireless sensor networks [14]. Chen *et al.* presented a map synchronization table strategy and map alternatives optimization for firefighters cooperative decision support in ad hoc solutions. Denef *et al.* did field study for design of new interactive technologies for firefighting [6]. Toups and Kerne also discussed developing education systems for teaching fire emergency responders [18].

While the previous research greatly extends the mission critical disaster applications of context-awareness techniques, they mainly focused on saving professionals' lives, such as firefighters or people involved in dangerous work. While we recognize that protecting professionals is very important, civilian occupants rather than professionals are the main casualties of disasters today. Occupants are different from professionals in some respects. For instance, we can have every professional wear certain electronic devices, but it is more difficult to have all the ordinary people wear electronic devices, although in some places this is possible if a certain regulation is set up for wearing such devices. Instead of posing requirements for ordinary people, we may turn to the ubiquitous computing support and let the environment wear electronic "eyes" and "mouths".

1.3 Our Work

The aim of this study is to investigate context-aware techniques for guiding individual occupant to safety in a fire from a technical perspective. The guide is mainly used before firefighters come, and the guide is not supposed to substitute existing firefighting systems but as a complementary user-centric technology that can help some people in the fire building.

We make the following assumptions in this study. 1) The layout of the building is available beforehand. 2) Fire locations can be detected by smoke and temperature sensors. 3) Building occupants may or may not be familiar with the structure of the building.

Upon the rise of a fire in a building, by sensing the context of the building on fire and the occupants in the building, FireGuide computes the fastest safe escape route or works out an action-list for "no-way-out" people. Our algorithm models a building in a graph, containing a number of vertices corresponding to doors, rooms, certain points at corridors, and exits. It then generates the escape route guide based on referential location (vertex in the graph). To achieve real-time response, FireGuide pre-computes all possible escape routes and store them in advance. In case of a fire, the fastest safe routes will be picked up immediately and delivered to the building occupants. Based on the locations of speakers, people in rooms, and “referential location”, FireGuide announces directional guide by means of lights and speakers without detecting the initial orientation of the people. For people who have mobile phones, a brief guide will also be delivered to their mobile phone screens, on the condition that their locations are available or can be detected, e.g. by bluetooth beacons.

We evaluate the empirical experiences, effectiveness, and efficiency of FireGuide by user studies and experiments, which show that context-awareness in such a fire response guide can help improve the egress time. We also pinpoint some lessons we learned in designing such mission-critical context-aware applications. To the best of our knowledge, this is the first effort to help individual victim in a fire using sensing and context-awareness technologies.

The remainder of the paper is organized as follows. We analyze solution requirements for the context-aware fire response guide FireGuide in Section 2, and describe the FireGuide architecture in Section 3. The techniques in FireGuide are discussed in Section 4. We report the performance study in Section 5, and conclude the paper in Section 6.

2 FireGuide Solution Requirements

Based on the analysis of some real fire disasters and the study of firefighting domain knowledge, we have the following criteria in designing FireGuide.

1) Timeliness. Fire usually expands in minutes to the whole building. Only 6 minutes after a fire starts, the near site temperature reaches about 300-400°C. At a fire site, people tend to be extremely sensitive to environmental and thermal radioactive temperature and can tolerate 65°C for a very limited time [16]. Thus, the FireGuide system should be efficient in decision making.

2) Reliability. As a mission-critical application, FireGuide must be reliable in terms of both hardware and software in fire emergencies. The underlying strategies should be as much reliable as possible. We once tried to incorporate several sophisticated fire expansion prediction models into the design of FireGuide, especially one of the leading fire simulation systems called FDS [1]. However, as such a system needs a comprehensive model of the building (including furniture, window status, wall materials, etc.), it takes hours to compute a simulated fire expansion result. More important, solid validation is still lacking for the precision of prediction in the real world, since in reality there are too many details to be modelled. In other words, we cannot predict exactly where the fire will expand in the next minute. Therefore, we mainly use the real-time sensor data in the FireGuide system.

3) Typical Building Models. As fire situations are complex, the building type greatly influences the design of FireGuide system. As a first step, we consider two typical building models, taking the former Shanghai College Dormitory Building and Shenzhen Club Building for reference. The first building has a regular-sized small room construction, while the second represents a large public room, such as a restaurant, a supermarket, a club, etc., where customers are relatively unfamiliar with the building. We can extend the developed technologies to more types of buildings later.

4) Simple Interaction. Since the people might be in a nervous state, the system should avoid involving complex interactions and emphasize some simple but effective information.

3 FireGuide Architecture

FireGuide works in a building, where each room has a smoke sensor at the middle of the ceiling, and another sensor, which can be either smoke or temperature sensor, at the entrance of the room. In addition, smoke sensors are attached on the corridors and exits of the whole building. Each room and exit has a speaker. Emergency red lights are also installed at each exit and along each corridor. In the prototype hardware deployment of FireGuide, smoke and temperature sensors are connected to the FireGuide system server via data acquisition cards. Multiple sensor data acquisition methods can be supported.

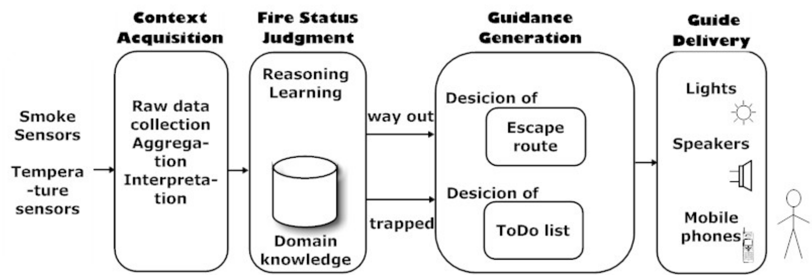


Fig. 1. Architecture of the FireGuide

Figure 1 depicts the software architecture of the system. It contains four major components, namely, *context acquisition, fire status judgment, guidance generation*, and *guidance delivery*. In response to a fire, the *context acquisition* module quickly gathers low-leveled raw sensor values, and performs context aggregation and interpretation according to the building and fire related ontologies, which is manipulated using the prot*égé* API [1]. These ontologies are used to model the devices, building structure, and their relationships. Based on the derived high-level context information and the domain knowledge, the *fire status judgment* decides the pattern of expansion and extinguishing of the fire in the fire building. The change of the fire status triggers the update of the occupants' situations in the system and influences the most appropriate fire response manner for them. There are basically two choices for occupants: 1) finding the shortest safe way out; or 2) taking proper self-protection actions if trapped. The *guidance generation* then works out the escape route under choice 1), or the ToDo list for those who are trapped under choice 2). Implementation of the guidance generation module will be detailed in the next section. In order to provide different guides to people at different locations, the *guidance delivery* module support multiple audio cards and different output communication protocols. Fire victims are notified via nearby speakers and wearable mobile phones, whose aims are to deliver the escape route if it exists and safe response guide. Red lights along the escape route and exits will be set on as well.

[1] http://protege.stanford.edu/

4 FireGuide Guidance Generation

FireGuide delivers two types of guidance for two types of fire victims, i.e., ones who can escape and ones who are trapped.

4.1 Escape Route Generation

The emergency evacuation route search algorithms have been already studied in [15,17]. Based on that research, we further designed the method to generate the guide output.

Firefighting experiences tell us the right way to escape from the fire is via emergency exits, which could be the inner or outer stairs for a tall building. The guidance using the command "Go west!" may be hard for the occupants to accurately use, particularly in an unfamiliar building. Also, if the system guides the occupants to "Go to the stairs!" directly, the stairs might be too far away to see. Hence, we adopt a near *referential vertex* approach, which avoid detecting people's orientation, to give the route guide. "Go through the door, turn left!" is one such guidance example. For a person in a room, "Go through the door" tells the orientation which s/he is supposed to be.

We developed a GUI tool to assist the FireGuide administrator to off-line annotate the floor plan of the building, and then this annotated floor plan is converted into a graph model by the program, as shown in Figure 2(a). Here, circles stand for ordinary vertices and the boxes stand for referential vertices. Two heuristics are employed to identify referential vertices.

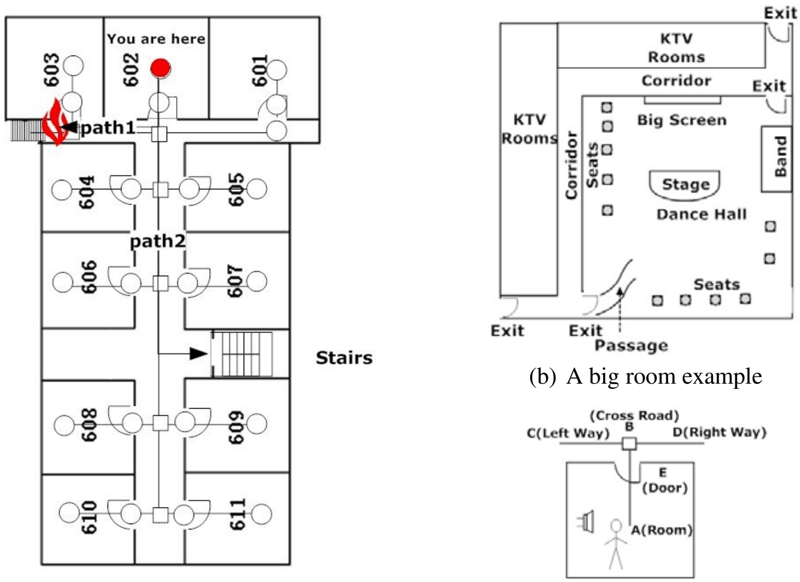
- Crossings in the path can be taken as referential vertices. This is judged by whether the out-degree of a vertex is greater than two. For example, in Figure 2(c), A is a room, E is its door, and B is a crossing out of the room. Taking B as referential vertex for E, we can go "left" or "right" at this vertex for the person in the room.
- The referential vertices can be obtained by user-defined rules, e.g., all the big screens on the wall are treated as referential vertices.

To obtain the correct way at run-time, FireGuide computes the shortest safe path between every two vertices in the graph and stores them in advance. In case of a fire, dangerous vertices will be detected. If the fire does block the shortest path between the two vertices, FireGuide will re-compute the shortest path. For example, in Figure 2(a), if we compute the route when there is no fire, the shortest route from "*You are here*" to the exit will be path 1. However, when a fire occurs nearby, the stair vertex is blocked, so the shortest safe route becomes path 2. We increase the weight of edges connected to the unsafe vertices to make the computed path pass around unsafe vertices.

However, using the shortest safe path algorithm directly turns out poor system scalability, making it difficult to meet the real-time response requirement in large buildings. We made further special optimizations in the algorithm, greatly decreasing the time complexity and making the final performance of FireGuide acceptable, as evidenced by our experiments.

1) We initialize all the shortest paths for each vertex offline. In the run-time period, only a subset of the paths will be recomputed online according to the context.

2) If a fire occurs at or expands to stairs, we leverage the structural characteristics of the building. That is, in a vertical staircase, if one location of the staircase is blocked



(a) Conversion from an annotated floor plan into a graph

(b) A big room example

(c) Referential vertex selection example

Fig. 2. The models for guide generation

by the fire, FireGuide will not guide people to pass down through the firing stairs in the upper floors. This optimization reduces the complexity from the 3-dimension graph model to an approximate 2-dimension graph model.

Another issue that needs to be tackled in the "referential vertex" method is to explain directions in natural language. Let $A(x_A, y_A, z_A)$ and $C(x_C, y_C, z_C)$ are different locations, $B(x_B, y_B, z_B)$ is the referential vertex from A to C. Let vector $a = B - A = (a_x, a_y, a_z)$ and $b = C - B = (b_x, b_y, b_z)$. $P_c = a_x b_y - a_y b_x$, $P_i = a_x b_x + a_y b_y$ and $r = \sqrt{a_x a_x + a_y a_y} \cdot \sqrt{b_x b_x + b_y b_y}$. Then the direction D is decomposed into three elements (d_1, d_2, d_3)

$$d_1 = \begin{cases} \text{left} & P_c > 0 \text{ and } |\frac{P_c}{r}| > \varepsilon_1 \\ \text{right} & P_c < 0 \text{ and } |\frac{P_c}{r}| > \varepsilon_1 \\ \text{center} & |\frac{P_c}{r}| \leq \varepsilon_1 \end{cases} \quad (1) \qquad d_2 = \begin{cases} \text{forward} & P_i > 0 \text{ and } |\frac{P_i}{r}| > \varepsilon_2 \\ \text{backward} & P_i < 0 \text{ and } |\frac{P_i}{r}| > \varepsilon_2 \\ \text{center} & |\frac{P_i}{r}| \leq \varepsilon_2 \end{cases} \quad (2)$$

where ε_1 and ε_2 are the thresholds, $\varepsilon_1 \in [0,1)$, and $\varepsilon_2 \in [0,1)$.

$$d_3 = \begin{cases} \text{up} & z_C > z_B \\ \text{down} & z_C < z_B \\ \text{center} & z_C = z_B \end{cases} \quad (3)$$

With all the methods stated above, FireGuide generates guide information in the natural language such as "Go through the door, turn right!" to the person in the room, in Figure 2(c). An algorithm description of a typical transformation from an evacuation path to guide speech is shown in Algorithm 1.

```
Input  : A path P that starts from o1 to o2, the layout graph G(V,E)
Output: Guide speech for the path P
foreach vertex v in P from o1 to o2 do
    if v has more than 2 adjacent vertices in G then
        v2 = v;
        break;
v3 = v2's next vertex in the path P;
v1 = v2's previous vertex in the path P;
direction = get the direction from v1 to v3 by referential v2 ;
if v2 is a door then
    if direction is "forward" then
        speech = "go through" + v2's name + ", go forward";
    else
        speech = "go through" + v2's name + ", turn " + direction;
else if  v2 is a corridor then
    if direction is "forward" then
        speech = "In " + v2's name + ", go forward";
    else
        speech = "In " + v2's name + ", turn " + direction;
```

Algorithm 1. A typical transformation from an evacuation path to guide speech

In addition, in a big room like a dance hall (Figure 2(b)), there may be more than one exit, and occupants might not see all the exits. To attract the occupants to the safe exits, we add speakers at the exits, repeating “Here is the exit!”

4.2 ToDo-List Generation for “No-Way-Out” Ones

Our current study is based on the following two rules to determine whether one is trapped by a fire.

Rule 1: *the door of the room where one is situated has been blocked by the fire.*
Rule 2: *all the paths from which one can escape out of the floor have been blocked by the fire.*

Situations complying with Rule 1 could be detected by either a temperature sensor or a smoke sensor close to the door, while those complying with Rule 2 could be detected via smoke sensors. A ToDo list advised by the FireGuide system for trapped ones includes the following:

Action 1: *Tie knotted sheets to the window/balcony and your body, climb downward!*
Action 2: *Close the door and keep smoke out!*
Action 3: *Ask for help!*
Action 4: *Crouching low or crawling along the wall!*

5 Evaluation

We implemented FireGuide using Java and C++ programming languages. C++ code is used for sensors and speakers, which are called by the Java Native Interface (JNI). The FireGuide server in the evaluation is a computer of 3GHz CPU and 2GB RAM. We conducted the evaluation of effectiveness and efficiency of FireGuide through user studies and experiments.

5.1 Interviews with the Domain Experts and the People Who Experienced Real Fires

We interviewed the firefighting administration department in the university. It is a subordinate division of the Public Safety Ministry. They gave us a brochure of fire response principles which were summarized by experienced firefighters and researchers. These principles were implemented in FireGuide, especially in the ToDo list.

We interviewed three senior experts in a professional firefighting system manufacturer. They acknowledged the usefulness of the system, and suggested that the system is suited for the places like hotels and dormitories. One of the experts is the member of the firefighting product standardization committee of the country. He emphasized the hazard of smoke inhalation, and suggested, if possible, guide people to put on smoke-proof masks. This suggestion was implemented as general information delivered in FireGuide, besides the adaptive route or ToDo list guide.

We interviewed two people who experienced real fires. One of them said that the particles in the smoke were large. The smoke burned the nose badly. When the fire started, almost all the people around the fire were in a sudden confusion. Some people even used a cotton quilt to put out the fire, but the fire grew even faster. A system like FireGuide is in demand to help the people protect themselves. The other person said, in the real fire which he experienced, all people were in a panic and ran and collided in the passageway chaotically. People do need a guide to reduce the collision so as to reduce the number of the injuries.

Reflection: The interviews above indirectly identified the requirement for FireGuide. To obtain an on-site experience, two of the lab members participated in one fire drill in a 14-floor dormitory building organized by the firefighting department. Hundreds of people took part in the drill. The experiences told us that when we were at the very stage of deciding where to go at the door, the flashlights which show the arrows and diagrams marking exit routes, were not always in font of us. Whichever way we selected, we were not confident. FireGuide does not mean to substitute the flashlights, but it will be useful if it can help people to encounter the right flashlights more quickly.

5.2 Empirical Experience with FireGuide

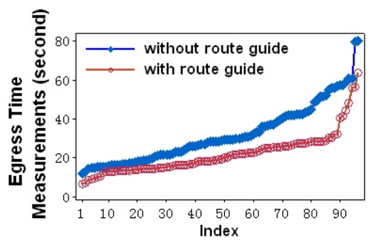
We conducted a empirical experience study with the FireGuide system. 20 subjects, including 3 university staff, 3 administrative staff, and 14 graduate students, participated in this study.

Method: Before the study, the subjects were told that one kind of emergency might happen, and we were to see whether they could behave correctly. But we did not tell them what would actually happen. We designed two scenarios one by one. The first scenario was to guide them to a safe exit. But this safe exit was not the one through which people entered the building. The second scenario was to simulate a situation that they are trapped by the fire outside near the door. The system guided them to close the door and wait for rescue. After the test, the subjects were asked to fill in a questionnaire.

Result: Except for one invalid questionnaire which selected more than one item for one question, we collected 19 completed valid questionnaires, whose results are tabulated in Figure 3(a) (Likert scale,1-5 point, 5 for best and 1 for worst). Overall, the system was perceived as useful (4.33) and had almost no adverse interference with people's normal life (4.64). People scored high on usefulness of voice guiding (4.14) and the clarity of short message guide on mobile phones (4.41). The reaction time of voice guiding was fine (4.11). However, the message receiving time was complained by some subjects (3.81). The mobile phones are provided by us, and the delay of short messages was compared to the voice guide.

Empirical Experience	*Mean*	*Standard Deviation*
System usefulness	4.33	0.60
Usefulness of voice guiding	4.14	0.78
Reaction time of voice guiding	4.11	0.93
Usefulness of short message guiding	3.92	0.84
Clarity effect of short message guiding	4.41	0.79
Reaction time of the system	3.81	0.69
Unobtrusiveness in people's normal life	4.64	0.61

(a) Users' experiences with FireGuide



(b) Egress time measurements for voice alarms with/without route guide

Fig. 3. The user study results

Reflection: 1) Benefits and constraints of using mobile phones. Using mobile phones are better than nothing. We want the victims to leave or take other self-protection actions before firemen come. People can adjust the position of mobile phone by themselves to see the guide more clearly. The mobile phone can use the public wireless communication channels which are not destroyed by the fire. However, there are some constraints with mobile phones. Firstly, not all occupants have access to mobile phones. Secondly, a slow message receiving time was incurred when we adopted the public mobile network to connect the system to users' mobile phones. A more efficient method might be using the combination of both public and local networks. Thirdly, it is still an open question that whether the mobile phone is useful in a very crowded building, where watching mobile phone might become difficult for users. Displaying very simple guide, such as only a directional arrow on the screen with the aid of the compass in the mobile phone, might be a good choice. 2) Crowd, noise and people's communications. In this user study. We observed that people communicated with each other when they were in a small crowd. Even if some people did not catch the guide, they could follow those who catch the guide. The people who had mobile phones could also help those who did not have. But the crowd also impeded the speed of evacuation. These phenomena deserve further researches (There has already been research on detecting group formations from wearable acceleration sensors [21]).

5.3 Effectiveness of FireGuide

To investigate more rigorously whether context-awareness in the guide takes effect in an emergent state, we performed this user study.

One challenge in designing such an experiment lies in designing an emergency state, because we cannot put people into real danger. Previous researches on disaster rescue and evacuation were often carried out in simulation environments [13,4]. Hence, our user study was also conducted as a drill. We referred to the famous bonus method successfully used by Helen Muir's group at Cranfield in the domain of aircraft evacuations [13]. In their pioneering aircraft egress experiments, bonus was used to simulate an emergent state among the subjects. Another challenge is that, as in reality, people's attention is not always on the emergency. To make the subjects' attentions not fully on fire alarms, we let the subjects focus on some work in the experiment, i.e., doing some mathematical exercises.

Method: 24 subjects, 8 female and 16 male of age 19-48, participated in the experiment. They were recruited through an open internet advertisement in the city and were arranged into 6 groups randomly, with 4 subjects per group. Each group was allocated to 2 rooms, with 2 subjects per room. Each group attended 8 tests. To eliminate the effect of position difference, the 4 subjects changed their positions clock-wise in every test, in turn. As soon as a voice alarm arose, the subjects at different rooms needed to evacuate to a safe place, i.e., one of the two stairs at the two ends of the corridor.

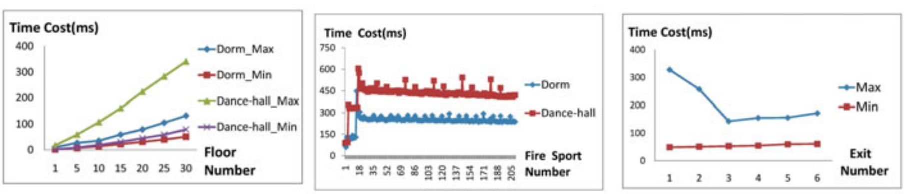
To investigate the effectiveness of fire route guide, we consider two types of voice alarms, one with route guide "*The building is on fire, please cover your mouth with wet cloth, in the corridor turn right/left, go to the safe exit quickly!*", and the other without route guide "*The building is on fire, please cover your mouth with wet cloth!*". The same voice alarm is repeated in one test until the test is finished. In each test, one of the two stairs was blocked by a "fire" randomly. When a subject reached the "fire" stairs, a staff would notify her/him about being blocked by fire, and the subject then needed to turn around, and run to the opposite side, i.e., the safe stairs. Which type of voice alarm would arise in one test was a random choice. The experiment took place in a real building, and the distances from the two rooms to the two stairs were not equal. This might have influenced the egress time. Therefore, for each kind of voice alarms, we let the number of "fire" on the left stairs be equal to the number of "fire" on the right stairs. We recorded both the egress time and the mathematical exercise score for each subject. The bonus was divided into 3 categories for a competitive atmosphere. (1) If a subject obtains top 50% exercise score and also is among top 50% to arrive at a safe place, s/he will get 40 RMB (Yuan) bonus; (2) If a subject does not obtain top 50% exercise score but is among top 50% to arrive at a safe place, s/he will get 30 RMB (Yuan) bonus; (3) The rest will get 20 RMB (Yuan) bonus.

Result: In total, 96 measurements of egress time for voice route guide and 96 measurements for voice without route guide were noted down. The ranked measurements are shown in Figure 3(b). The Wilcoxon-Mann-Whitney statistic [12] was used to do the inference test. Let f be the continuous cumulative distribution function for egress time e_1 without route guide, and g be the continuous cumulative distribution function for egress time e_2 with route guide. We examined the hypothesis $H_0 : f = g$ against the alternative $H_1 : f(x) > g(x)$ for every x (i.e., e_1 is statistically larger than e_2). The statistic U is 2516.5 and z is 5.43. The statistics reject H_0 in favor of H_1 with the significance level $\alpha < 0.001$. The result supports a first but valuable finding, that is, the voice alarm with context-aware route guide leads to less egress time than the voice alarm without route guide in a similar building layout statistically, although we observed not all the subjects followed the guide exactly in this experiment.

Reflection: 1) Building type. It is noteworthy that this user study was conducted in an office building. The result is for the situation when the fire is just beginning and the crowd is relatively small. This might not be suitable for very crowded club. 2) Equipment failure. In this study, the result was obtained in a situation where the equipments, e.g. the sensors, or at least some of them work. Equipment failure should be considered for the system deployment. For example, reserve battery, fire proof materials should be used to protect the equipments. Actually, the influence of equipment failure and how to reduce such a negative influence deserve further researches.

5.4 Efficiency of FireGuide

As context judgment, path finding, and ToDo-list generation constitutes the inner kernels of the system, while context sensing and delivery time are beyond the control of the system, we evaluate the system *reaction time*, which is the time interval when a smoke sensor value is read till FireGuide guidance is issued to the user in a building.



(a)Performance when the total number of floors varies (*exit_num_per_floor*=2)

(b)Performance when the fire spot number varies(*exit_num_per_floor*=2, *floor_num*=30)

(c)Performance when the total exit number varies (*floor_num*=30)

Fig. 4. Performance study

We simulated on the two real building models, i.e., Shanghai Business College Dormitory and Shenzhen Wuwang Club , where the former has 11 rooms per floor and the later has 7 small rooms and 1 big room (dance hall) per floor. Then we varied the floor number, exit number, and fire spot number of the buildings to examine the system performance. The experimental result showed that all the response times are less than 1 second (Figure 4).

6 Conclusion and Outlook

In this paper, we report our technical design of a context-aware fire response guide which can guide possibly panicked occupants to properly react to an emergent fire, evaluate its performance through user studies and experiments. The experimental results show the feasibility, effectiveness and some limitations of the system. To have the system truly work in the real world, there exist some other issues that deserve further consideration. First, besides users' mobile phones, public displays in buildings can be used as complementary interfaces. Second, although the shortest path solution is presented to direct fire victims to different exits, finding the optimal guiding strategy still remains challenging, for example, when some exits are blocked by the crowd or collapsed by secondary effects from the fire. Bottlenecks may arise if leading building occupants to exploit the same escape routes. In addition to sensor data, some other factors, like crowd density detection and phased evacuation, can be incorporated in computing appropriate escape routes.

Acknowledgements

We thank the support from National Natural Science Foundation of China (60773156), Chinese National 863 Advanced Technology Program (2008AA01Z132), Research Fund for the Doctoral Program of Chinese Higher Education (20070003089), and Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry. We would also thank Jianwen Chen, Xiang Li, Hao Wang, Tangjian Deng, Heng Zhang, Jiping Zheng , Professor Lizhu Zhou, Scott Simmons and Shanghai Huyi firefighting electronic equipment company for the help in the experiments and useful comments.

References

1. Fire dynamics simulator (FDS), http://www.ohloh.net/p/4709 (accessed April 8, 2009)
2. Blohberger, F., Grundler, G.: Evacuation system having escape identification lights. United States Patent Application 20090102619 (2009)
3. Brushlinsky, N., Sokolov, S., Wagner, I.P., Hall, J.: World fire statistics (2006), http://ec.europa.eu/consumers/cons_safe/presentations/21-02/ctif.pdf (accessed April 13, 2009)
4. Burke, J., Murphy, R., Coovert, M., Riddle, D.: Moonlight in Miami: Field study of human-robot interaction in the context of an urban search and rescue disaster response training exercise. Human-Computer Interaction 19(1), 85–116 (2004)
5. Chen, G., Li, M., Kotz, D.: Data-centric middleware for context-aware pervasive computing. Pervasive and Mobile Computing, pp. 216–253 (2007)
6. Denef, S., Ramirez, L., Dyrks, T.: Letting tools talk: interactive technology for firefighting. In: Proceedings of the 27th International Conference on Human Factors in Computing Systems, pp. 4447–4452 (2009)
7. Jiang, X., Chen, N., Hong, J., Wang, K., Takayama, L., Landay, J.: Siren: Context-aware computing for firefighting. In: Ferscha, A., Mattern, F. (eds.) PERVASIVE 2004. LNCS, vol. 3001, pp. 87–105. Springer, Heidelberg (2004)
8. Klann, M.: Playing with Fire: User-Centered Design of Wearable Computing for Emergency Response. In: Löffler, J., Klann, M. (eds.) Mobile Response 2007. LNCS, vol. 4458, pp. 116–125. Springer, Heidelberg (2007)
9. Landgren, J., Nulden, U.: A study of emergency response work: patterns of mobile phone interaction. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 1323–1332. ACM, New York (2007)

10. Liu, D.: Shanghai business college dormitory fire: four students dead, http://news.xinhuanet.com/newscenter/2008-11/14/content_10357255_1.htm (accessed July 27, 2009)
11. Luyten, K., Winters, F., Coninx, K., Naudts, D., Moerman, I.: A situation-aware mobile system to support fire brigades in emergency situations. In: Meersman, R., Tari, Z., Herrero, P. (eds.) OTM 2006 Workshops. LNCS, vol. 4278, pp. 1966–1975. Springer, Heidelberg (2006)
12. Mann, H., Whitney, D.: On a test of whether one of two random variables is stochastically larger than the other. In: The Annals of Mathematical Statistics, pp. 50–60 (1947)
13. Muir, H., Bottomley, D., Marrison, C.: Effects of motivation and cabin configuration on emergency aircraft evacuation behavior and rates of egress. The International Journal of Aviation Psychology 6(1), 57–77 (1996)
14. Salam, H.A., Rizvi, S., Ainsworth, S., Olariu, S.: A durable sensor enabled lifeline support for firefighters. In: IEEE Conference on Computer Communications Workshops INFOCOM, pp. 1–6 (2008)
15. Shastri, J.: Safe Navigation during Fire Hazards using Specknets. Master's thesis, The University of Edinburgh (2006)
16. Song, G.: Firefighting in public places. Chinese people's public security university press (2002)
17. Tabirca, T., Brown, K., Sreenan, C.: A Dynamic Model for Fire Emergency Evacuation Based on Wireless Sensor Networks. In: Proceedings of the Eighth International Symposium on Parallel and Distributed Computing, pp. 29–36. IEEE Computer Society, Los Alamitos (2009)
18. Toups, Z., Kerne, A.: Implicit coordination in firefighting practice: Design implications for teaching fire emergency responders. In: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 707–716. ACM, New York (2007)
19. Wang, P., Ying, A., Jiang, B.: Analysis of the fire caused heavy casualties in shenzhen: the fire was mistaken as a show, http://news.sina.com.cn/c/2008-09-22/121716333822.shtml (accessed November 16, 2008)
20. Wilson, J., Wright, P.: Head-mounted display efficacy study to aid first responder indoor navigation. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science 223(3), 675–688 (2009)
21. Wirz, M., Roggen, D., Tröster, G.: Decentralized detection of group formations from wearable acceleration sensors. In: Proceedings of 2009 IEEE Int. Conf. on Social Computing, pp. 952–959 (2009)

How to Log Sleeping Trends? A Case Study on the Long-Term Capturing of User Data

Holger Becker, Marko Borazio, and Kristof Van Laerhoven

Embedded Sensing Systems, TU Darmstadt,
Hochschulstr. 10, 64289, Darmstadt Germany
{h.becker,borazio,kristof}@ess.tu-darmstadt.de
http://www.ess.tu-darmstadt.de

Abstract. Designing and installing long-term monitoring equipment in the users' home sphere often presents challenges in terms of reliability, privacy, and deployment. Taking the logging of sleeping postures as an example, this study examines data from two very different modalities, high-fidelity video footage and logged wrist acceleration, that were chosen for their ease of setting up and deployability for a sustained period. An analysis shows the deployment challenges of both, as well as what can be achieved in terms of detection accuracy and privacy. Finally, we evaluate the benefits that a combination of both modalities would bring.

Keywords: actigraphy, sleep postures, long-term monitoring.

1 Introduction

Long-term monitoring of users at home has been the focus of many recent studies. The applications that are argued for in research often tend to be medical in nature, such as the support for elderly care with systems that observe activity, fitness, and distinct events like falls. Other work targets a wider audience with context-aware home environments that display and activate services in appropriate situations, such as ubiquitous displays [8] or smart appliances [10].

Sleep posture logging and analysis is part of an emergent field in sleep monitoring research, with patients in professional sleeping laboratories being monitored by infrared cameras and a chest-worn tilt measurement unit. Sleep postures has been identified as a key factor in a variety of scenarios, including personality evaluation [2] and obstructive sleep apnea [9]. This scenario will be used here as a case study for long-term domestic monitoring systems, where the sleeping postures of the user will be detected by a system that remains running for a long period, from weeks to years of continuous operation.

The emphasis in this paper is put on how the modality of the sensed data affects the performance of a *long-running* system that is supposed to monitor the user in her *domestic* environment. This set of broad requirements tends to bring in several challenges that researchers are facing when deploying such systems.

P. Lukowitz, G. Kortuem, and K. Kunze (Eds.): EuroSSC 2010, LNCS 6446, pp. 15–27, 2010.

1.1 Challenges

We argue that three challenges in particular are important to observe in long-term deployments in domestic environments. These three will be used to frame our evaluation and discussion in the remainder of the paper:

Reliability is in general a concern for most monitoring systems; for those that are supposed to run over longer time spans (from months to even years), this is even more of a challenge. The system not only has to keep running, it also has to perform and keep an accurate detection of the phenomena to be monitored. The presence of noise and irrelevant data, as well as limitations on the sensor's behalf, mean that a perfect capturing of the phenomena is often impossible to attain out in real-world deployments. This for instance will become clear in our sleep case study, where direct video footage is often covered and body-worn sensors cannot be worn in the most optimal places (chest or back).

Privacy is another factor that becomes pressing as larger sections of our lives are recorded. As the sensors can pick up more detail and cover more in both time and space, their data become more sensitive and should be safeguarded, and designing the system should very much take this into account. For the case study of sleep posture logging, the sensor could automatically record and reveal any activities related to the user's bedroom environment, which is together with the bathroom a most sensitive place in the home [5].

Deployment of the monitoring system needs to be easy and modular, so little time is spent on installation and moving the system from one environment to another does not bring along a costly installation procedure. Dependability and usability of the system also belong in this category (e.g., not having to frequently reboot or maintain the system, or avoiding failures in critical medical applications). In our case study, the recording system needs to remain active for at least several hours per day, and this without intervening much in the user's daily life (as frequent battery changes or system interaction would cause).

1.2 High-Fidelity versus Low-Fidelity Sensors

The two types of modalities that are studied in this paper also emphasize a trade-off between good sensor modalities from the point of views related to a detection and monitoring, privacy, and reliability.

The stronger, high-fidelity sensors tend to produce rich information at a high rate so that monitored phenomena are captured in detail. As a result, high accuracy, precision, and recall figures are feasible. Often, maintaining this system is harder and the data is, especially in home environments, sensitive in nature. Video footage of the user's bedroom is by far the best sensor data to observe sleeping postures, but as it is also placed in one of the most sensitive environments, few users would actually permit installation. Keeping a video system running also tends to require more maintenance and bulkier hardware.

The weaker, low-fidelity sensors on the other hand often are associated with less privacy concerns. This comes at the price of less detailed data being captured however, with a significant amount of uncertainty being present for the analysis algorithms to handle. These sensors produce less data that are harder to mine.

1.3 Paper Overview

The contributions of this paper are threefold. An evaluation of all feasible scenarios is carried out to find the most workable solution to capture, process, and log images from sleeping postures. Two studies then investigate how well both types of sensors on their own would work in terms of accuracy when deployed in a domestic setting, and assuming they are trained by the user. Finally, a combination of both modalities is proposed and an experiment is described to indicate under which parameters this combination would result in a better system.

This paper is structured as follows: First, the case study of sleep posture detection and logging is presented as an illustration for the type of long-term domestic monitoring task we pursue in this paper. Then, video footage and wrist acceleration are presented and analyzed on their individual merit for the purpose of sleep posture monitoring. Section 5 will then offer the combination of these two as a better alternative and studies associated trade-offs. The last section will finally present our conclusions and point to future and ongoing work.

2 Case Study: Sustained Logging of Sleep Postures

The posture in which we sleep can have a large effect on the quality of sleep, with research [4] looking at sleep postures as one of the main pieces of information to record in studies. Patients with obstructive sleep apnea, a sleep disorder characterized by pauses in breathing during sleep, should avoid sleeping on the back, and are strongly encouraged by sleeping laterally (on one's side). Other research [7] correlates usual sleep postures with a person's personality profile.

There are several types of postures that are of interest in the aforementioned studies: Medical articles mostly investigates the *lateral* (lying on the left or right side), *supine* (lying on back), and *prone* (lying on chest) sleep postures. Other studies take a more detailed look at the full body and follow a different naming convention for common postures such as *foetus*, *soldier*, or *starfish*. Some examples for both are depicted in figure 1.

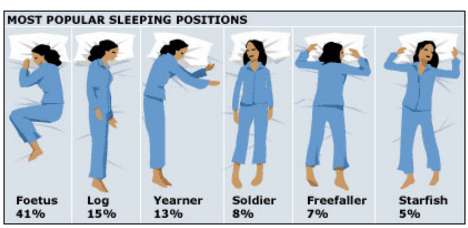


Fig. 1. Some common sleeping postures (reprinted from [2] with permission from Prof. Idzikowski from the Edinburgh Sleep Centre.)

Sleep postures are therefore not only important indicators to watch for a typical one-to-two-night study, having a longer record would mean that trends could be observed and correlated to daily activities and habits. In the following sections we will discuss two methods to capture these sleep posture trends.

3 High-Fidelity Data: IR Video Footage

Videometry is a fixed part of every cardiorespiratory polysomnography, which is the central measuring method of stationary diagnostics in sleep laboratory [11]. However, in most cases the posture is measured by tilt sensors, which are worn on the chest. The data from the night vision camera is used redundantly to monitor the patient. Although this is currently not done, this video material could be used to automatically detect the posture of the patients.

The reliability of performing (semi-) automatic posture analysis can be expected to be quite high. The resolution and camera angle can be optimally chosen for covering the user's full body, and state-of-the-art cameras are widely available with built-in IR lighting that is strong enough to reach an entire bed. The straightforward approach taken in this paper is based on having enough personal training data, but other approaches in pedestrian pose recognition [1] could be employed for immediate deployments.

The privacy aspects of video footage are not as good as for other polysomnographic data sources. The pictures, respectively videos, of the patient's sleep are highly personal and tend to be a major concern of patients, even more so than the other data recorded by polysomnography since that requires additional analysis and interpretation. In the sleeping laboratory, the privacy of the video data together with the recorded sensor data is determined by local policies and regulations. For a deployment at home such regulations in terms of a privacy policy are mandatory.

Videometry is in general easy to deploy and maintain, since video cameras can be wall-powered in the sleeping environment. Another fact that makes videometry relatively easy to deploy is that the video camera is independent from the patient, unlike other sensors that are being worn by the patient, and that due to the patient's motion during sleep tend to be affected. In our research a mounted camera and a recording medium are needed, and a wide selection of both is available as standard components.

3.1 Implementation

Physical setup. Our prototype for recording videographic data consists of a night vision camera and a stand. The camera is a commercial product with built in IR-LEDs and a pan-tilt camera head. The lightweight stand that supports the camera is capable of positioning the camera above the subject, so that at least the subject's head and chest area (see figure 2) can be recorded in detail. This particular camera is capable of different communication channels. Its interface can be accessed either by LAN or WLAN with WEP, WPA, or WPA2 encryption. The video data can also be stored via USB mass media, or via video streaming over LAN, WLAN, or BNC. We will discuss some possible communication strategies in section 3.2.

Posture detection. To detect the patients' postures, we are using a semi-automatic procedure. As we are operating on images recorded by the night vision

(a) A stand allowing full body capture (b) Example of a commercial IR camera (c) Camera's good resolution and lighting

Fig. 2. Components for the video modality

camera, the challenge of the procedure is to extract those images, which mark a new sleeping posture by pruning images automatically in three steps followed by a manual analysis of the user. Thus the procedure is subdivided into four steps:

1. Automatically reducing the number of images by limiting the camera's frame rate to 1 fps. This can be done without loosing important information, because we want only to detect sleep phases, that are lasting longer than a few seconds.
2. Automatically reducing the number of images by using the camera's built-in motion detection algorithm. To ensure 100% recall the threshold of the motion detection algorithm is lowered as much as possible. This causes a comparatively high false positive rate, but reduces the false negative rate.
3. Automatically reducing the number of images by applying an averaged image differences algorithm (ADIA) on the remaining images. The algorithm is described below.
4. Manually pruning false positives from the remaining images by the user on the automatically reduced image set.

The averaged image differences algorithm (ADIA) works on the differences between successive images. It can be subdivided into three steps:

1. Calculate the differences of the successive images. We only store the average number of differing pixels, which is calculated as follows:

$$0 \leq ad = \frac{sum}{width * height * depth * num\_channels} \leq 1 \tag{1}$$

   where *sum* is the sum of all pixels for every channel. The averaged difference for every pair of successive images is stored in a vector *ads*.
2. The vector of averaged differences is then mean filtered with a dynamic kernel size. A size of one percent of the total vector size has shown to be working well. While changing a posture we typically detect heavy variation of

the averaged differences. By mean filtering *ads* we can reduce this unwanted behavior.

3. To detect posture transitions, we threshold the filtered vector *ads* and extract the falling edges to ensure that we select images that differ enough from each other. The threshold is calculated as the arithmetic mean of the vector of averaged differences.

   Next, the image is determined were the motion has settled down and the postures are "stable". It turned out that the images, which correspond to the next local minima after falling edges in *ads* are appropriate for this procedure. The new posture starts in the vicinity of the last local maxima before falling edges in *ads*.

The images which correspond to the local minima are shown to the user who then annotates these images. If the posture has changed compared to the last identified posture, we store the last local maximum as a starting point.

3.2 Analysis/Experiments

Privacy and deployment. Security is a fundamental concern, especially due to the personal nature of the recorded video data. Therefore, we introduce five possible scenarios for collecting the data and discuss their advantages and disadvantages. The analysis is subdivided into three parts: configuration, recording and transfer. The configuration comprises the user's setup of the camera which consists of the configuration of the recording scope by panning and tilting the camera's head. This can be done only by accessing the camera via the provided web interface and therefore requires a connection via LAN or WLAN. The recording part consists of the storage for the pictures or video material on an external device (via the built-in USB port or by accessing directly the video stream of the camera via LAN or WLAN). The transfer part comprises the transmission of the recorded visual data from the storage medium to a computer, where it can be post-processed. The different scenarios are depicted in figure 3.

The data has to be securely transferred due to the personal nature of the recorded data. In scenario 4 data is recorded via an ad-hoc connection. The camera's software is only capable of transferring data with WEP encryption in ad-hoc mode. As WEP encryption is highly insecure [12], scenario 4 is dismissed. Another important factor is that a person's sleep should not be influenced by the used equipment to ensure an usual sleeping environment. Scenario 2 and 4 contradict this assumption since a wireless connection would disturb a person's sleep due to radiation from a wireless connection [6]. Besides this fact the reliability of the overall system is increased by using a wired connection in contrast to the more failure-prone wireless connections. Scenarios 1, 3 and 5 differ only in the configuration part. In scenario 1 a secure WPA2 encrypted connection is used. However this depends on an additional WLAN router. This router is not needed in scenario 3, where the camera is configured via an insecure WEP encryption. In scenario 5 no WLAN router is needed at all and the security factor can be dismissed here. However, this scenario depends on a cabled connection, which requires a nearby PC or laptop.

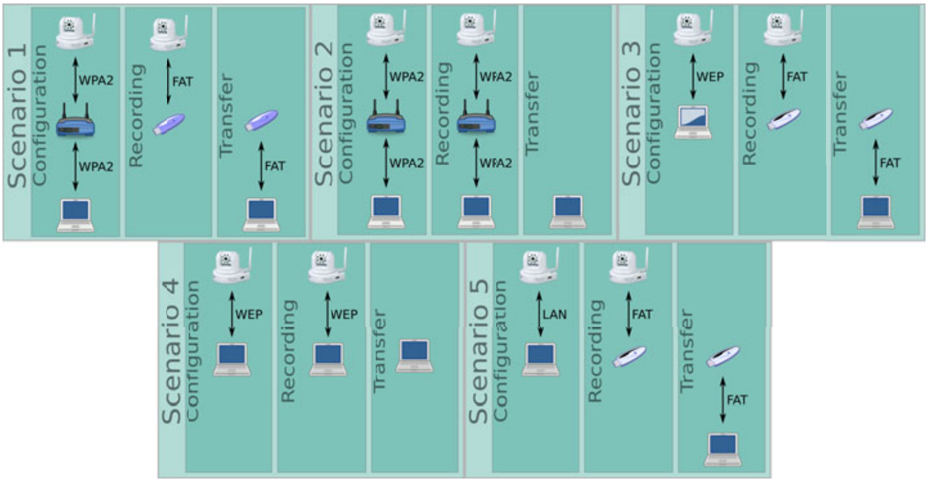


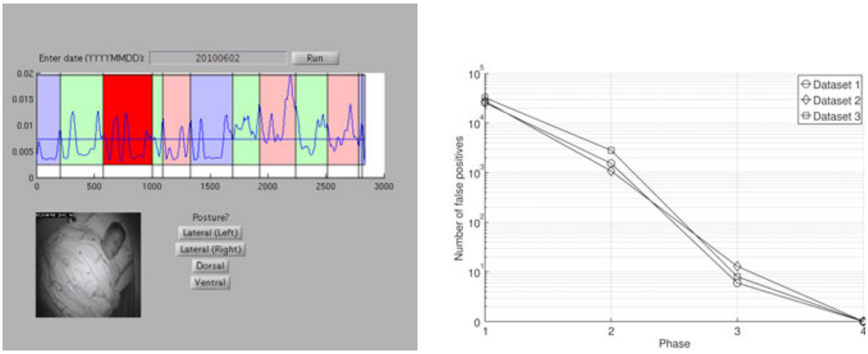
Fig. 3. Video capture scenarios

As already mentioned, security is crucial and an additional WLAN router would be far more complex than using a simple cabled connection, reducing the required hardware to a minimum. With scenario 5 the subjects will consequently be confronted with the fewest inconveniences. Trust in the setup is increased by the fact that the camera is physically disconnected from any network connection during recording.

Prototypical implementation. To test the described algorithms we have implemented a prototype. Figure 4a shows a screenshot of the prototype with an already tagged dataset. The postures are described by the colored areas in the plot. We have tested the prototype on three datasets which where recorded on different nights. The number of false positives were determined immediately after recording (phase 1), the built-in motion detection algorithm (phase 2), ADIA (phase 3) and the manual pruning of the user (phase 4). The reduction of the false positives to zero is depicted in figure 4b.

Many of the false positives after running ADIA on the images result from heavy motion without changing posture. To automatically eliminate those false positives a highly sophisticated algorithm would be needed. Due to the low number of false positives, it is not necessary to implement another algorithm which can detect the postures automatically. Because the recorded scenes vary heavily for different users, it is doubtful that an algorithm can automatically detect the postures reliably for every user. There are some promising research results in sleep posture detection [14].

The implemented algorithm ADIA deteriorates on changing light conditions. The number of false positives raises when the illumination of the recorded scene is changing fast. This behavior can be observed e.g. on cloudy days after dawn. Another problem arises from sleep postures which cannot be visually detected because the blanket is covering the complete body.



(a) GUI of the prototypical implementation

(b) Reduction of false positives

Fig. 4. GUI and reduction of false positives

4 Low-Fidelity Data: Wrist-Worn Accelerometer

Actigraphy devices are traditionally used for monitoring sleep disorders outside sleeping labs, as they detect movements which are used to infer sleep/wake patterns and quality of sleep. Here, we use an actigraph-like device with a 3D accelerometer instead which can also record postures.

Unlike the video camera of the previous section, this sensor modality is a lot weaker and therefore the detection reliability can be expected to be lower. The 3D accelerometer allows to detect various fine-grained orientations and can be read a relatively high frequencies (typically 100Hz or more). However, as this device is worn at the wrist, a correlation between wrist- and body posture is assumed. An indication for this can be found in [13], where a wrist-worn tilt sensor was successfully used to extract sleeping data and sleeping postures.

The data is stored directly on the sensor unit, and therefore this approach is not as sensitive as wireless monitoring. Also, the data is not immediately human-readable. However, privacy issues still remain when the device is handed over or the data is uploaded into a central system.

The sensor unit is designed to run over long periods of time and continuously without user interaction and therefore is relatively easy to deploy. Although current implementation of the prototype requires that users take off the sensor before showering or swimming, the device needs minimum maintenance compared to the video modality.

4.1 Implementation

The sensor unit is worn by the user at the dominant wrist. Before the sensor unit is handed to the user it is fully charged and switched on to record accelerometer data. The user is not concerned about recharging it or restarting the sensing mode. Should nevertheless the sensing unit not perform appropriate would this be observable after the sensing unit was given back by the user. As already mentioned, the current design of the sensor does not allow to be used with water.

The sensor unit is equipped with a 3D accelerometer and captures data at 100Hz, storing it on a 2 GB SD card. The sensor board is powered by a lithium battery and lasts for almost a month when used only in accelerometer mode. The board can be equipped with even more and especially different sensors (e.g. temperature and light), but currently only the accelerometer is used. Such a set-up is very cost-effective and can be deployed easily. The program running on the sensor board is very power-efficient for the hardware since data is only stored when necessary and equal sensor readings are enumerated and stored with the counter to the memory.

After the sensor board was given back by the user the data is analyzed. Night segments are being extracted by calculating the variance in the accelerometer data over a window of one minute. For that a threshold t is set that detects movement against non-movement by marking a window sleeping whenever the variance is below t.

The residual data consists of values that describe a persons postures. These values remain constant over a longer time span and characterize a certain posture. We want to detect these postures and describe our approach in the following section.

4.2 Analysis/Experiments

Experiments were conducted with one test subject who wore the sensor at the dominant wrist for five nights. For initial examination the recording started one hour prior and stopped one hour after sleep. To display how well postures can be detected, three different cases are described.

Case 1: artificial data set. The first case uses an artificial dataset to detect different postures. The artificial dataset is obtained by lying for ten seconds in the postures left-, right lateral, supine and prone each time before sleep.

The posture estimation is shown in figure 5. The ground truth with the estimation of the postures is shown in the top plot. For case 1 we will focus on the artificial set, the other two sets will be explained in the following parts. By visual inspection it is evident that posture changes are detected accurately, whereas the posture itself has been detected poorly. A few overlaps in ground truth and estimation of the posture are visible.

Precision and recall for the postures are in the range of 62% and 35% respectively (see figure 6, left plot). Remarkably, the detection of the right lateral posture exhibits a recall of 95% and a precision of 53%. The false positive rate for the other postures is very high and can be explained by the fact that each posture is not represented by the value obtained from the artificial dataset. Due to the spontaneity of a person's positioning during sleep it is almost impossible to simulate an accurate artificial dataset. The position of the wrist during a posture while sleeping varies frequently, resulting in different accelerometer values.

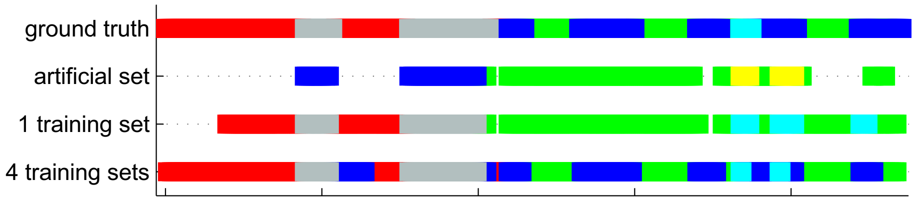


Fig. 5. Detected postures with increasing amount of training sets

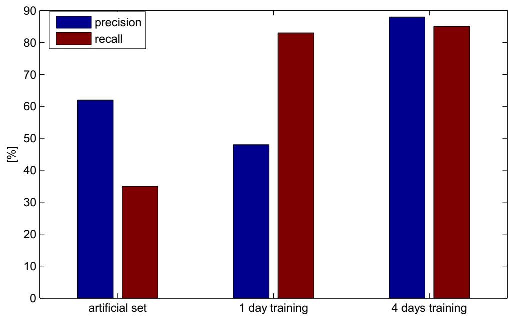


Fig. 6. Precision and recall for all three cases

Case 2: 1 training set. To compare the first case to a different approach, in the second case a k-nearest-neighbor (KNN) classifier is trained on one night only. The classifier is then tested on the four other nights and compared to the ground truth. This procedure is repeated for each of the 5 nights.

The visual results are shown in figure 5, described by the one day training set. In contrast to case 1 the results of detecting posture changes and the posture itself improved. Overall a precision of 48% and recall of 83% are obtained. Precision dropped in contrast to case 1 but note that recall increased (see figure 6, middle plot). A classifier was obtained by training on one night only. In theory a person takes in only characteristic postures, limiting the amount of postures of a person [4]. Therefore the number of training datasets (or nights) has to be sufficient to gain a classifier that covers all possible postures.

Case 3: 4 training sets. Finally in the last case a KNN classifier is obtained by training with posture labels and accelerometer data using 5-fold cross-validation, where one fold corresponds to one night. The difference to case 2 is that the classifier is trained on four datasets.

The results are summarized in figure 5, showing again an improvement of detection of posture changes and the estimated postures, almost overlapping with the ground truth. Notice that in contrast to the previous plots of case 1 and 2 no unclassified posture is detected, leaving no whitespaces in the plot. By training a classifier on more than one night the precision is increased to 88%, whereas recall stays steady in the range of 80% (see figure 6, right plot).

Using only an artificial dataset is not feasible due to user-specific postures and is therefore not further followed.

4.3 Combining Both Modalities

We have shown that both systems can be operated individually to detect sleeping postures. We have found that, as expected, the video footage is superior to a wrist-worn sensor in terms of detection reliability, since its data contains more details on the sleeping posture, assuming that the user prunes the system by sorting out false positives manually after each night. This is first and foremost feasible since the camera's output can be directly interpreted by the user. The wrist-worn sensor on the other hand operates on its own without the need of installing environmental devices and without capturing data as sensitive as video. Although a full study still remains to be performed on the privacy issues of this paper's particular wrist-worn sensor, we can deduce from similar devices, actigraphs, that they bring along far less privacy concerns compared to video footage.

By combining both modalities into one, we can overcome the limitations of the individual parts: In the combined system we propose to use the visual data to obtain ground truth, which is then used to train the wrist-worn sensor unit for multiple nights. By following this approach, we improved for the earlier mentioned dataset the overall detection rate to over 80% in both precision and recall by only training the wrist-worn sensor units on the data from four nights (see figures 5, 6). This approach maintains the privacy of the wrist-worn sensor unit by using the inertial posture data for detecting the subjects sleeping postures, and storing this information alone. The data of the camera can be immediately discarded after usage in the training phase, which can be done at the user's home without involvement of third parties. The video data is thus accessible only by the user, thereby reducing the privacy concerns of the overall system to the wrist-worn sensor unit's privacy issues. The price to pay for these two advantages are (1) a slightly degraded applicability, due to the deployment of the video system and the fact that the inertial data has to be downloaded after each night *for the duration of the training phase*, and (2) the video-based privacy issues remaining during the training phase as well.

As the main weakness of the combined system is the training phase, it is important that this lasts as short as possible. To investigate how many days are needed to obtain an acceptable precision and recall for each posture, we increased the number of training nights in the cross-validation.

Figure 7 shows the given training sets' results for the right lateral posture: recall and precision are increased from 77% to 84% and 72% to 82% respectively by using two training sets instead of one only. Both are improved by four training sets, leading to a recall of 97% and a precision of 84%. We conclude that one training set is not sufficient to detect the postures appropriately, but a training set of four is leading to an overall precision and recall of over 85% respectively.

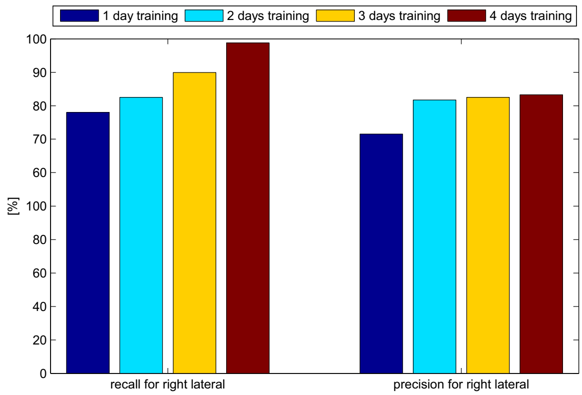


Fig. 7. Precision and recall for right lateral: more training nights improve the recognition

5 Conclusions

This paper reported on our efforts to design a domestic sleep posture monitoring system for long-term deployment, which allows sleep researchers to track their patients outside the laboratory. We argued that the long runtime and the home environment make for a difficult design: We identified three challenges that such a system must meet, and used these as a framework to describe and analyze two sensor modalities: video monitoring of the sleeping user, and a wrist-worn accelerometer logger. Through several studies, we have explored how well these fare on their own, and proposed finally a multi-modal solution that uses both in a system that reduces all challenges.

By combining video information and sensor data we obtained precision and recall of over 80% respectively which are promising results for long-term studies. The setup is easy to deploy and requires only little user interference, which increases the acceptance on the persons' side. Such a system provides a potentiality for the use in sleep studies where patients suffer from for example sleep apnea and have to be monitored in their usual environment.

Further experiments over more nights have to be conducted to confirm our findings. The sensor modality is continuously improved to make it more energy efficient and thereby enhancing recording time for long-term experiments. We are also researching on a full automatic posture detection by using only the video modality. In our approach we follow the procedure, which is used to detect people in [3].

References

1. Andriluka, M., Roth, S., Schiele, B.: Pictorial structures revisited: People detection and articulated pose estimation. In: Proc. CVPR, vol. 1, p. 4 (2009)
2. BBCNews: Sleep position gives personality clue (2003), http://news.bbc.co.uk/2/hi/health/3112170.stm

3. Dalal, N., Triggs, B., Rhone-Alps, I., Montbonnot, F.: Histograms of oriented gradients for human detection. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, pp. 886–893 (2005)
4. Gordon, S., Grimmer, K., Trott, P.: Self reported versus recorded sleep position: an observational study. Internet Journal of Allied Health Sciences and Practice 2(1) (2004)
5. Hong, J., Ng, J., Lederer, S., Landay, J.: Privacy risk models for designing privacy-sensitive ubiquitous computing systems. In: Proceedings of the 5th Conference on Designing Interactive Systems: Processes, Practices, Methods, and Techniques, pp. 91–100. ACM, New York (2004)
6. Hyland, G.: Physics and biology of mobile telephony. The Lancet 356(9244), 1833–1836 (2000)
7. Idzikowski, C.J.: Learn to Sleep Well. Duncan Baird (2000)
8. Ojala, T., Kukka, H., Linden, T., Heikkinen, T., Jurmu, M., Hosio, S., Kruger, F.: Ubi-hotspot 1.0: Large-scale long-term deployment of interactive public displays in a city center. In: International Conference on Internet and Web Applications and Services, vol. 0, pp. 285–294 (2010)
9. Oksenberg, A., Arons, E., Greenberg-Dotan, S., Nasser, K., Radwan, H.: The significance of body posture on breathing abnormalities during sleep: data analysis of 2077 obstructive sleep apnea patients. Harefuah 148(5), 304 (2009)
10. Peng, J., He, H., Zhu, P., Liu, Y., Zhang, X., Jin, Y.: Zigbee-based new approach to smart home. Journal of Shanghai University (English Edition) 14(1), 12–16 (2010)
11. Peter, H., Penzel, T., Peter, J.: Enzyklopädie der Schlafmedizin, 1st edn. Springer, Berlin (2007)
12. Tews, E., Weinmann, R., Pyshkin, A.: Breaking 104 bit WEP in less than 60 seconds. Cryptology ePrint Archive 2007/120, Cryptography and Computeralgebra group, Darmstadt University of Technology (2007), http://eprint.iacr.org/2007/120
13. Van Laerhoven, K., Borazio, M., Kilian, D., Schiele, B.: Sustained logging and discrimination of sleep postures with low-level, wrist-worn sensors. In: ISWC (2008)
14. Wang, C., Hunter, A.: Robust Pose Recognition of the Obscured Human Body. International Journal of Computer Vision, 1–18 (2010)

Utilizing Social Context for Providing Personalized Services to Mobile Users

Athanasios Karapantelakis[1,2] and Gerald Q. Maguire Jr.[1]

[1] Royal Institute of Technology (KTH), Kista, Sweden
[2] Conformiq AB, Stockholm, Sweden
{athkar,maguire}@kth.se

Abstract. We are currently witnessing a growing tendency for users to engage in social activities on the Internet. Consequently, social networks - tools that enable such activities - are increasingly used. This paper describes a system that uses information users share on these networks (personal context), to recommend Web feeds of related content to users. The system mines data from popular social networks and combines it with information from third party websites to create user profiles. Subsequently, these profiles are matched with appropriately tagged Web feeds and are displayed to users through a mobile device application. We evaluate the system with a 6-month test run involving real users. We measure the system's performance as well as user satisfaction and assessment of the accuracy of recommendations by including a feedback mechanism on the mobile application.

Keywords: social context, recommender systems, user profile, web feeds, personalized services, active streams.

1 Introduction

The use of social networks has increased in recent times: statistical evidence indicate that not only are more people joining these communities, but there is an increase in the average amount of time spent [1]. This increase in the number of users has helped aggregate value to social networks; mainly due to the network effects, i.e. value increases, as the numbers of users increase.

Realizing the value of the ever-increasing user-base, companies behind social networks in collaboration with *content providers*[1] are trying to leverage these strong demographics by providing personalized services to their subscribers, which not only render social networks more attractive to other users, but they can potentially be a source of revenue [2].

[1] In this study, we define content providers as third party entities (companies, teams of people, or individuals) willing to provide services to interested users. Note that we do not make a distinction as to the form of services provided - thus grouping all such service providers under the same term. The reader should also note that the terms service and application are used interchangeably, to indicate social services for users, built on top of social networks.

P. Lukowitz, G. Kortuem, and K. Kunze (Eds.): EuroSSC 2010, LNCS 6446, pp. 28–41, 2010.

Currently, personalized content services from social networks use personal context superficially. One category of services requires user intervention in order to function; for example, profile page customization and installation of third-party applications. Some networks also provide content based on proactive recommendations. The criteria for content selection are typically direct matches of personal context to the content target requirements. Examples include location and age aware advertising services, new contact suggestion based on the current network of contacts and a birthday reminder service.

In order to survive the competition and due to privacy concerns, popular social networks do not support sharing of personal context between them, and therefore prohibit the use of aggregated information which could be used to provide better personalized services. The usefulness of aggregated information is accentuated by the results of statistical studies showing that typically users join multiple networks and submit different information to each [3].

In this study, we describe a system that aggregates information from multiple social networks (Facebook, MySpace and LinkedIn) to provide personalized content to end users through Web feeds. We implemented and exposed the system for public use for a 6-month period and measured the performance and perceived quality. The rest of this paper is structured as follows: Section 2 provides an overview of prior work upon which this system is based. Section 3 describes the architecture and implementation details of the system. Section 4 presents our measurements and evaluation results. In section 5, we outline our contributions and suggest future directions for this system.

2 Related Work

In this study, we use ontologies to model personal context. Advances in semantic web representation technology have facilitated the process of modelling personal data efficiently. A combination of schemas based on the Resource Description Framework (RDF) can be used to accurately represent users and their relationships in social networks [4] [5]. In order to accommodate information from different sources, we extend these RDF schemas with custom namespaces (see section 3.1).

Contemporary Web feed providers such as Google News [6], Yahoo [7] and BBC [8] provide the current state-of-the-art in personalised content by aggregating feeds from various sources (i.e. Web sites) and letting the users choose the content they are interested in. However, this choice is limited to generic categories (e.g. News, Sports, Entertainment, etc) which describe thematic areas of Web feed sources and not the content of individual feeds. The result is that Web feed content may not be sufficiently described and matched against user preferences.

One solution is to use Natural Language Processing (NLP) algorithms for automated text classification. However, due to inherent ambiguity, classifying documents in natural language is difficult without any prior domain knowledge [9]. Therefore, only systems offering personalized content for specific domains such as email [10] and multimedia [11] are sufficiently accurate in their predictions. In these cases, both the vocabulary and the syntax of the content follow formalized rules, making them easier to classify.

In this system, we are not limiting the provided content to a specific domain, therefore we delegate the task of tagging the content properly to the content providers. Our solution follows the findings of Kurki et al. who have identified the importance of the role of professional editors, who are responsible for creating and properly tagging their content, as well as the role of automated software agents, which perform user profiling, document matching and delivery [12].

3 System Architecture

In [13] we described our implementation of a system for maintaining user subscriptions to a printing service. This system retrieves context and allocates available printing resources to incoming user requests: In this study, we have adapted the original design to add new functionality: instead of printing resources, we now have internet website content (see figure 1). SIP/SIMPLE [14] is used as the protocol for subscription and notification. The rest of this section further describes the technical details of the four system components.

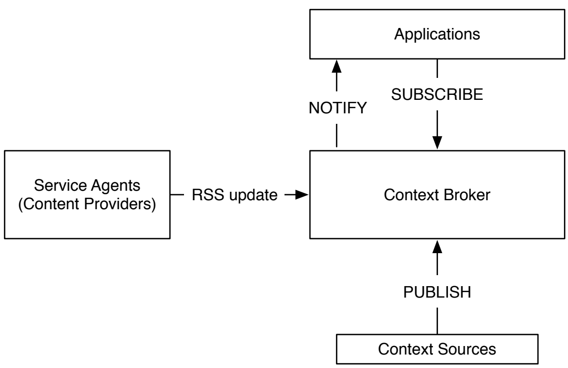


Fig. 1. The four components of the system. Context sources publish user profiles to the context broker and content providers provide updated content using RSS feeds (see section 3.2). Users subscribe to the context broker (see sections 3.3 and 3.4) . This broker runs a matching algorithm (see section 3.3.2) in order to establish whether a users's profile information matches any of the offered content. In the event of a match, the user's application gets a notification with the content data that matches their interest.

3.1 Context Sources

In this paper we introduce as a *Context Source* software that is responsible for mining personal context from social networks and sending this information to the context broker. For our study, we chose to retrieve information from three popular social networks (see table 1). Our measurements have verified

Table 1. Retrieved user context from social networks

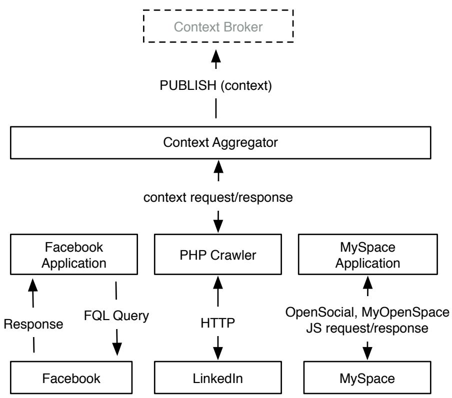
Social Network	Personal Information	User relationships
Facebook	Given name, birthday, sex, current location, status message, religion, political views, activities, hobbies (interests, tv shows, movies, books).	Groups and networks membership, event attendance, list of friends.
MySpace	Given name, age, location, mood, users's music and video	List of friends
LinkedIN	Given name, work experience, education	Groups and associations, recommendations, connections

that the user information contained in these networks is complementary and not redundant [15]: as each network specializes in distinct social activities, users tend to store different information in each one. In cases where we found similarities, we have chosen to use only the information included in the Facebook profile since our measurements have indicated that profiles on this network were in general more complete.

A software entity we call a *Context Aggregator* (CA) coordinates requests for user context using a timeout counter (see figure 2a). Whenever the counter expires, the CA issues a new request towards the three networks in order to retrieve any available information for this specific user. Once all three responses are received, the CA compares the user information contained in these responses, with the information retrieved from previous requests (note that the CA has sufficient volatile storage to store user context from several previous requests). If the CA determines the existence of new information about the current user, it integrates all the received information into a single data stream, which is sent as a payload to a PUBLISH message to the Context Broker (see section 3.3). The reader should note that the CA does not do any transformations on the retrieved data which is transmitted as received to the Context Broker (see figure 2b).

3.2 Service Agents

The Service Agent concept, first presented in [13], relates to entities that provide services to users. In the scope of this study, a service agent can be likened to a *Content Provider* (CP). A CP is an on-line service providing public content by syndicating one or more web feeds: this way, users can subscribe to the feeds they want, and receive new information as it becomes available. For this study we will be using Really Simple Syndication (RSS), one of the most popular formats for feed distribution [16]. RSS allows for configuration of each item in the feed, by including one or more category tags therefore the CP can describe the feed with high precision. For our system we have manually chosen a number of different websites already providing RSS feed coverages. The choice criteria for these sites were thematic diversity and daily activity. Another prerequisite was that the website should tag their RSS content with the thematic category/categories



(a)

[fb] given_name = 'Athanasios Karapantelakis', birthday = '1982', sex = 'male', current_location = 'Stockholm', profile_picture = 'http://www.it.kth.se/~athkar/me.jpeg', activities = 'red cross volunteer', tv = 'bbc news', movies = 'comedy', books = 'the long tail, the shelfish gene', groups = 'tripadvisor, conformiq', networks = 'Sweden', event_attendance = 'muse live at hovet, greenday live at globen', friends = 'John Smith, Mike Brown, Richard White' [/fb]

[ms] music1 = 'Frank Sinatra, New York New York, 1977', music2 = 'Nice Work If You Can Get It, Ella Fitzgerald, 1983' friends = 'Markus Blue, John Smith' [/ms]

[li] work1 = 'solutions engineer, Ericsson AB, Stockholm, Telecommunications' work2 = 'solutions architect, Conformiq Inc., Satratoga CA, Computer Software' connections = 'Mike Brown, Markus Blue' [/li]

(b)

Fig. 2. Block diagram of context sources (a), and an example of user information transmitted as payload to PUBLISH messages (b). The only information added by the CA are the beginning and ending tags indicating the the source of the data.

that the content refers to (i.e. location, news, work, entertainment - and more specific description such as media type, industry, social activiies etc). Web feeds are sent to the Context Broker, which also subscribes to news feeds from content providers.

3.3 Context Broker

The Context Broker (CB) is responsible (a) for receiving raw user context from context aggregators and transforming this context to machine-readable user profiles using ontologies, (b) retrieving and storing web feeds from content providers, (c) matching user profiles against web feeds, and (d) notifying users of feeds that may be of interest to them . It is a software solution based on the OpenSIPS SIP Server [17] with custom software extensions (modules) that provide the desired functionality (see figure 3). The rest of this section describes each module separately.

3.3.1 Receiving, Processing and Storing User Context and Web Feed Content. Personal context is received from the CB as the payload of PUBLISH requests. We implemented a context transformation engine to transcode user context from a raw format (see section 3.2) into machine readable ontologies. The context transformation cycle follows the process described in section 2.1.2: Initially, raw context information is segmented and stored into three arrays, one per social network. Subsequently, these arrays are used as input to information extractors. These extractors create XML documents of personal context based on formalized descriptions, or vocabularies. For the purposes of this study we are using the Friend-of-a-Friend vocabulary (FOAF) [18].

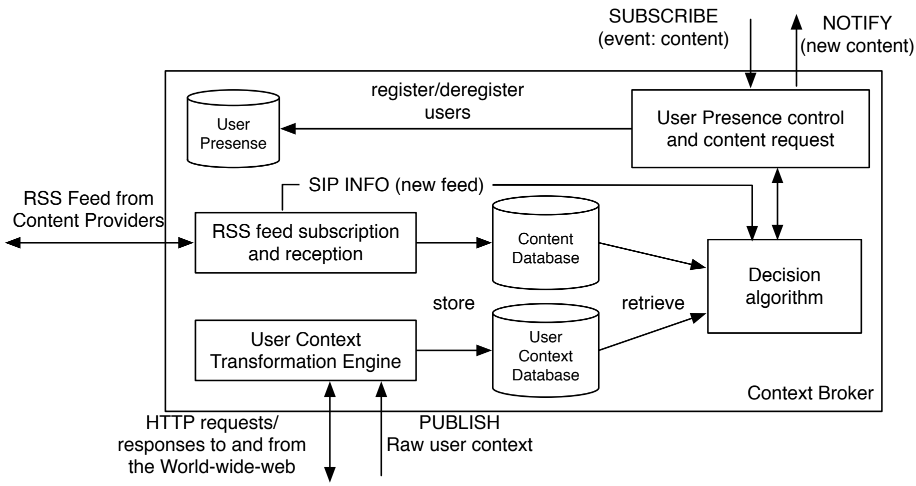


Fig. 3. Modules of the context broker

A FOAF document describes a user's personal context to a network and is identified by a unique Universal Resource Identifier (URI). It is possible to link all multiple FOAF documents together in case a user maintains accounts in multiple networks [22].

Although FOAF provides a generalized description of a user's personal context, we have found cases where the vocabulary is insufficient to describe this context. For example, although the original FOAF specification has defined properties describing a person's place of work (*foaf:workInfoHomepage* and *foaf:workplaceHomepage*), it cannot match the depth of information received from LinkedIN (see also figure 2). For this reason, we extended the existing FOAF vocabulary to include more detailed descriptions of personal context. We defined two new schemas based on RDF: namely *businessBackground* and *mediaFiles*. The combination of these schemas together with FOAF allow for a complete description of personal context. The resulting document is a *user profile* [15].

After the user profile is compiled, the transformation engine attempts to deduce general user interests by combining context from the document with Web directories. The rationale behind this processing is to attempt to match the user's interests to the general content categories defined by content providers. This will later simplify the decision making process (see section 3.3.2):

– Extracting location-based information using Google Maps API: Personal context typically includes information about the user's current location. However, we have found out that different users tend to specify their location differently. For example, the city name Athens, Aten, Athinai and $A\theta\eta\nu\alpha$ all refer to the same capital city of Greece. Other users may be more specific and specify a specific neighbourhood of the city, street name, etc. Instead of providing a fuzzy view of the user's location, we use the provided information as argument to queries to the Google Maps database using the Google Maps API [19] which returns geographical coordinates (latitute and longtitude). This information is used to compute relative distances (see section 3.3.2). Measuring the efficiency of this process, we have found out that Google Maps is reliable, in spite of the variety of location entries that actually appeared [15].

– Extracting genres from movies using information stored in the Internet Movie Database (IMDB): We have used the IMDB PHP library of functions to write software which queries IMDB about movie genre. As per previous, this software supports partial matching of movie titles to genres, which helped maintain true positive matchings at a relatively high percentage [15].

Data is retrieved from the web via HTTP requests. In the event of a positive match, the new data is written to the document replacing the original resource. The final document is stored inside a local user context database. If a prior document for a user already exists, it is overwritten by the new one. Interested readers can access examples of user profiles before and after post-processing at [15].

Similar to the context transformation engine, we implemented software for Web feed subscription and reception (i.e. RSS module), which subscribes to, and receives Web feeds from content providers using the RSS protocol (see also section 3.2). These feeds are temporarily saved into a local database, to be retrieved later, by the Decision Algorithm (see section 3.1.2). The RSS module also augments the received information using HTTP requests so that location-based entries are converted to geographical coordinates and movie reviews are tagged with genre information if they are not already tagged.

3.3.2 Matching User Context with Web Feeds. The decision algorithm module is responsible for matching any web feeds stored in the content database, with user profiles stored in the user context database. An incoming SIP INFO message informs the module that a new feed is stored in the content database (see figure 3). The module retrieves the feed and all the user profiles stored in the context database. The algorithm calculates possible alignments for every <*user profile, new Web feed*> ontology pair. For every true positive alignment, the Web feed is forwarded to the user presence control module. Our formalization for the matching algorithm is based on the work of Shvaiko et al. [20]. We adopt the following symbolism:

– Let w symbolize the ontology used to describe an incoming web feed. Also, let $U = \{u_1, ..., u_n\}$: n $\in \mathbb{N}$ the set of ontologies stored in the user context database (here n is the total number of ontologies). The input of the algorithm in this case is all the pairs of combinations $< u_x, w > \forall x \in \{1,...,n\}$
– A_i is a degree of likelihood two ontologies match. It is expressed in a normalized scale [0, 1] where 1 means absolute similarity. Positive results are those where $A_i \geq C_m$, where C_m is a confidence measure for true positive alignments indicating an acceptable match. In this case, the Web feed can be forwarded to the user application (see section 3.4). From empirical measurements we have concluded to use a C_m value of 0.6.

The comparison is based on a predefined table of taxonomized categories based on the nature of the content. Matching is conducted on different levels, from general to more specific content. Each layer aggregates larger values to the final alignment. An exact matching, means that the item being offered is the same as the one the user is interested in (for example, a Web feed about a movie review of one of the user's favourite movies), yielding an alignment value of 1. We obtained a list of content categories from the Open Directory Project [21]. Table 2 illustrates the 4-layer categorization approach for this project. Each keyword is assigned an alignment weight. General category keywords have the lowest alignment weight. The alignment weight is higher when a keyword matches an entry in the user context more closely the highest alignment level is achieved when an item in the content description syntactically matches a user entry in the context document (direct match). Note that keywords shown here do not represent the full set of keywords used for this project and that the full set can be found in [15].

Table 2. Categorization of keywords for matching web feeds against user context

General Category Description ($A_1 = 0.2$)	Sub-Category Description ($A_2 = 0.4$)	Specialization Description ($A_3 = 0.1$)	Direct match ($A_4 = 0.3$)
News	Politics, Technology, Social and Lifestyle, Travel, Financial, Business	Location, Author/Editor, Industry (i.e. Technology, Lifestyle etc).	Syntax matching
Entertainment	Movies, Songs, TV Series	Genre, Artist or Director	
Work	General company news, Meeting announcements, CEO letter	Department or workgroup, Author	
Hobbies	Skydiving, surfing	Location proximity	

As an example, consider a situation where two user profiles are stored in the context database. Additionally, consider an incoming web feed from the RSS feed subscription and reception module. All of this content is shown in table 3. Based on this information, new incoming content from the web feed triggers execution of the decision algorithm. The algorithm calculates alignments for user A and user B. For user A, the closest match is the category of the web feed with the interests of the user. The user's location is too far from the event described on the web feed (the relative distance being around 2394 km). For user B however, the locations match since the address of the event is only 714 m from the user B's given location. Additionally since the web feed is also tagged as a social event, it matches user B's socializing interests, leading altogether to a better match than user A: According to table 3, alignment for user A is calculated to be 0.6 and for user B 0.7, meaning that both feeds will be relayed to the users.

Table 3. User profiles for users A and B and content of an RSS feed, including geographical coordinates and movie genre information retrieved from Google Maps and IMDB respectively

User A	User B	Incoming Web Feed
Interests: entertainment, movies Gone With the Wind Genre: drama, romance, war Location: 37.97918, 23.716647	Interests: news, lifestyle Location: 51.500152,-0.126236	Title: London Film Festival: The awards Category: News, Social-lifestyle, Entertainment, Movies Location: 51.503441,-0.117394

Greater alignment values play a role in the user application space, where we have implemented web feed prioritization (see section 3.4).

3.3.3 Processing User Subscriptions and Notifying the User for New Feeds. The user presence control module uses the *SIP/SIMPLE* protocol to accept user subscriptions and notify the users of new web feeds relayed by the decision algorithm (see section 3.3.2). A user application (see section 3.4) initially sends a *SUBSCRIBE* request with event header *feedinfo* and receives *NOTIFY* requests whenever an incoming relevant news item is available. Applications have to send *SUBSCRIBE* requests periodically in order to keep the session alive. The NOTIFY message contains an RSS document with a description of the file and a link to the origin (i.e. the content provider's website). It also contains an alignment value which is useful to the user application for prioritizing feeds (see also section 3.4).

3.4 User Application

We have implemented a user application for the iOS platform. The application allows a user to subscribe to the *feedinfo* service (see section 3.3.3), and to receive incoming notifications (see figure 4a). The items can be displayed on order of arrival (the newest one listed on top), or by relevance (see figure 4b). When evaluating the relevance of a notification, the user application compares the alignment values of each item, so that items with larger values are listed higher than those with lower alignment. When selecting an item on the list, the application automatically opens a new web browser window containing the webpage of the content provider who publishes the feed (see figure 4c).

The application also includes a user feedback form, where the user is able to submit their empirical evaluation of the system's performance (see figure 4d). The results of each evaluation are sent through the application to a separate server which stores the data for later processing. The form asks a series of questions with the answers based on a semantic scale from 1 to 5, where larger values signify a more positive experience.

(a) (b) (c) (d)

Fig. 4. Screenshots of the user application

4 Measurements

4.1 Introduction

In order to evaluate our approach, we operated the system during a 6-month period from February 2009 until July 2009. Any user with an internet access; one or more active profiles on Facebook, LinkedIN, and MySpace; and an iOS based mobile device could join. During this period, a total of 462 unique users used the system. All participants gave their permission to allow the system to access their personal information stored in social network profiles to be used for anonymous statistical analysis. We measured aspects of system performance, as well as the perceived user experience.

4.2 System Performance

The propagation delay metric measures the delay for a web feed item to propagate from the content provider, to an eligible active subscriber (including any network propagation delays, the time spent in storing the query into the content database, as well as the computation time of the decision algorithm). The decision algorithm has to match the feed item against each user's context document, so it is interesting to see the relation between the propagation delay and the number of users. Figure 5 shows the results using datasets retrieved towards the end of each month of system's utilization, starting from February 2009.

We observe that the propagation delay increases as the number of users increases. This occurs because the algorithm has to match a larger number of user context documents with a feed item. This performance could be improved by prioritizing the computation such that the content algorithm computes the feed alignment for active subscribers, as the computation of the alignment for the off-line users can take place at a later time, when the system utilization is low. When

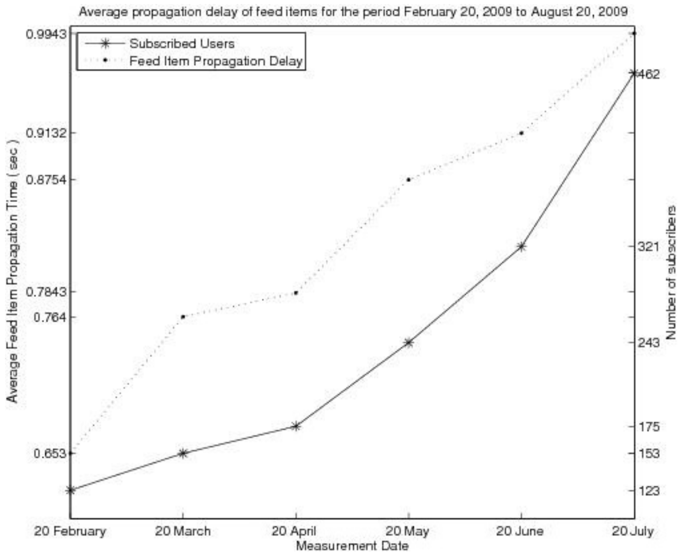


Fig. 5. System response time during a measurement period of 6 months

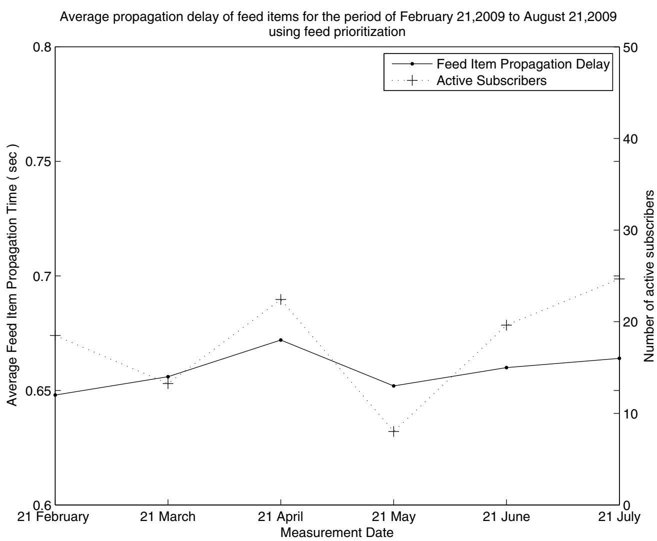


Fig. 6. This figure illustrating the average system response time after prioritization for active subscribers. Active subscribers are users who were using the system at the time of the measurement.

using this optimization, we have observed a great reduction in propagation delay, since the number of active subscribers at the time of the measurements was much less then the number of total subscribers (see figure 6).

4.3 Perceived User Experience

We measure perceived user experience through user feedback submitted from the user application (see section 3.4). Prior to using the system, users were asked to install this application on their mobile devices and input their social profile credentials in order for the system to be able to access their personal context. This information was transmitted from the device to the system, which successively provided the user with personalized Web feeds. As previously described, the system would update the list of Web feeds dynamically in case user changed information on his/her personal profile which affected the relevance of the links, or when new incoming relevant links were provided from content providers.

At the end of the test-run period, each user was able to submit an evaluation of the system via the client application (see section 3.4). In total, 460 out of 462 users participated in the survey. The users were asked to rate overall satisfaction of using the system, which includes quality of the provided feeds (i.e. if the feeds were interesting to the users), the update rate of the Web feeds, as well as the time it took for the system to adapt to changes in the personal context (for example, when a user entered information that he has relocated to a new hometown, or added a movie to his/her list of favourite films). The results illustrated in figure 7 clearly show a general satisfaction with the system's performance.

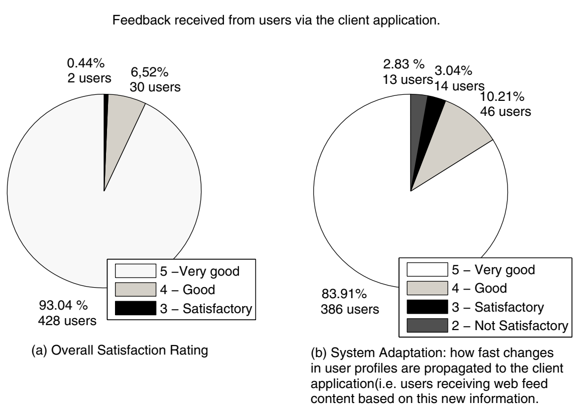


Fig. 7. These figures illustrate the overall results of the user satisfaction survey

5 Conclusion

We have presented a system for retrieving user context from social networks, then using this context together with other web resources to create user profiles. These user profiles allow the system to match incoming feeds from content providers with a users's personal interests, and suggest relevant content to users. The novelty of our approach lies in the lack of user intervention, as automated content offerings have both time and cost benefits for content providers and users alike: Content providers do not have to search for target audiences and users do not have to search for content themselves.

By subjecting the system for public use we have shown that it is possible to achieve high user satisfaction using relatively limited context information and simple matching procedures. Additionally, performance measurements helped identify weaknesses and optimize system performance. Future directions to improve the system include taking advantage of social relations (e.g. friends, business contacts, etc.) and mining other popular Web sources (e.g. twitter account and blogging sites) in order to further augment more personal context to user profiles. Finally, we intend to further study the dataset obtained during our measurement period and identify social network usage patterns.

References

1. The Nielsen Company (Whitepaper). Global faces and networked places: A nielsen report on social networking's new global footprint (March 2009)
2. Prahalad, C.K., Ramaswamy, V.: The Future of Competition: Co-creating Unique Value With Customers, Experience Personalization, ch. 5. Harvard Business Press, Boston (2004)
3. The Economist. Online social networks: Everywhere and nowhere. Economist Print edn. (March 2008)
4. Bojars, U., Passant, A., Breslin, J.G., Decker, S.: Data portability with sioc and foaf. In: The Web on the Move, XTech 2008 (May 2008), http://www.slideshare.net/CaptSolo/data-portability-with-sioc-and-foaf
5. Rowe, M., Ciravegna, F.: Getting to me: Exporting semantic social network from facebook. In: ISWC 2008 Workshop Social Data on the Web (October 2008)
6. Google news: Aggregated headlines and a search engine of many of the world's news sources (July 2010), http://news.google.com/
7. Yahoo news: Rss feed selection (July 2010), http://news.yahoo.com/page/rss
8. Bbc: The feed factory, an introduction to the rss feeds that are available from the uk website of bbc (July 2010), http://www.bbc.co.uk/feedfactory/
9. Zaihrayeu, I., Sun, L., Giunchiglia, F., Pan, W., Ju, Q., Chi, M., Huang, X.: From web directories to ontologies: Natural language processing challenges. In: Aberer, K., Choi, K.-S., Noy, N., Allemang, D., Lee, K.-I., Nixon, L.J.B., Golbeck, J., Mika, P., Maynard, D., Mizoguchi, R., Schreiber, G., Cudré-Mauroux, P. (eds.) ISWC/ASWC 2007. LNCS, vol. 4825, pp. 623–636. Springer, Heidelberg (2007)
10. Paik, W., Yilmazel, S., Brown, E., Poulin, M., Dubon, S., Amice, C.: Applying natural language processing (nlp) based metadata extraction to automatically acquire user preferences. In: Proceedings of the 1st International Conference on Knowledge Capture, K-CAP 2001, pp. 116–122. ACM, New York (2001)
11. Vallet, D., Castells, P., Fernández, M., Mylonas, P., Avrithis, Y.S.: Personalized content retrieval in context using ontological knowledge. IEEE Trans. Circuits Syst. Video Techn. 17(3), 336–346 (2007)
12. Kurki, T., Jokela, S., Sulonen, R., Turpeinen, M.: Agents in delivering personalized content based on semantic metadata. In: Proceedings of AAAI Spring Symposium Workshop on Intelligent Agents in Cyberspace, pp. 84–93 (1999)

13. Karapantelakis, A., Devlic, A., Zarify, M., Khamit, S.: Printing in ubiquitous computing environments. In: Zhang, D., Portmann, M., Tan, A.-H., Indulska, J. (eds.) UIC 2009. LNCS, vol. 5585, pp. 83–98. Springer, Heidelberg (2009)
14. Rosenberg, J.: A presence event package for the session initiation protocol (SIP). RFC 3856, Internet Engineering Task Force (August 2004)
15. Karapantelakis, A.: Additional paper resources, http://web.it.kth.se/~athkar/ontologies/ (retrieved on October 4, 2009)
16. Winer, D.: Rss 2.0 specification (July 2003)
17. The OpenSIPS community. Opensips sip server, http://www.opensips.org/ (retrieved on October 4, 2009)
18. Brickley, D., Miller, L.: The Friend Of A Friend (FOAF) vocabulary specification (November 2007)
19. Google Inc.: Google maps geocoding api overview, http://code.google.com/apis/maps/documentation/geocoding/ (retrieved on October 4, 2009)
20. Shvaiko, P., Euzenat, J.: A Survey of Schema-Based Matching Approaches. Journal on Data Semantics IV, 146–171 (2005)
21. ODP Community.: The open directory project, http://www.dmoz.org/ (retrieved on October 4, 2009)

Activity Recognition Using Biomechanical Model Based Pose Estimation

Attila Reiss, Gustaf Hendeby, Gabriele Bleser, and Didier Stricker

German Research Center for Artificial Intelligence (DFKI)
Department of Augmented Vision
Trippstader Str. 122, 67663 Kaiserslautern, Germany

Abstract. In this paper, a novel activity recognition method based on signal-oriented and model-based features is presented. The model-based features are calculated from shoulder and elbow joint angles and torso orientation, provided by upper-body pose estimation based on a biomechanical body model. The recognition performance of signal-oriented and model-based features is compared within this paper, and the potential of improving recognition accuracy by combining the two approaches is proved: the accuracy increased by 4–6% for certain activities when adding model-based features to the signal-oriented classifier. The presented activity recognition techniques are used for recognizing 9 everyday and fitness activities, and thus can be applied for e.g., fitness applications or 'in vivo' monitoring of patients.

1 Introduction

With recent progress in wearable sensing it is reasonable to detect the wearer's context or location in real life situations. Within this area of research, activity recognition has become important in recent years. The applications of activity recognition vary from industrial scenarios to fitness applications and health care.

Physical activity monitoring is not only used in therapies for various chronic diseases, but also in preventive health care. Physical inactivity is a major risk factor for diseases including diabetes, cardiovascular diseases and certain types of cancer. Wearable devices that can monitor the type and duration of activities over a patient's day can not only be used for appropriate preventive actions, but could also motivate the patient itself and so promote a more active lifestyle. The basic daily physical activity level (30 minutes) can be achieved by everyday activities like walking or ascending/descending stairs instead of using the elevator. Performing a wider spectrum of activities, e.g., endurance-enhancing activities like Nordic walking, jogging or cycling, can help to achieve even more health benefits.

Previous work has shown that the recognition of basic physical activities, such as resting, walking, running and cycling, is possible even with just one 3D-acceleration sensor [2,3,11,12]. However, when increasing the number of activities, the classification performance drops significantly. The number of sensors must then be increased and new classification techniques introduced. A sensor

P. Lukowitz, G. Kortuem, and K. Kunze (Eds.): EuroSSC 2010, LNCS 6446, pp. 42–55, 2010.

placed by the wrist is preferable when trying to distinguishing everyday or fitness activities with similar lower-body, but significantly different upper-body movement, [1,4], e.g., distinguish walking from Nordic walking [15]. Sensors placed on both upper and lower body improve the classification accuracy [1,16]. Using other sensors than acceleration or inertial sensors was also investigated in e.g., [15], but it was shown that 3D-acceleration sensors are the most powerful sensors for activity recognition.

Recently model-based methods have been introduced as an alternative to signal-oriented methods for activity recognition ([22,23]). In [22] a model-based approach is compared to signal-oriented approaches and it is shown that model-based methods have the potential to increase the recognition rate by introducing new features. However, the data set in [22] was recorded from a very specific industrial scenario: 20 well-defined activities were performed during a quality inspection of a car production process. Therefore, further analysis is needed to prove the benefits of body-model derived features, especially for everyday and fitness activity recognition (resting, walking, cycling, etc.).

For model-based activity recognition, it is fundamental to use a reliable and accurate human body pose estimation. Therefore, in this paper an upper-body pose estimator is introduced from which upper-body related features are deduced and used in the activity classifier. The estimator fuses a biomechanical model of the upper body with measurements from inertial sensors to estimate the pose of the torso and the arms. A similar approach is described in [21] where the authors use inertial sensors and a kinematic model to monitor stroke patients rehabilitation process. The papers [17,19] describe motion capturing systems using inertial sensors with support of ultrasound and magnetic sensors for capturing the body pose.

This paper presents a model-based method for recognizing 9 everyday and fitness activities, which can be used for e.g., ‘in vivo’ monitoring of patients. A model-based and a signal-oriented approach is compared , and it is shown that combining the two methods can significantly improve the recognition accuracy for different activities. Furthermore, as for deriving body-model related features, this paper goes beyond previous work: instead of just concatenating the orientation information of the intertial sensors, as presented in [22] and [23], a kinematic upper-body model is used for the upper-body pose estimation.

2 Data Collection

To obtain data, 6 Colibri inertial measurement units (IMU) from Trivisio [18] were used. The sensors are lightweight (22 g without cable) and small ($30 \times 30 \times 13$ mm). Each IMU contains a 3-axis MEMS accelerometer, a 3-axis MEMS gyroscope, and a 3-axis magneto-inductive magnetic sensor, all sampled at 100 Hz. The accelerometers have a resolution of $0.038\,\mathrm{ms}^{-2}$ in the range ± 16g, the gyroscopes $0.37°$/s in the range $\pm 1500°$/s, and the magnetometers have a scale of $\pm 1100\,\mu$T with a resolution of $0.4211\,\mu$T. The inertial sensors and magnetometers are individually calibrated for offsets, scale factors and alignment errors using the method of [5] and [7], respectively. Of the 6 IMUs used for data collection, 5 were placed on the upper body (cf. Section 4.1 for exact sensor placement) and one was shoe-mounted.

As pointed out in Section 1, data was collected from everyday and fitness activities. A criterion for selecting activities was to have activities differing mainly only in upper-body movement (like walking and Nordic walking), so that there is a high demand on good features derived from the upper-body pose estimation. Finally, data of 9 different activities were recorded: lying, sitting, standing, walking, Nordic walking, running, cycling, ascending stairs, and descending stairs. Sitting mainly consists of computer work, while standing consists of standing still or standing still and talking. Walking and running were performed both indoors and outdoors, while cycling was performed outdoors with a real bike. Nordic walking was performed outdoors on asphaltic terrain, using asphalt pads. Approximately 2 h of data was collected from the 9 different activities with one subject (male 26 years, BMI 25.66).

3 Signal-Oriented Activity Recognition

3.1 Feature Extraction

For the activity classifier that uses signal-oriented features only, acceleration data derived from the 3-axis accelerometers is used. Standard signal features were calculated over a window of 512 samples (about 5 s of data), in both time and frequency domain. Time-domain features were mean, median, standard deviation, peak acceleration and energy. For the frequency-domain features, the DC component of the signal was first removed, then the power spectral density (PSD) was calculated. Frequency-domain features were peak frequency of the PSD, power ratio of the frequency bands 0–2.75 Hz and 0–5 Hz, energy of the frequency band 0–10 Hz and spectral entropy of the normalized PSD on the frequency band 0–10 Hz.

The extracted features of the 3-axis accelerometers are computed for each axis separately, and for the 3 axes together, too. As for the window size, to receive a characteristic window, at least 2–3 s of data is necessary for certain activities, such as walking or ascending stairs. Hence, a window size of 512 samples was chosen — which also enables efficient implementation of the FFT — with a window shifting of 0.25 s.

3.2 Feature Selection

Features were selected based on both visual and statistical analysis. To compare and select features for distinguishing different groups of activities (like resting vs. non-resting activity classes), distribution bar graphs as presented in e.g., [15] were used. To identify for each activity the feature having the best performance in discriminating the corresponding activity from all the other activities, the measure presented in [8] was utilized. The K-means algorithm with $k = 100$ clusters was used for clustering the different features. The fraction for each cluster and activity was then computed, and the cluster precisions for each activity were obtained from the fractions, as presented in [8].

The cluster precisions can also be used to identify the best features to separate 2 groups of activities, or just to find the best feature to distinguish 2 different activities. Questions like "what is the best feature to separate running from all other activities including footsteps?" or "what is the best feature to distinguish lying and sitting?" can be easily answered by using the cluster precisions method only on the 2 groups of activities as a binary classifier. The selected signal features for classification are the following:

1. peak absolute value of the up-down acceleration measured on the shoe-mounted sensor (this feature is basically used to indicate footsteps)
2. mean value of the up-down (transversal) acceleration measured on the chest sensor
3. energy of the forward-backward (horizontal) acceleration measured on the shoe-mounted sensor
4. median value of up-down (transversal on initial position) acceleration measured on the right wrist sensor
5. energy of the frequency band 0–10 Hz of the up-down (transversal) acceleration measured on the chest sensor
6. standard deviation of the forward-backward (coronal on initial position) acceleration measured on the right wrist sensor
7. spectral entropy of the normalized PSD on the frequency band 0–10 Hz of the accelerations summarized for the 3 axes measured on the right wrist sensor
8. mean value of left-right (sagittal on initial position) acceleration measured on the right wrist sensor

3.3 Activity Classification

The selected features are used as feature vectors in a classifier to determine the currently performed activity. For activity classification, different approaches exist and yielded good results, such as SVMs, Naive Bayes classifiers, HMMs, ANNs or Boosting. Here the decision fell on custom decision trees, as they also have been successfully applied to activity recognition in previous work [1,2,3,4,11,15]. The advantages of decision trees include low computation requirements and a simple implementation. A custom decision tree can also be used to observe the effect of selecting different features by keeping the structure of the tree and only changing the features in the decision nodes. This method was used to compare the signal-oriented features with the model-based ones (cf. Section 5).

The custom decision tree is built using a priori knowledge about the activities, inspection of the distribution bar graphs, and the results of the cluster precisions method. The structure of the tree is depicted in Figure 1. The tree has 8 binary decision nodes and 9 leaf nodes, the latter representing the activities. The numbers in the decision nodes correspond to the selected features for each decision, as described in Section 3.2. The first decision node divides all activities into activities with and without footsteps, all other decisions are used to separate one activity from the remaining other activities. The tree has been constructed

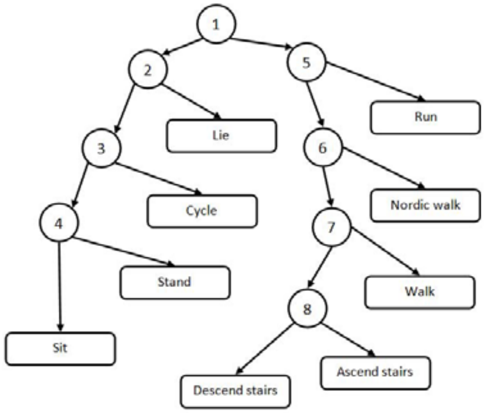


Fig. 1. Structure of the custom decision tree

so that "Sit" and "Descend stairs" are the default activities, so if the current activity shows no matching to any others while passing the decision tree, it falls through to this default classes. Classification results are shown and discussed in Section 6.

4 Upper-Body Pose Estimation

The pose and motion of the upper body are contained in the measurements from the five IMUs that are attached to it (cf. Section 2). The measured accelerations, angular velocities, and magnetic fields are compared to predictions based on a simple biomechanical body model, and this way the information is obtained with an extended Kalman filter (EKF) [10]. First the biomechanical model used for this and the sensor calibration needed are described. Then, it is explained how these are brought together to determine the pose kinematics using model based sensor fusion.

4.1 Biomechanical Body Model

A schematic of the used biomechanical upper-body model is depicted in Figure 2 (left). It consists of five rigid bodies (bones: upper arms, forearms, and torso) connected by anatomically motivated restricted joints. The right side of the figure shows how an IMU is attached to each segment on a test subject. The torso and shoulder joints are modeled with three degrees of freedom (DOF), whereas the elbow is restricted to two DOF.

Within the global reference frame, G, each individual joint constitutes its own coordinate system: the torso frame, T, the left and right shoulder frames, S, and the elbow frames, E. In the nominal pose, where all coordinate systems are aligned with the global frame, the upper body is upright, with the arms vertically

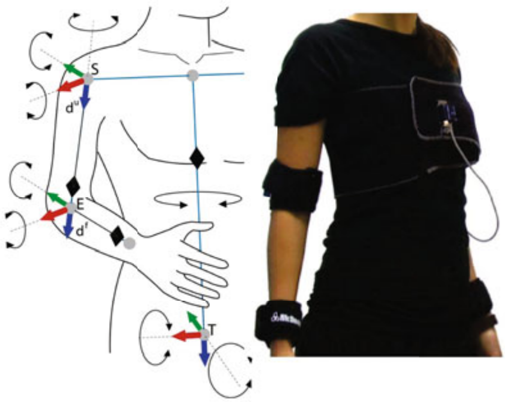


Fig. 2. Biomechanical body model and fixation of IMUs on a test subject

and the thumbs pointing forward. A coordinate system is also attached to each IMU: the torso IMU, I^t, the upper arm IMUs, I^u, and the forearm IMUs, I^f[1].

The relative positions that are important for the upper-body pose estimation are: the positions of the elbows, d^e, and the upper arm IMUs, d^u, relative to the shoulders, and the positions of the forearm IMUs, d^f, relative to the elbows. The torso is during estimation modeled as dimensionless. Hence, both the shoulders and the center of rotation in the torso are assumed to be collocated in one point.

The procedure for calibrating the relative rotations of the different IMUs with respect to the body is described in Section 4.2. The relative positions are determined by measuring the distances along the arm segments.

The biomechanical model does not take into account the effects of soft tissue, or the non-orthogonality of the two rotation axes in the elbow. However, it has proven to be a sufficient description of the human body for the purpose of this paper.

4.2 Calibration

To be able to use the IMU measurements to estimate the body pose, the way the sensors are attached to the segments must be known. The relative sensor orientations can be calibrated in two steps. First the torso sensor is calibrated, and then the orientation of the arm sensors aligned.

The orientation of the torso sensor is obtained using two accelerometer measurements, y_1^a and y_2^a, standing in the reference pose and bending forward, respectively. The direction of the z axis in the sensor system is then given by y_1^a (the gravity). Bending forward is a rotation around x, hence given by $y_2^a \times y_1^a$. The y axis then follows in order to obtain a right handed coordinate system. The three directions specify R_{I^tT}.

[1] The abbreviations are used as subscripts to indicate, in which frame a quantity is resolved. For transformations, subscripts with two letters denote the mapping.

Once the torso has been calibrated, the sensors on the arms can be calibrated. This is done with a measurement in the nominal pose, and then aligning the z axes with gravity and using the magnetometer measurement of the torso as a reference direction to determine the x and y axes. This yields the rotations R_{I^uS} and R_{I^fE}.

As an alternative to using the magnetometer measurement to calibrate the arm sensors it is possible to use a two step procedure only using acceleration measurements [13]. The arms then need to be held horizontally in the second step. A drawback with this method is that there is no guarantee that the arms get aligned to each other or the torso.

4.3 Sensor Fusion

The EKF is an elegant algorithm for fusing the angular velocity, acceleration, and magnetic field measurements provided by the five IMUs. Moreover, it allows for recovering the upper-body pose based on forward kinematics only.

In this work, three separate EKFs are used: One EKF estimates the absolute torso orientation, R_{TG}, from the torso IMU measurements. It uses a standard attitude and heading reference system approach [6] as implemented in most of the commercially available IMUs [20,9].

The remaining two EKFs estimate the movements of the left and the right arm from the respective shoulder and forearm IMUs. These EKFs are losely coupled with the torso EKF, insofar as they use its results as control input to the measurement model. The two arms are estimated independently using the same state-space model, which is detailed in the following two sections. Afterwards, the initialization of the complete EKF structure is described.

The upper body tracking assumes gravity being the only force acting on the IMUs. If gravity is not the dominant force over longer periods of time, the estimated pose may be negatively affected.

Arm system model. Knowing the segment lengths, the arm pose is fully specified by the shoulder rotation, R_{ST}, and the elbow rotation, R_{ES}. Together with the calibrated quantities (cf. Sections 4.1 and 4.2) and the orientation obtained from the torso EKF, R_{TG}, this information is sufficient to compute the IMU orientations:

$$R_{I^uG} = R_{I^uS} R_{ST} R_{TG} \tag{1a}$$

$$R_{I^fG} = R_{I^fE} R_{ES} R_{ST} R_{TG} \tag{1b}$$

and positions:

$$I^u_G = R^T_{TG} R^T_{ST} d^u \tag{2a}$$

$$I^f_G = R^T_{TG} R^T_{ST} (d^e + R^T_{ES} d^f) \tag{2b}$$

with respect to the global frame.

In order to obtain a minimal parametrization and to be able to constrain the elbow joint to two DOF, Euler angles are used to represent the configuration of the joints. Hence, the system state, $x = [\theta, \dot{\theta}, \ddot{\theta}]$, comprises the joint angles, $\theta = [\theta_{ST}, \theta_{ES}]^T$, with $R_{ab} = \mathrm{rot}(\theta_{ab})$, their velocities, $\dot{\theta}$, and their accelerations, $\ddot{\theta}$. Assuming constant angular acceleration, the dynamic model is:

$$\begin{bmatrix} \theta \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix}_{t+T} = \begin{bmatrix} I & T & \frac{T^2}{2} \\ 0 & I & T \\ 0 & 0 & I \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix}_{t} + \begin{bmatrix} \frac{T^2}{2} \\ T \\ I \end{bmatrix} v_t \tag{3}$$

where v_t denotes zero-mean Gaussian process noise and T denotes the sample time. Variances of $0.01\,\mathrm{rad/s^2}$ for the shoulder joint and $0.04\,\mathrm{rad/s^2}$ for the elbow joint worked well for the motions conducted in the experiments.

Arm measurement model. The measurement models relate the measured angular velocities, y^ω, accelerations, y^a, and magnetic fields, y^m, in the local IMU frames, I, to the state.

The accelerometers measure a combination of body acceleration, $\ddot{I}$, and acceleration due to gravity, g, in the local IMU frame. Hence, the acceleration measurement model is:

$$y^a = R_{IG}\big(\ddot{I}_G(\theta, \dot{\theta}, \ddot{\theta}) - g_G\big) + e^a. \tag{4a}$$

Here, the body acceleration in the global frame, $\ddot{I}_G$, is a function of θ, $\dot{\theta}$, and $\ddot{\theta}$. It results from differentiating (2) with respect to time twice and transforming the result to the local IMU frame using R_{IG}. The latter is obtained from (1).

The gyroscopes measurement model is:

$$y^\omega = \omega_I(\theta, \dot{\theta}) + e^\omega, \tag{4b}$$

where the angular velocity in the IMU, ω_I, is obtained by transforming $\dot{\theta}$ to the local frame. The tranformation can be derived from the relation $\mathrm{S}(\omega) = (R_{IG}\dot{R}_{IG}^T)$, where $\mathrm{S}(\omega)$ is the skew-symmetric matrix of ω [14].

When using only the inertial sensors, the rotation around the global z axis is not observable. Hence, sensor noise, and calibration and model errors cause a drift in the estimate. In order to correct for this, the magnetometers are used as aiding sensors. They provide a common forward direction, m_G. The respective measurement equation is:

$$y^m = R_{IG}(\theta) m_G + e^m. \tag{4c}$$

To simplify the equation and lessen the influence of magnetic disturbances (4c) is reduced to the heading direction. This is achieved by comparing the $\arctan(y, x)$ of both sides of the relation.

In (4), e denotes mutually independent zero-mean Gaussian measurement noise. Reflecting the actual noise of the inertial sensors, $0.002\,\mathrm{rad/s}$ variance is used for the gyroscopes and $0.02\,\mathrm{m/s^2}$ for the accelerometers. For the heading direction the variance was set to $9°$, which worked well in the experiments.

Initialization. Assuming that the subject is standing still, the initial rotation of the torso is fully determined by the acceleration and magnetic field measurements of the torso IMU. During initialization, the global heading direction is reset to the current one.

By further assuming that the rotation around the upper arm is zero, the joint angles of the arms are fully determined by the accelerometer measurements of the upper and forearm IMUs.

5 Model-Based Activity Recognition

The upper-body pose estimation presented in Section 4 provides 3-DOF torso orientation, 3-DOF shoulder joint angles and 2-DOF elbow joint angles. Features from the pose estimation are computed, selected and used for activity recognition in this section. A classifier using both signal-oriented and model-based features is desribed, too.

5.1 Feature Extraction and Selection

From the torso orientation and joint angles mean, median, standard deviation, and minimum and maximum value were computed as time-domain features, and energy and spectral entropy of the frequency band 0–10 Hz as frequency-domain features. These features were computed for each DOF separately and in the five joints for all dimensions together. Furthermore, features were also computed from the corresponding left and right shoulder and elbow joint angles together. Moreover, the position of the elbows and hands in the torso reference system were computed using the joint angles, and the aforementioned features extracted on the position information in all 3 dimensions. Finally, as an additional high-level feature, an energy equivalent of the two arm movements was introduced and added to the feature set. All features were computed over a window of 512 samples, as defined in Section 3.1.

The same feature selection methods were applied on the model-based features as on the signal-oriented ones, as described in Section 3.2. The structure of the custom decision tree also remained the same, only the features used in the decision nodes were changed. The selected features for the model-based classifier, numbered accordingly to the decision nodes of Figure 1, are the following:

1. energy equivalent of the arm movements
2. median value of the torso orientation in sagittal plane (bending forward-backward)
3. median value of the sagittal elbow joint angle added for the left and right arm
4. median value of the horizontal position of the right hand
5. mean value of the sagittal elbow joint angle added for the left and right arm
6. spectral entropy of the normalized PSD on the frequency band 0–10 Hz of the coronal torso orientation (bending left-right)

7. energy equivalent of the arm movements
8. standard deviation of the sagittal elbow joint angle added for the left and right arm.

5.2 Combining Signal-Oriented and Model-Based Features

When analyzing the selected model-based features, it is reasonable that for certain decision nodes they can improve the performance on separating the two groups of activities. For instance, the model-based feature of the first decision node uses a completely different characteristic of the activities to be separated: instead of using the information whether footsteps were detected or not, it uses the fact, that significant movement of the arms is present while performing walking related activities. Although the accuracy of the separation decreases from 99.21% to 98.29% compared to when using the signal-oriented feature, when using the two features together, an improvement of the classification performance is expected.

The combined classifier uses the same structure for the custom decision tree as the classifiers of Section 3.3 and Section 5.1. However, instead of just training for one, either signal-oriented or model-based feature, the decision in e.g., the first node is as following: use the model-based feature (the energy equivalent of the arm movements) if its value is under a trained limit and classifiy the sample for the group of the activities lying, sitting, standing, and cycling, otherwise use the signal-oriented feature (in this case the peak absolute value of the up-down acceleration measured on the shoe-mounted sensor) for the classification of the current sample. Section 6 shows the results of the combined classifier, and discusses for which decision nodes the incorporation of the model-based features improved the classification.

6 Results and Discussion

Table 1 shows the confusion matrix of the signal-oriented classifier. The overall performance of the classifier is 96.08%. For certain activities, such as lying or running, this classifier already provides a highly reliable classification. Also, the separation of cycling from the resting activities is accurate by using the shoe-mounted sensor data. However, more than 7% of the cycling-samples wrongly indicate footsteps, which lead this samples to the right side (cf. Figure 1) of the decision tree and thus resulting a performance of only 92.61% for the activity cycling.

The confusion matrix also shows some misclassifications between sitting and standing, and between ascending and descending stairs, where improvement is expected by using model-based features in addition. Ascending and descending stairs also introduced difficulties in the classification of walking. The selected signal feature for discriminating walking uses the fact, that is is usually a more regular movement than going upstairs or downstairs. However, the detection accuracy could benefit from introducing model-based features.

Table 1. Confusion matrix of the signal-oriented activity classification

	Recognized activity								
Annotation	**Lie**	**Sit**	**Stand**	**Walk**	**Nordic walk**	**Run**	**Cycle**	**Ascend stairs**	**Descend stairs**
Lie	3106	0	0	0	0	0	0	0	0
Sit	0	4235	18	0	0	0	0	0	0
Stand	0	42	3007	0	0	0	0	0	0
Walk	0	0	0	3402	0	0	0	97	103
Nordic walk	0	0	0	0	1364	0	0	244	0
Run	0	0	0	0	0	2304	0	0	0
Cycle	0	0	0	0	1	0	1943	0	154
Ascend stairs	0	0	0	103	93	0	28	1531	0
Descend stairs	0	0	0	4	0	0	0	18	1299

Table 2. Confusion matrix of the model-based activity classification

	Recognized activity								
Annotation	**Lie**	**Sit**	**Stand**	**Walk**	**Nordic walk**	**Run**	**Cycle**	**Ascend stairs**	**stairs**
Lie	2977	0	0	0	4	3	122	0	0
Sit	32	3874	32	0	0	0	315	0	0
Stand	0	42	2971	0	1	0	28	5	2
Walk	0	0	0	2873	407	0	0	147	175
Nordic walk	0	0	0	286	1321	0	0	0	1
Run	0	0	0	0	0	2304	0	0	0
Cycle	0	194	25	55	9	38	1687	0	90
Ascend stairs	0	0	0	284	64	46	0	1361	0
Descend stairs	0	0	189	142	224	0	0	0	766

Nordic walking was detected with 84.83% and walking with 94.45%. For discriminating Nordic walking from the remaining activities, the increased arm motion in the forward-backward direction was used. However, the characteristic of ascending stairs overlap in this respect, which implies the need for a better performing feature. Interestingly, no misclassifications were detected between walking and Nordic walking, what conforms to the requirement of a significantly increased arm movement while performing Nordic walking compared to normal walking.

The confusion matrix of the model-based classifier is shown in Table 2. The overall performance of the classifier is 87.18%. Before further analyzing the detection accuracy, it has to be mentioned, that the upper-body pose estimation failed on one part of the collected data, mainly locomotion activities performed indoors were affected. The magnetometers of the IMUs are sensitive to magnetic disturbances, which explains this inaccurate pose estimation. However, even for the indoor activities, the estimated torso orientation and joint angles were reliable in the majority of the samples, and thus useful for activity classification. Another limitation of the pose estimation method described in Section 4 is, that the used state-space model assumes that the upper body is not subjected to severe body accelerations. However, this barely effected the pose estimation on the collected data.

The detection accuracy of walking was 79.76% and for Nordic walking 82.15%. Most of the misclassified samples can be explained by the above mentioned inaccurate pose estimation while performing walking indoors. Interestingly, the best feature to discriminate Nordic walking (the feature of decision node number 6) was found the spectral entropy of the left-right torso orientation. This conforms to the requirement of an increased torso movement while performing Nordic walking, which results in a regular left-right bending of the torso, thus a low entropy of the the PSD.

Table 3. Confusion matrix of the combined activity classification

Annotation	Lie	Sit	Stand	Walk	Nordic walk	Run	Cycle	Ascend stairs	Descend stairs
	Recognized activity								
Lie	3106	0	0	0	0	0	0	0	0
Sit	0	4235	18	0	0	0	0	0	0
Stand	0	42	3007	0	0	0	0	0	0
Walk	0	0	0	3402	0	0	0	79	121
Nordic walk	0	0	0	0	1364	0	0	8	236
Run	0	0	0	0	0	2304	0	0	0
Cycle	0	0	0	0	0	0	2065	0	33
Ascend stairs	0	0	0	78	46	0	28	1603	0
Descend stairs	0	0	0	4	0	0	0	0	1317

Difficulties were encountered when distinguishing cycling and sitting, resulting in a detection accuracy of 80.41% and 91.09%, respectively. This difficulties are understandable, since the upper-body pose of these two activities is similar, and thus clearly highlight the need of a sensor placed on the lower body for recognizing everyday and fitness activities, in contrast to the upper-body dominant activities of the industrial scenario presented in [22,23].

The confusion matrix of the combined classifier is shown in Table 3. The overall performance of the classifier is 97.00%. For this classifier, the reliability of the pose estimation was first tested for each sample, due to the problems encountered for locomotion indoor activities. This investigation relies on the elbows' and hands' position, verifying first if the current positions are anatomically possible, and for each decision node if they are in the expected area. If the pose estimation turns out to be not reliable, only signal-oriented features are used to classify that sample.

Three decision nodes of the custom decision tree of the combined classifier only use signal-oriented features, the other five decision nodes benefit from model-based features. The former three nodes are, where lying, cycling and running are discriminated from remaining activities. As mentioned before, for discriminating cycling from resting activities (especially from sitting), a feature calculated on data of a lower body sensor is preferable. As for running, both the selected signal-oriented and the model-based feature performed highly reliable, no improvement was needed. Since the signal-oriented feature is applicable under all circumstances, even indoor when high magnetic disturbances are present, the combined classifier uses this feature.

For distinguishing standing and sitting, both the signal-oriented and the model-based features are used, but no improvements were achieved in the recognition accuracy as one can see by comparing the corresponding rows of Table 1 and Table 3.

Similarly, both features were used when distinguishing ascending and descending stairs, which solved the misclassifications compared to the signal-oriented approach, and thus increased the recognition accuracy of descending stairs from 98.33% to 99.70%.

Even more improvement on the classification results can be observed when incorporating model-based features for the decision nodes number 6 and 7. This results in less misclassification of ascending stairs into Nordic walking and walking, thus increasing the classification accuracy of ascending stairs from 87.24% to 91.34%.

The most significant improvement in recognition accuracy was observed when combining the footstep detection (signal-oriented feature) with the energy equivalent of the arm movements (model-based feature) to separate the activities in the first decision node. This leaded to a nearly perfect separation in this node, since resolving most of the misclassifications of the cycling-samples. The recognition accuracy of cycling was thus increased by nearly 6%, from 92.61% to 98.43%.

7 Conclusion and Future Work

In this paper, a model-based activity recognition method was introduced and applied to recognizing everyday and fitness activities. Model-based and signal-oriented approaches were compared and the benefits of body-model derived features for activity classification were proved. It was also shown, that the recognition accuracy can be significantly improved when adding model-based features to a signal-oriented classifier. The overall classification performance of the presented combined classifier showed an improvement of approximately 1% compared to the signal-oriented classifier. For certain activities the classification results improved even more significantly, with up to 6%. The presented results can be regarded as a proof-of-concept of improving activity recognition accuracy when incorporating body-model features. For the future, data collection over a longer period, and with multiple test subjects is planned. It also has to be mentioned, that the custom decision tree used as classifier does not necessarily provide the most suitable classifier for a given set of features, thus in future work, the solution space also should be exhaustively explored.

The model-based features were calculated from shoulder and elbow joint angles and torso orientation, provided by a biomechanical model based upper-body pose estimation. This pose estimation turned out to be reliable in most cases, and the classifier based only on this estimator showed a high recognition accuracy of over 87%. However, data collected while performing indoor activities also showed some limitations to the applicability of this approach, due to the sensitivity of the magnetometers to magnetic disturbances caused by ferreous material in the environment. Hence, if mainly indoor activities should be classified, hard and soft iron effects resulting from such disturbances need to be handled in the model.

Acknowledgements. This work has been performed within the project PAMAP funded under the AAL Joint Programme (AAL-2008-1). The authors would like to thank the project partners and the EU and national funding authorities for the financial support. For more information, please visit the website www.pamap.org

References

1. Bao, L., Intille, S.: Activity recognition from user-annotated acceleration data. In: Proc. 2nd Int. Conf. Pervasive Comput., pp. 1–17 (2004)
2. Bieber, G., Peter, C.: Using physical activity for user behavior analysis. In: PETRA 2008 (2008)

3. Ermes, M., Pärkkä, J., Cluitmans, L.: Advancing from offline to online activity recognition with wearable sensors. In: 30th Annual International IEEE EMBS Conference, pp. 4451–4454 (2008)
4. Ermes, M., Pärkkä, J., Mäntyjärvi, J., Korhonen, I.: Detection of daily activities and sports with wearable sensors in controlled and uncontrolled conditions. IEEE Trans. Inf. Technol. Biomed. 12(1), 20–26 (2008)
5. Ferraris, F., Grimaldi, U., Parvis, M.: Procedure for Effortless In-Field Calibration of Three-Axis Rate Gyros and Accelerometers. Sensor and Materials 7, 311–330 (1995)
6. Harada, T., Mori, T., Sato, T.: Development of a Tiny Orientation Estimation Device to Operate under Motion and Magnetic Disturbance. The International Journal of Robotics Research 26, 547–559 (2007)
7. Hu, X., Liu, Y., Wang, Y., Hu, Y., Yan, D.: Autocalibration of an Electronic Compass for Augmented Reality. In: ISMAR (2005)
8. Huynh, T., Schiele, B.: Analyzing features for activity recognition. In: sOc-EUSAI 2005, pp. 159–163 (2005)
9. InterSense, http://www.intersense.com
10. Jazwinski, A.H.: Stochastic Processes and Filtering Theory. Mathematics in Science and Engineering, vol. 64. Academic Press, Inc., London (1970)
11. Karantonis, D., Narayanan, M., Mathie, M., Lovell, N., Celler, B.: Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring. IEEE Trans. Inf. Technol. Biomed. 10(1), 156–167 (2006)
12. Long, X., Yin, B., Aarts, R.M.: Single-accelerometer based daily physical activity classification. In: 31st Annual International IEEE EMBS Conference, pp. 6107–6110 (2009)
13. Luinge, H.J., Veltink, P.H., Baten, C.T.M.: Ambulatory measurement of arm orientation. Journal of Biomechanics (40), 78–85 (2007)
14. Ma, Y., Soatto, S., Kosecka, J., Sastry, S.S.: An Invitation to 3-D Vision. Springer, Heidelberg (2003)
15. Pärkkä, J., Ermes, M., Korpipää, P., Mäntyjärvi, J., Peltola, J., Korhonen, I.: Activity classification using realistic data from wearable sensors. IEEE Trans. Inf. Technol. Biomed. 10(1), 119–128 (2006)
16. Pirttikangas, S., Fujinami, K., Nakajima, T.: Feature selection and activity recognition from wearable sensors. In: Youn, H.Y., Kim, M., Morikawa, H. (eds.) UCS 2006. LNCS, vol. 4239, pp. 516–527. Springer, Heidelberg (2006)
17. Schepers, H.M., Roetenberg, D., Veltink, P.H.: Ambulatory human motion tracking by fusion of inertial and magnetic sensing with adaptive actuation. Medical and Biological Engineering and Computing (48), 27–37 (2010)
18. Trivisio, http://www.trivisio.com
19. Vlasic, D., Adelsberger, R., Vannucci, G., Barnwell, J., Gross, M., Matusik, W., Popovic, J.: Practical Motion Capture in Everyday Surroundings. In: SIGGRAPH (2007)
20. XSens, http://www.xsens.com
21. Zhou, H., Hu, H.: Inertial sensors for motion detection of human upper limbs. Sensor Review 27(8), 151–158 (2007)
22. Zinnen, A., Blanke, U., Schiele, B.: An Analysis of Sensor-Oriented vs. Model-Based Activity Recognition. ISWC, pp. 93–100 (2009)
23. Zinnen, A., Wojek, C., Schiele, B.: Multi activity recognition based on bodymodel-derived primitives. LoCA, pp. 1–18 (2009)

Energy Efficient Cooperative Multimodal Ambient Monitoring

Michele Magno[1], Davide Brunelli[2], Piero Zappi[1], and Luca Benini[1]

[1] DEIS - University of Bologna viale Risorgimento 2, 40136 Bologna
name.surname@unibo.it
[2] DISI - University of Trento, Via Sommarive 14, Povo (TN), Italy
davide.brunelli@disi.unitn.it

Abstract. Wireless Video Sensor Networks (WVNs) are lively interest in the research community as flexible means for monitoring isolated areas. WVN effectiveness can be augmented when coupled with a network of low-power, low-cost Pyroelectric InfraRed (PIR) detectors to form a multimodal surveillance system. Autonomy is a key issue as battery replacement is often impractical and even using energy scavenging techniques, such as solar harvester, ad-hoc power management policies are essential to extend network lifetime. In this paper we propose a cooperative policy to manage power consumption of a WVN powered by solar scavengers and supported by a network of PIR sensors that perform a coarse classification of movements. A cost function is calculated by each VSN according to its available energy and information from the PIR network. Such functions are used by a distributed energy aware policy that selects the best VSN to observe the activity. This VSN locally analyzes the image and detects whether or not it represents a person, and only this information is forwarded to users. The effectiveness of this technique is evaluated through simulation and compared to an approach presented in previous work. Results show an increase in system lifetime without any loss in people detection ratio.

Keywords: Wireless Video Sensor Network, Multimodal Surveillance, Pyroelectric InfraRed, Solar Harvester, Power Aware Design.

1 Introduction

Recent years are witnessing the ever-growing demand for security in both public and private spaces. This feeling has pushed for the development of video surveillance and other security-related applications. As new generation of low-power, low-cost devices hits the market it is likely that new scenarios were a large number of cameras are embedded in the environment will emerge.

Thanks to their flexibility and ability to provide accurate real-time visual data *Wireless Video Sensor Networks* (WVN) are gaining credits. A WVN is made up of several wireless *Video Sensor Nodes* (VSNs) and each of them embeds a low-power imaging sensors, processors, and communication units to survey the Area of Interest (AoI). Power-aware design and maximization of the sensor network lifetime become the foremost objectives. [4,6].

P. Lukowitz, G. Kortuem, and K. Kunze (Eds.): EuroSSC 2010, LNCS 6446, pp. 56–70, 2010.

Typical approaches for energy consumption reduction in Wireless Sensor Networks (WSN) include: selection of low-power components, use of improved wireless protocols [17] and adapting parameters such as clock rate [12] or sample rate [9].

Exploiting renewable energy resources in the devices's surrounding is an alternative solution to increase nodes lifetime [15]. In particular, photovoltaic (PV) harvesters are good candidates to achieve perpetual operation of WSN [2]. Unfortunately if the power consumption of a device can have an estimated runtime of a certain interval in the future, the power generated by a PV module changes nonlinearly under varying temperature or solar irradiance. Techniques which automatically tune the operating point of the solar cell should be considered to provide the maximum output power, since they lead to several benefits such as: the possibility to use smaller PV modules, to reduce the capacity of the energy reservoir, or to allow higher power consumption operations onto a sensor node.

The major constraint when dealing with circuits for high efficient energy harvesting is that implementing maximum power point tracking techniques using small-size PV modules is practicable only if the power consumed by the additional hardware is considerably lower than the amount of output power that it gains. Thus the area of the deployment and the availability of the environmental energy need particular attention at design time when an estimation of the energy intake during the day or along a year is fundamental.

These approaches try to address the power consumption issue by extending single nodes lifetime. However, WVNs prompt for the development of high level *Power Management* (PM) policies. For example in [20] the author proposes two scalable and flexible techniques for WVNs power management by considering the content of the video data sensed both locally and by other video nodes within the network.

If we consider the energy issue at a network level we can exploit the use of heterogeneous network. Surveillance, as well as target tracking and classification, are classical applications which require global information of a certain spatial-temporal region and exploiting Multi modal-sensors is a promising approach to increase effectiveness of such systems [3]. On the other hand, generally sensor nodes only has a local view and spreading global information increases the communication traffic and the overall power consumption.

In addition, the redundancy provided by a mesh of heterogeneous nodes can be used to perform power-performances trade-off. The typical approach here is to support high-power CMOS imagers with a mesh of low-power, low-cost sensors densely spread in the environment [7]. While the former are kept into a low power state, the latter operate as a trigger to provide continuous area monitoring. In Boettcher et al. [1] low-power acoustic sensors are used to detect position of moving vehicle through a time-difference of arrival technique. This information is used in conjunction with an imager used to take an image of the vehicle and send it to the base station. Another example can be found in the work of Wang et al. [19] in which a WVN is supported by a network of microphones. The latter are used to provide an indication of the distance of a vehicle from the video sensor node. This information, together with the recognition accuracy of the video sensor node estimated at training time and the actual energy of the video sensor node, is used to evaluate a cost function used by a cluster head to select which video sensor node should be turned on.

In this paper we present a combination of PM techniques optimized for multi modal surveillance systems:

- a set of wireless VSN are used in conjunction with a network of low-cost, low-power Pyroelectric InfraRed (PIR) sensors to detect presence of people moving along a path in an outside area;
- the architecture is scalable and can be extended to an ambience of any size and shape. The minimum cluster of node is composed by 2 VSNs used in conjunction with a network of low-cost, low-power Pyroelectric InfraRed (PIR) sensors to detect presence of people moving along a path in an outside area;
- particular surveillance applications are interested in distinguishing the presence of human bodies rather than objects or animals in the field of view before generating alarms or sending information through the wireless link. Furthermore we implement algorithms capable of processing images, to detect the particular target and finally to send only the image that shows the face of the person. It is a particular needs for unobtrusive video-surveillance solutions which has to handle both security and privacy issues, guaranteeing to not process or record private data, while still detecting and identifying potential threats;
- the video sensor nodes are powered by a solar scavenger using a 70 cm^2 photovoltaic panel, to guarantee the maximum energy autonomy of the systems and flexibility for the reuse of the system or the adjustment of the deployment.
- to guarantee a balanced energy usage a trade-off between energy availability and quality of the service is adopted and VSNs are activated only when they can provide a useful contribution at the minimum energy expenditure. Thus a bidding-like protocol is engaged to select the most suitable offer to perform the image analysis.

The PM techniques are performed in a distributional manner, as constantly active nodes do not feasibly meet the energy requirements. We use the information from the PIR sensor network to activate only a subset of wireless VSNs enhancing an efficient collaborative approach. In fact, the pir-network is capable to detect and to estimate the position and direction of movement of the people along the track and therefore it identifies which VSN faces the persons. When the PIR sensor network detects a body moving in the area of interests it broadcasts a message to the VSNs with an indication of the body presence and direction of movement. According to this information and the available energy provided by the harvesting system, each VSN calculates a cost function and broadcast this value to the other VSNs. Each node compares its own cost function with the ones received from the other nodes of the network. The node with higher cost function wakes up and monitor the area of interest. This PM policies guarantees that for any passage at least one node wakes up even in presence of message losses. At the same time we can keep the majority of VSNs into a low power state to preserve or replenish their energy storages.

Our approach is similar to the one presented in [19]. However our distributed policy is more robust to nodes failure and messages loss. In fact in the work of Wang et al. if the cluster head fails the whole cluster is not able to operate until maintenance is performed. Furthermore, if node communication is compromised at a certain time, the nodes of the network can not be waken up to classify object passing by. In our work, instead, all nodes check locally if they are the one that should be turned on, therefore even if communication among nodes is not possible, in the worse case, all VSNs will wake up and analyze the image.

The rest of the paper is organized as follows. Sections 2 and 3 presents the WVN and the PIR sensor network that compose our system. Section 4 explains how the two sensor networks are used in conjucntio and the distributed power management techniques. Section 5 describes our network simulation and compares our approach with the case presented in [13]. Finally Section 6 concludes the paper providing further comments on the results we achieved and comparison with the state of the art.

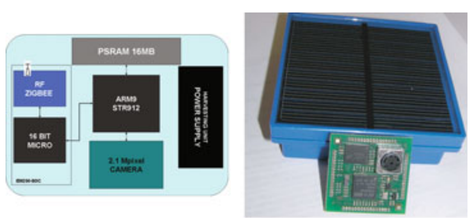
2 Video Sensor Nodes Description

The hardware architecture of the solar-powered video sensor is composed of four main modules (see figure 1) and designed to achieve low power consumption of the overall system. Each module can autonomously operate in different states to save energy when its contribution is not needed.

The core of the video node consists of an ARM 9, STR912F microprocessor with 16 MBytes of external RAM. It provides a set of powerful single-cycle DSP instructions and it is responsible to acquire and classify images from the CMOS image sensor in addition to node components management. This MCU offers configurable and flexible power management controls to match system requirements, in fact power consumption can be dynamically managed by adjusting MCU operating frequency. For example typical current consumption in RUN mode is about $1,7\,mA/MHz$ and only a few mA in sleep mode. To further reduce MCU power consumption most of its peripheral interfaces have been disabled.

The vision module includes a 2.1 Mpixel SXGA CMOS color digital camera targeted for mobile applications featuring low-size and low-power consumption.

Wireless capabilities of the VSN are provided by EM250 modules from Ember. This module is a SOC that integrates in the same package a programmable



a) b)

Fig. 1. a) Video sensor node architecture. b) Prototype of the video sensor node and solar cell.

Table 1. Power consumption of the video sensor node

Component	Power $[mW]$
ARM9 (RUN mode)	450
ARM9 (IDLE mode)	49,5
ARM9 (SLEEP mode)	15
Video sensor (ON mode)	165
Video sensor (IDLE mode)	23
TX/RX module (ACTIVE mode)	90
TX/RX module (IDLE mode)	17
Solar Harvester	0,98
Video Node (Active)	650
Video Node (Sleep)	56

16-bit microcontrollers with 128kB of Flash program memory and 5kB of RAM and a 2.4 GHz Zigbee compliant transceiver.

An overview of the node power consumption in different operating states il reported in Table 1, while a picture of our prototype is presented in Figure 1.

The whole system is powered though a solar energy harvester. The solar cell array has a nominal output power of 500 mW under full outdoor irradiance. A harvesting management circuit is used to extract the maximum power available under varying operating conditions. In particular it employs a Maximum Power Point Tracking (MPPT) mechanism to maximize the energy transfer between the solar cell and the reservoirs under a wide range of irradiance levels.

In fact, energy harvesting has recently emerged as the key technology which can enable systems to operate perpetually, ridding the periodical maintenance for battery replacement or recharging. In particular, energy harvesters convert energy from the surrounding environment replenishing energy storages. For perpetual applications energy-neutral conditions must be guaranteed and the energy balance defined as the difference between the average energy intake and the energy consumed by the system must be always positive (or at least zero). One of the primary issues to tackle is minimizing the power consumed by the harvester itself. Less power will require the circuit, faster will be the growth of the harvested energy in the accumulator. Another key design challenge is to optimize the efficiency of solar energy collection under non stationary light conditions. There are several methods and algorithms to track the MPP [5], we adopt one based on Fractional Open-Circuit Voltage (FOCV) which is the most used and cost-effective in micro solar harvesters. This method exploits the nearly linear proportional relationship between the operating voltage at MPP (V_{MPP}) of the main photovoltaic module and the open circuit voltage of a small additional PV array used as pilot-cell ($V_{pilot\,cell}$) under the same light L and temperature T conditions. In particular it does not use any additional DSPs or microcontrollers to adjust the working point and it is inexpensive to implement. Figure 2 describes the conceptual scheme of the solar scavenging circuit for implemented video sensor node and proposed in [2]. By measuring the pilot-cell voltage, the

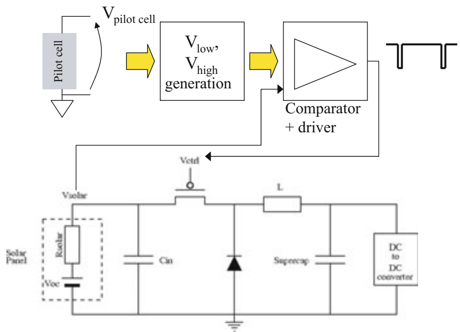


Fig. 2. Conceptual schematic of solar harvester: buck power converter and MPP tracker

circuit estimates the maximum power point of the main photovoltaic module and compares continuously it with the current operating point adjusting dynamically the power converter circuit. The implemented system exploits supercapacitors as energy storage devices, since they overcome many drawbacks of batteries that are critical in WSN applications and for long-live maintenance-free embedded systems. The harvester achieves an efficiency of the 80% and depending on solar irradiation can provide a maximum output power of about 500 mW while the power consumed by energy harvesting process is less than 1 mW.

2.1 Human Detection Application

A human body detection algorithm has been implemented on the video sensor node [13]. figure 3 highlights algorithm main steps.

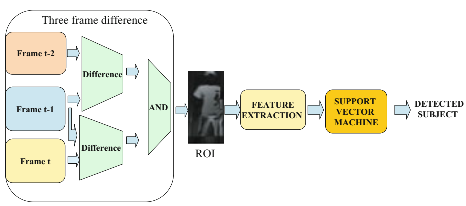


Fig. 3. Flow chart of people detection algorithm

The algorithms starts when the CMOS imager acquires a YCbCr 4:0:0, grayscale, 8-bit format, 240×160 pixel frame. Through the three-frame algorithm sub-image [10] the microprocessor isolates the event region-of-interest (ROI) (the part of the image where the body lays). The three-frame algorithm operates over three successive frames. It performs a pixel-by-pixel subtraction between the first and second frame, then another pixel-by-pixel subtraction between the second and third frame. The result of these operations gives an binary image with the detected objects. From the region of interest a features vector, with the average value over ROI columns and rows, is extracted and classified using a SVM algorithm [11]. The output is a simple binary (yes/no) report that the body in the

Table 2. Energy requirement

Task	Energy $[mJ]$	time $[ms]$
Three Frames Acquisition	354	561
Three Frame Difference	651	1057
Labeling	58	95
Feature Extraction	9	16
SVM	44	70

picture is human. This result is wirelessly sent to the sink station. Algorithm timing and energy consumption on our architecture are reported in Table 2.

3 Pyroelectric InfraRed (PIR) Sensor Nodes

Pyroelectric InfraRed (PIR) sensors are devices able to transduce changes in their temperature, due to incident infrared radiation, into an electric signal. A pyroelectric element behaves like a polarized planar capacitor whose charge varies according to $\Delta Q = A \cdot p \cdot \Delta T$ (where A is the area of the sensing element and p is the material specific pyroelectric coefficient). Typical PIR sensors embed 2 elements placed in series with opposite polarization. As a consequence when a body moves in front of the sensor 2 peaks, one positive and one negative, are produced (see figure 4).

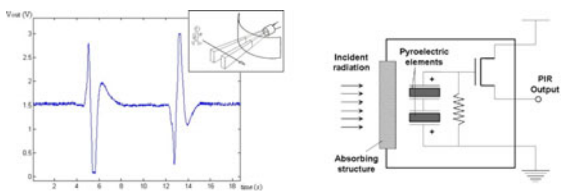


Fig. 4. PIR schematics and output when passages in the two directions (left to right and right to left) occur

PIR sensors are used in conjunction with an array of Fresnel lenses used to shape the sensor Field of View. Our prototype PIR sensor board has been designed using Commercial Off-the-Shelf (COTS) components. The detector is Murata IRA E710 [14] and the signal conditioning circuit is a double stage amplifier, which achieves a total gain of about 1400 and operates as a band-pass filter between 0.57Hz and 11Hz. This is a suitable range for detecting moving people [16]. Furthermore, it biases the output voltage at $\frac{V_{dd}}{2}$ when no movements are detected. The conditioning circuit board includes also a low power voltage regulator used to decouple power supply lines from the transceiver ones and a comparator used to generate a wake up signal when the board is in a low power state. The sensor and its conditioning circuits are hosted in the package of a PIR presence detector, IS-215T [8].

To form a PIR sensor node, this board is connected to a MotionBee Zigbee module that provides wireless connectivity with the other nodes of the network

Table 3. Power consumption of the PIR sensor node

Component	Power $[mW]$
Active Radio TX	48
Active Radio RX	37
Active Radio Off	13
Sleep	0.6
PIR board prototype	0.2

[18]. In our tests, the MotionBee on board accelerometer has been disabled to reduce power consumption.

An overview of the node power consumption in different operating states is reported in Table 3.

3.1 PIR Based People Tracking

In this paper the PIR sensor network should provide to the video system a coarse estimation of people position and direction of movement. For this reason we adopt the solution presented in [21]. In this setup the area of interest is covered by an array of PIR sensor nodes that are organized in small clusters. Each cluster is an autonomous network basic block and is made up of two sensors facing each other that locally detect body position and direction of movement through the classification of simple features (signal duration and peak to peak amplitude) extracted from PIR output. One of the two nodes act as a block manager. It receives the features from the other node and performs the classification step. This information is used in conjunction with the simple detection of the first peak direction (either positive or negative) that indicates the direction of movement (see figure 4).

A linear Support Vector Machine (SVM) classifier has been used to classify people position into three classes according to their distance from the two PIRs (see figure 5). According to the results presented in [21] linear SVM presents a good trade-off between correct position detection (86.06 %) and computational and memory cost (respectively, 6 multiplication, 6 sums and 2 max, and 6 bytes of Flash) and can be efficiently implemented by the low-cost low-power micro-controller that manage the PIR sensor nodes. The confusion matrix for the linear SVM classifier is presented in table 4.

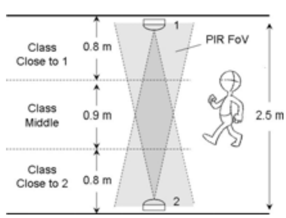


Fig. 5. Basic configuration used to estimate people position and distance

Table 4. Support Vector Machines classifier's confusion matrix

	classified as		
	close to 1	middle	close to 2
close to 1	166	32	0
middle	14	181	12
close to 2	0	29	190

As can be seen from this table, this classifiers present limited uncertainty since passages in proximity of one PIR sensor are never confused with passages close to the other one.

4 Cooperative Ambient Monitoring

In this paper we address the scenario where an outdoor, isolated Area of Interest (AoI) is covered by a heterogeneous network made up of our VSNs and PIR sensor nodes. While PIR sensors provide a coarse but continuous coverage of the AoI, VSN are used to identify and report relevant events such as people passages.

The nodes of the network are organized in clusters made up of 2 or 4 VSN and 2 PIR sensor nodes each arranged as presented in figures 6 and 6. The PIR sensor nodes are in the configuration presented in figure 5, while the VSNs point toward the PIR sensors. Each cluster monitors a small part of the AoI and is used in conjunction with other identical ones to cover bigger areas. Each cluster works independently from the others and, with the exception of the final people recognition result, wireless communication is performed only locally among the nodes of a cluster. In our scenario we assume that people move only along three passages, namely *Zone 1*, *Middle Zone* and *Zone 2*.

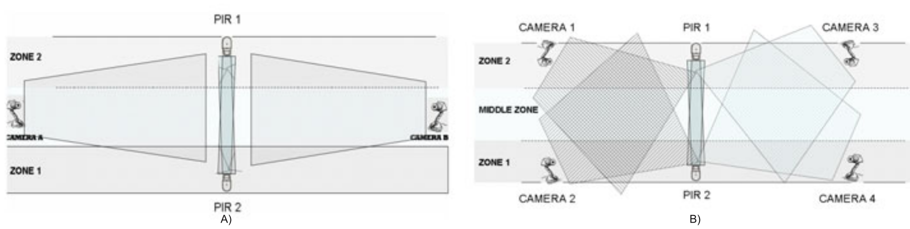


Fig. 6. A)Cluster with 2 VSN. B) Cluster with 4 VSN.

4.1 Multimodal Distributed Power Management

When no transit occurs, the sensor nodes of the network are kept into a low power state. Periodically the VSNs wake-up and poll the PIR manager for synchronization and indication of passages. As the PIR motes detect a transit, the direction of movement is evaluated as well as the zone the body is moving. This information is broadcast to the VSNs of the cluster.

Based on body direction and position and available energy, each VSN computes a cost function that represent the ability offered to wake-up the camera and to correctly identify the body. For this reason the cost function returns a higher value when the body is moving toward the VSN and in its field of view center.

The cost function has the following expression:

$$CF = \frac{E}{C} \cdot \frac{1}{P \cdot D} \cdot \gamma(E, C) \tag{1}$$

Where E denotes the actual energy available of the node, C the max capacity of battery, P and D are weights factors depending on body position and direction and $\gamma(E, C)$ is a non linear factor used to decrease the weight of nodes with low energy. The ratio of the energy used in comparison to the accumulator capacity represents an important parameter to trade off with the accuracy of the classification and the selection of the best camera.

Position and direction influence the CF and the performance during simulations. Tables 5, 6 and 7 present the optimal values.

As can be seen from tables 5 and 6 when a body is moving toward a VSN the value of D is much smaller. As a consequence the VSNs that see the face of the person are selected even if the others have much more available energy. When the two VSNs facing the front of the person are close to run out of energy the others result in a higher CF and can still provide some information on the people passing.

The value of the P factor is used only in the 4 VSNs cluster (in the 2 VSNs cluster is always 1) in order to distinguish between VSNs that face the front of the body and select the one that better points toward it.

Table 5. Values for the D factor for the 2 VSNs cluster

Direction	VSN A	VSN B
Left to Right	20	1
Right to Left	1	20

Table 6. Values for the D factor for the 2 VSNs cluster

Direction	VSN 1	VSN 2	VSN 3	VSN 4
Left to Right	20	20	1	1
Right to Left	1	1	20	20

Table 7. Values for the P factor (used only in the 4 VSNs cluster)

Zone	VSN 1	VSN 2	VSN 3	VSN 4
Zone 1	1	3	1	3
Zone Middle	1	1	1	1
Zone 2	3	1	3	1

Finally, the value of γ helps to reduce the probability that a node with low available energy is activated. This parameter assumes the following values:

$$if \quad \frac{E}{C} \quad \geq TH \quad \gamma = 1 \tag{2}$$

$$else \quad \gamma = 0.5 \tag{3}$$

Once the nodes compute their own CF they broadcast it to the other VSNs. A timeout is used in order not to stuck at if any other nodes message is lost. Than locally, each VSN check if its own CF is higher than the others. If any of the CF from the other VSNs is higher than the local one the VSN switches on is imager and start processing the image.

Since the camera mote knows at least its own CF, when the timeout expires, if any message has been received it consider itself the best VSN and starts acquiring the image. Therefore, if a transit is detected by PIR sensors at least one VSN turns on. Such approach is robust and guarantees that every event detections will be served. In fact, if some bidding messages does not arrive, the camera deems to have the highest CF providing an activation. In the worst case, more cameras will be activated after a single event. This guarantees to not miss any events, but on the other hand there is an overhead form the power consumption point of view.

To show the influence of the chosen parameters figure 7 presents the energy level of the four nodes in the hypothetical case where passages happen only from right to left in Zone 2 and no energy is harvested. In this case, if all VSNs have the same amount of energy, VSN 1 is the best candidate to detect the body. However as its energy decreases at a certain point VSN 2 will result in a higher CF and starts detecting transits. After a while also VSN 3 and 4 start processing images even if they can not see the face of the person since also VSN 2 energy is depletehd and its CF is lower than the one of VSN 3 and 4.

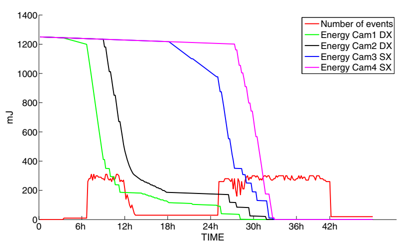


Fig. 7. Simulation of VSNs energy level when passages occur always in the same position

5 System Lifetime Evolution

We compared the two variants of the proposed approach with the case where the area of interest is covered by 4 VSNs equipped with a PIR sensor that produce a wake-up signal in presence of bodies [13]. In the latter case the camera and the ARM9 microcontroller are active when a person enter in the field of view of the imager. Once the node is awake, it processes the image from the CMOS camera in the same manner as described above. The size of the VSN field of view is modulated by changing the threshold above which the PIR sensor produces a wake-up trigger.

In this case VSNs do not have to broadcast the value of their cost function, thus they can save energy. However, the system do not use efficiently its resources. In fact the presence of other VSN that cover the same field of view is not taken into account and when a body moves in proximity of multiple VSNs all of them wake up. Moreover the work presented [13] do not consider people direction of movement in order to select which camera can better identify the subject.

5.1 Experimental Result vs. Camera with PIR

To evaluate the effectiveness of our approach, we simulated how the energy of the VSNs evolved as people passed across the PIR sensors.

People passages are modeled according to a profile of events that describes passages during 2 consecutive days in front our our lab (see figure 8). The energy intake from the energy harvesters has been modeled by measuring the incident solar light intensity measured during the same period (see figure 8). Figure 9 compares VSNs over the 2 days period time. All VSNs are equipped with a 40F supercapacitor to store energy from the solar harvester.

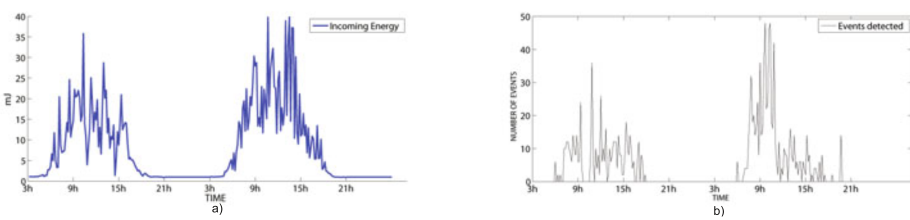


Fig. 8. a) Energy incoming from energy harvesters during 2 consecutive days. b) Number of events detected from PIR sensors in 2 consecutive days.

At simulation start, when no events are detected and no energy is harvested we see how the solution proposed in [13] presents less power consumption. This is related to the fact that no wireless messages are sent for synchronization and can be seen comparing the nodes energy levels on the box on the left part of figure 9.

As people start passing the solution with on board PIR (dotted line) quickly consumes its energy since all four VSNs are waken-up at every passage, despite person position and direction of movements. As a results after few hours the node has already exhausted his energy and can not monitor the area of interest anymore. To perform continuous operation in the proposed scenario, each sensor node should be equipped with a 300F supercapacitor.

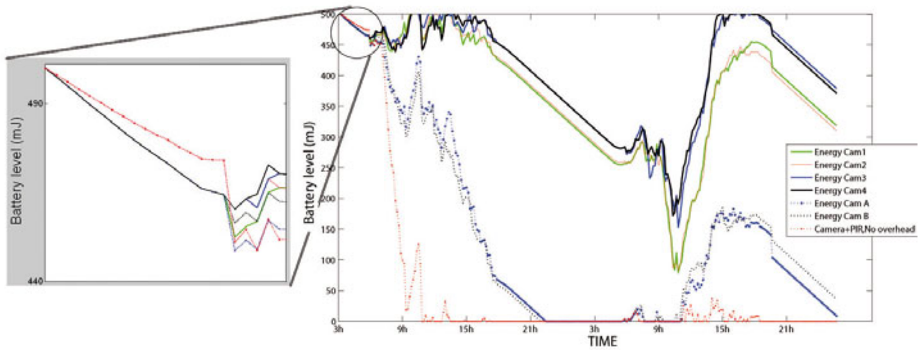


Fig. 9. Simulation 4 cameras against a camera with PIR and 2 cameras in 2 sides

The solution with 2 VSNs performs better than the previous one. As we can see from figure 9 computation is balanced among the two VSNs, thus the system is able to monitor the area of interest until evening when the gates of the building close. However, before the sun set the VSNs do not collect enough energy to operate all night long. Furthermore, as the second day starts, they need time to replenish their energy, so this system is not able to continuously operate during the second morning. To perform continuous operation each VSN should be equipped with a 102F supercapacitor.

Finally the solution with 4 VSNs is able to operate continuously with a 40F supercapacitor.

6 Conclusion

In this work we presented a multimodal ambient monitoring system where all system design steps are optimized for low-power consumption. This systems stems from the conjunction of 2 sensor network: a low power, low cost PIR based sensor network and a WVN. The former is responsible to provide a coarse, yet continuous monitoring, the latter is activated only when events are detected and aim at a better classification of the event itself. Nodes are organized in clusters made up of 2 or 4 VSN and 2 PIR sensors.

We proposed a distributed policy where each VSN, on the basis of the information from the PIR sensor network and its available energy, computes a cost function that is broadcast to the other nodes of the network. By comparing its own cost with the one received from the other nodes a VSN understands if it must monitor the event or it can stay into low power state. In the former case the VSN CMOS imager is turned on, an image is acquired and classified in order to understand if the event was generated by a person.

This is a robust policy, since for every event at least one VSN is activated, despite some messages may be lost. In fact each VSN has at least its own cost, therefore, if any other message is received it turns on. Furthermore, since this policy is distributed among the nodes of the network we do not have single point of failure for the whole system.

We compared our solutions with the one proposed in a previous work [13]. We showed that with our approaches we can achieve continuous operation with a 40F or 102F supercapacitor (4 or 2 VSN respectively) which are respectively 7.5 and 3 times smaller than the one needed for continuous operation of the system described in the previous work (300F).

References

1. Boettcher, P.W., Shaw, G.A.: Energy-constrained collaborative processing for target detection, tracking, and geolocation. In: Zhao, F., Guibas, L.J. (eds.) IPSN 2003. LNCS, vol. 2634, pp. 254–268. Springer, Heidelberg (2003)
2. Brunelli, D., Moser, C., Benini, L., Thiele, L.: An efficient solar energy harvester for wireless sensor nodes. In: Proceedings of the Conference on Design, Automation and Test in Europe, DATE 2008 (2008)
3. Datcu, D., Yang, Z., Rothkrantz, L.J.M.: Multimodal. In: Multimodal Surveillance: Sensors, Algorithms, and Systems, pp. 311–338. Artech House Publishers, Boston (July 2007)
4. Dong, Q.: Maximizing system lifetime in wireless sensor networks. In: Proceedings of the 4th International Symposium on Information Processing in Sensor Networks, IPSN 2005, Piscataway, NJ, USA, p. 3. IEEE Press, Los Alamitos (2005)
5. Esram, T., Chapman, P.: Comparison of photovoltaic array maximum power point tracking techniques. IEEE Transaction on Energy Conversion 2, 439–449 (2007)
6. Giridhar, A., Kumar, P.R.: Maximizing the functional lifetime of sensor networks. In: Proceedings of the 4th International Symposium on Information Processing in Sensor Networks, IPSN 2005, Piscataway, NJ, USA, p. 2. IEEE Press, Los Alamitos (2005)
7. He, T., Vicaire, P., Yan, T., Cao, Q., Zhou, G., Gu, L., Luo, L., Stoleru, R., Stankovic, J.A., Abdelzaher, T.F.: Achieving long-term surveillance in vigilnet, pp. 1–12 (April 2006)
8. Honeywell security and custom electronic. Is-215t datasheet (2008)
9. Junker, H., Lukowicz, P., Tröster, G.: Sampling frequency, signal resolution and the accuracy of wearable context recognition systems. In: McIlraith, S.A., Plexousakis, D., van Harmelen, F. (eds.) ISWC 2004. LNCS, vol. 3298, pp. 176–177. Springer, Heidelberg (2004)
10. Yoshinari, M.M.K.: A human motion estimation method using 3-successive video frames. In: Proc. of Int. Conf. on Virtual Systems and Multimedia GIFU, pp. 135–140 (1996)
11. Kerhet, A., Magno, M., Leonardi, F., Boni, A., Benini, L.: A low-power wireless video sensor node for distributed object detection. Journal of Real-Time Image Processing 2(4), 331–342 (2007)
12. Lee, Y.-H., Krishna, C.: Voltage-clock scaling for low energy consumption in real-time embedded systems, pp. 272–279 (1999)
13. Magno, M., Brunelli, D., Zappi, P., Benini, L.: A solar-powered video sensor node for energy efficient multimodal surveillance. In: Proceedings of the 2008 11th EUROMICRO Conference on Digital System Design Architectures, Methods and Tools, DSD 2008, Washington, DC, USA, pp. 512–519. IEEE Computer Society Press, Los Alamitos (2008)

14. Murata manufacturing. Pyroelectric infrared sensors (2005)
15. Paradiso, J., Starner, T.: Energy scavenging for mobile and wireless electronics. IEEE Pervasive Computing 4(1), 18–27 (2005)
16. Elmer, P.: Frequency range for pyroelectric detectors, http://www.perkinelmer.com
17. Shyi, C.-N., Weng, W.-F., Wu, H.-T.: Low power platform for wireless sensor network, vol. 2, pp. 417–421 (July 2005)
18. ST Microelectronics. Motionbee(tm) modules. Technical report
19. Wang, X., Bi, D., Ding, L., Wang, S.: Bidding and voting strategy for energy efficient collaborative target classification in wireless sensor networks. In: Proceedings of the 2008 Congress on Image and Signal Processing, CISP 2008, Washington, DC, USA, vol. 4, pp. 23–27. IEEE Computer Society Press, Los Alamitos (2008)
20. Zamora, N., Kao, J.-C., Marculescu, R.: Distributed power-management techniques for wireless network video systems, pp. 1–6 (April 2007)
21. Zappi, P., Farella, E., Benini, L.: Pyroelectric infrared sensors based distance estimation. To be published in Sensor 2008 (2008)

On the Use of Magnetic Field Disturbances as Features for Activity Recognition with on Body Sensors

Gernot Bahle, Kai Kunze, and Paul Lukowicz

Embedded Systems Lab
University Passau, Passau, Germany
first.lastname@uni-passau.de

Abstract. We investigate the use of magnetic field disturbances as features for motion based, wearable activity recognition systems. Such disturbances are mostly caused by large metallic objects and electrical appliances, both of which are often involved in human activities. We propose to detect them by subtracting angular velocity values computed from the changes in the magnetic field vector from gyroscope signals. We argue that for activities that are associated with specific objects or devices such features increase system robustness against motion variations, sensor displacement and inter user differences. On a previously published data set of 8 gym exercises we demonstrate that our approach can improve the recognition by up to 31% over gyroscope only and up to 17% over a combination of a gyroscope and 3D accelerometer. Improvements of 9.5% are also demonstrated for user independent training as well as for the case of displaced sensors.

A particularly interesting result is the fact that adding the magnetic disturbance features significantly improves recognition based on the vector norm of accelerometers and gyroscopes. The norm is often used when the orientation of the sensor is not known. This is common when using a mobile phone or other consumer appliance as a sensor.

1 Introduction

Gyroscopes and magnetic field sensors are often used modalities in activity recognition ([7,4,17,15]). In general, magnetic sensors are used to determine orientation (possibly together with an accelerometer) while the gyroscope helps filter out distortions due to disturbances in the magnetic field.

In this paper, we investigate a different way of using the information from a magnetic sensor. Rather than try to filter out the disturbances we specifically use them as features. This approach is aimed at activities that are related to motions in the proximity of fixed metallic objects with each activity being associated with a different object. The example considered in this paper are exercises on gym machines. Such machines are mostly metallic and differ significantly in the shape of metallic parts which users hands come close to.

P. Lukowitz, G. Kortuem, and K. Kunze (Eds.): EuroSSC 2010, LNCS 6446, pp. 71–81, 2010.

Metallic objects disturb the earth magnetic field in a way determined by their shape and composition. This means that while the absolute value of the magnetic field signal is predominantly given by position and the specific motion that the user has been performing, the disturbances are more dependent on the object itself (although position and motion also play a role). As will be shown in the paper this reduces the dependence of the recognition performance on sensor placement and motion variations that occur when looking at different users.

We demonstrate this on a data set recorded from gym exercises on different machines that we have previously used to study the effect of sensor placement variations on activity recognition [9].

1.1 Related Work

In general, gyroscopes are used together with magnetic field sensors to filter out magnetic disturbances. Levi et. al. present a patent describing how to improve the magnetic heading information using a gyroscope [12]. Ladetto et. al improve a dead-reckoning system fusing the gyro and compass information [10]. Young et al. use the magnetic field as source of orientation data in motion capturing applications ([16]). The application scenarios for magnetic field sensors range from life recording, over music performances to novel interface designs ([7,3,14,5]). Still the use is as described above.

Merging compass and rotation data for activity recognition is also quite common([11,6,2,1]). Lester et. al. use a combination of accelerometer, sound, gyro and magnetic field sensors embedded in a mobile phone size device [11]. Junker et. al. describe how to combine inertial sensors to spot sporadic user gestures in a data stream [6].

Complementary to the work presented in this paper, Minnen et. al. describe a way to detect unsupervised "Motifs" in sensor data [13]. This might be used to detect "interesting" disturbances in the magnetic field automatically without training.

In previous work, we have shown how to estimate the angular velocity form magnetic field sensor data only [8].

While all of the above research deals with magnetic sensing in activity recognition, our work is fundamentally different as it aims to use magnetic disturbances rather then the pure magnetic signal caused by user motion. As such it is complementary to previous work. Features derived using our method can thus be used and can be applied in situations where magnetic field signals were previously of little use.

2 Approach

To isolate the magnetic disturbances we compare the angular velocity given by the gyroscopes with the value computed from the magnetic field data. To this end we use the approximation of angular velocity from 3D magnetic sensor data described in [8]. It is based on the fact that the three field strength components that a 3D magnetic field sensor outputs represent a vector $\boldsymbol{B}$ that is tangential to the magnetic field line at sensor location. This vector is given in the local coordinate system of the sensor with the vector length representing the scalar field strength (norm of the field vector at the location). Thus, if we orient the sensor in such a way that one sensor axis (e.g. x-axis) points in the direction of the magnetic field (is tangential to the field line) then the sensor reading will be $\boldsymbol{B(t)} = (b, 0, 0)$ with $b = \|\boldsymbol{B(t)}\|$ being the magnetic field strength at the location. If we orient the sensor with the (x, y) plane being tangential to the field line than the output will be $(b * cos(\varphi), b * (sin(\varphi), 0)$. Generalizing to arbitrary orientations of the sensor with respect to the field line we have:[1] $B_i(t) = \|\boldsymbol{B(t)}\| \cdot cos(\varphi_i)$ The angle $\varphi_i(\boldsymbol{B}(t))$ between the i-th axis and the magnetic field strength vector $\boldsymbol{B}(t)$ measured at time t is then given by $\varphi_i(\boldsymbol{B}(t)) = \arccos \frac{B_i(t)}{\|\boldsymbol{B}(t)\|}$ where $B_i(t)$ is the i-th component of $\boldsymbol{B}(t)$. Angular velocity then equals the first derivative of the angle.

2.1 Analyzing the Distortion

Obviously, the presence of metallic objects and artificial magnetic fields (e.g. caused by electric appliances) has significant influence on the quality of the approximation. Figure 1 shows an example comparing the magnitude of gyro data and the magnitude of estimated angular velocity based on magnetic data. The third graph in 1 highlights the principle we exploit in this work: when angular velocity measured by the gyro and angular velocity estimated by use of the magnetic sensor are equal, their difference is zero. When they diverge, however, the difference takes on non zero values, signifying a disturbance in the magnetic field. In previous work we have tried to filter out this influence and focus on applications where it is negligible. In this paper we do the opposite: we explicitly try to analyze the distortions and use them as a source of information. We focus on scenarios where the user wears a combination of a magnetic field sensor and a gyroscope while performing activities in the proximity of metallic objects and other sources of magnetic disturbance.

As described in [8], the distortion depends on many factors and is not easily described in exact terms. However, in most cases the nature of the source of magnetic distortion is a more significant influence then fine variation in the motions associated with the activity. Thus, in activity recognition tasks where the classes are associated with different objects or devices (=different sources of magnetic field disturbance), the use of magnetic disturbance as a feature can increase the robustness of the system in comparison to purely motion analysis based approaches. This is because the sources of disturbance often remain largely unchanged, whereas the specific motion signals associated with the activities can significantly vary. Such variations are particularly relevant when looking at user independent recognition, but can also be non negligible for a single user. In

[1] This assumes the magnetic sensor delivers data in a standard basis coordinate system; it is, however, possible to perform a basis transformation if that is not the case for a given device.

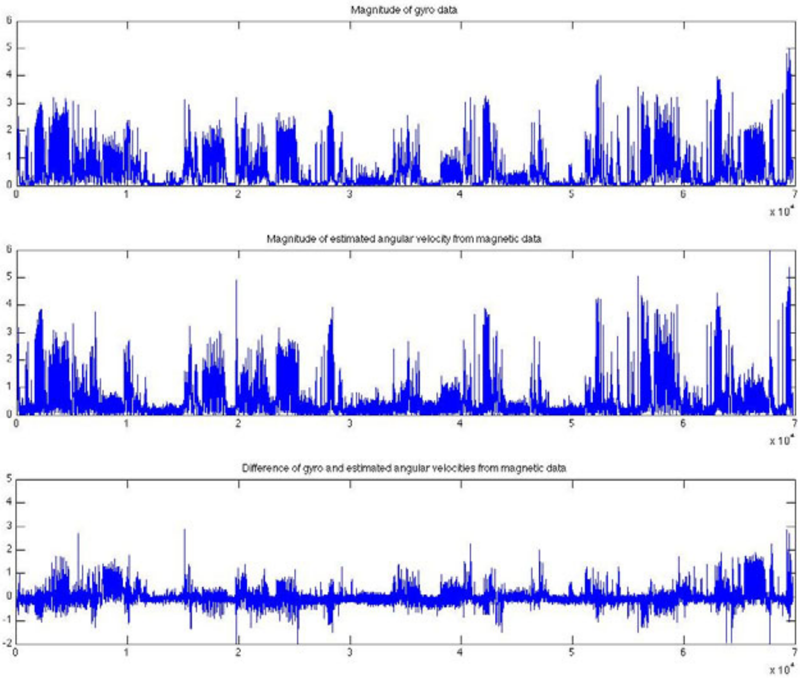


Fig. 1. Magnitudes of (estimated) angular velocities from gyro and magnetic data and difference thereof

addition, as was shown in [9], recognition rates drop dramatically if training data is recorded from a different sensor location than testing data (e.g. because of the sensor being loosely attached and slipping). In all of the above cases using features gained from difference of estimated and real angular velocity, recognition rates can be improved.

2.2 Problems and Limits

The main problem we encountered was that the approximation of angular velocities using magnetic data on individual axes was not good enough to yield useful information when subtracted from the three individual gyroscope signals. As [8] states, the estimate depends on the basis vectors chosen for the projections of the magnetic field strength vector. Calculating the magnitude over all three estimated axes skirts this difficulty, as the magnitude is not dependent on the exact distribution of the velocities between the three axes. Unfortunately, it also destroys potentially useful direction specific information. At the moment, this boils down to using the total amount of distortion as a feature instead of the distortion on each axis. In the future, we will try to further refine the approximation technique to allow for individual axis estimates good enough to use in the way described in this paper.

3 Evaluation and Results

3.1 DataSet

We evaluated our approach on part of the gym data set previously published in [9]. This data set consists of 8 gym exercises (lat machine, butterfly, shoulder press, upper back, arm extension, arm curl, pull down, chest press). The exercises are performed by two subjects with 10 to 15 repetitions on each machine (total of around 200 instances), taking about 5 minutes for each set.

The data set is recorded with XSENSE MTX inertial sensor modules containing both a gyroscope and a magnetic field sensor, as well as an accelerometer. For both test subjects, five modules have been used to record the entire training session; we exploit this to demonstrate our approach for displaced sensors (using data of one sensor for training and then testing on the other 4).

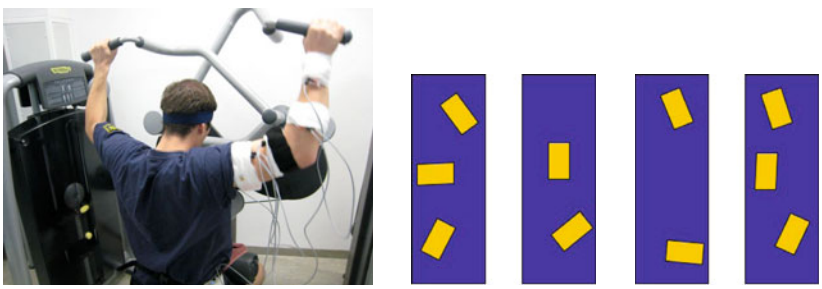


Fig. 2. Sensor placement for gym dataset for subject 1 and 2; blue rectangles represent the rigid bodies (upper / lower arm), yellow boxes the placement of the sensors

The sensors were split into two groups of three and two sensors respectively. Figure 2 shows the positions of the modules).

It is noteworthy that during the recording of subject two, one of the sensors attached to the upper arm shifted position, resulting in a new position between both upper and lower arm sensor groups.

3.2 Evaluation

We calculated features in a sliding window one second long with 0.5 seconds overlap. The features used were *mean, var, amplitude, rms, centre of mass and fluctuation of spectrum and the first five cepstrum coefficients as defined by ift(log(abs(fft(signal)))).*

All features were calculated on the magnitude of acceleration, gyroscope and magnetic 3D data vectors. As stated above, the magnetic field based approximation of angular velocity is not yet sophisticated enough to work on individual axes. Furthermore, using the individual axes for classification performs significantly better only as long as there is no sensor displacement (for reference, we did include the 3D data in the case of non displaced sensors). Once sensor displacement happens, the opposite is actually the case. As translations and rotations change the distribution of force to individual axes far more than the overall amount of force measured by the sensor, using features based on magnitudes of sensor signals is far more robust under displacement. E.g., using gyro and acceleration data, recognition rates are as good as 96.8% using 3 axes but drop down to 24% when the sensor is displaced. On the other hand, resorting to the magnitude features only yields 69.2% without displacement, but only drops to 56.5% in a different sensor location.

A note on computational complexity: All the calculations required for our approach are quite simple; both the estimation of angular velocity from magnetic data and the features used do not require extensive computational power; while we used an already available data set for this work and performed our analysis offline on a workstation running Matlab, it is computationally feasible to implement it in an online setting, e.g. on a mobile device.

For the classification itself, we tested different frame-by-frame classifiers (C4.5 decision tree, kNN, NaiveBayes). Results from these classifiers were within 1% of one another, so without loss of generality, the numbers we present are derived from the NaiveBayes classifier. Training data consisted of half a set (obtained by a 50% percentage split) of full data for non displaced sensors and of a full set of any one sensor in the case of displaced sensors. Testing data likewise was either the other half in the single sensor case or the full data of the remaining four sensors in the displaced case. In all evaluations safe the last one, both training and testing was done on a per subject basis. The results were then averaged. While user independence is a desirable quality in activity recognition, it is also difficult to achieve; in the gym setting, e.g., different users perform motions slightly differently. We address this topic in the last part of the evaluation.

In order to determine whether the difference of gyro signal and magnetic estimated angular velocities contained useful information, in a first step we compared two sets of features. The first set contained only features derived from the individual gyro and magnetic data vectors; the second set was expanded by adding features calculated from the difference of gyro and estimated magnetic angular velocities.

As outlined above, it is not a good idea to use raw magnetic data for activity recognition. In our data set, rotating the sensors (or, just as bad) the gym machines would result in completely different magnetic sensor readings. We therefore performed two more evaluation steps.

In a second evaluation, we compared two sets of data; one set only containing features derived from the gyro readings, with the second set once again including the gyro and magnetic velocity difference features.

In a third step, we added the acceleration data, comparing one set of features calculated on magnitudes of acceleration and gyro data to a set containing those and additionally the information derived from differences in gyro and magnetic angular velocity.

As a final evaluation step, we have tested how well the additional features derived from the angular velocity comparison perform when training data and test data do not originate from the same person. To that end, we used sensor data from subject 1 for training and data from subject 2 for testing; afterwards, we switched persons and repeated the process, averaging over the results.

3.3 Results

Carrying out the evaluation strategy outlined above, we achieved the results shown in Figure 3. The details of each evaluation step are described in the corresponding subsection below.

Comparison: gyro + magnetic data to same + difference in angular velocities. In the case of no sensor displacement, classifying the gym data set with features calculated from gyroscope and magnetic sensor data yields average recognition rates of 90.0% using individual axis data and 74.6% using magnitude of axes data. Extending the feature set by features derived from the difference of gyro angular velocities and estimated magnetic angular velocities can boost these rates to 93.0% in the 3D case and 83.9% in the magnitude case; the gains of 3.0% and 9.3% respectively prove that the features stemming from angular velocity comparison of gyro and magnetic data do contain additional information

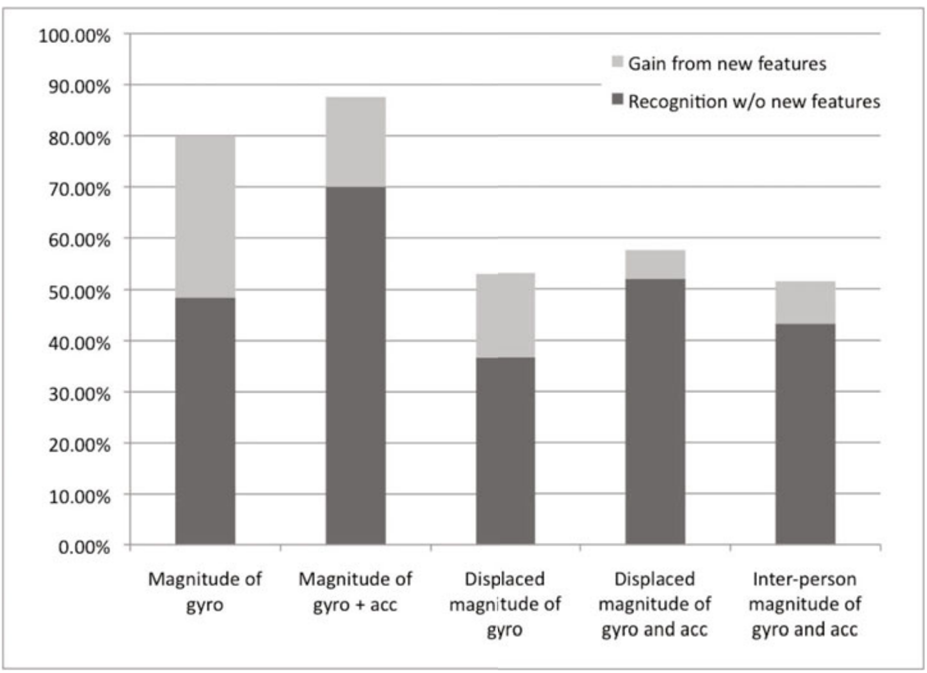


Fig. 3. Overview of base recognition rates and gains for various cases

that can be used to further improve recognition rates. The lower gain in the 3D case is partly due to the much higher base recognition rates, which limit the room for improvement.

In the case of displaced sensors, adding the difference features boosts average recognition rates from 52.4% to 59.7%, resulting in an average gain of 7.3%. Individual gains between each pair of sensors were uniformly distributed even between the different sensor groups of upper and lower arm; this implies that - at the very least - the additional features are less sensor location dependent than e.g. raw gyro or magnetic data. Looking at the individual axis data under displacement underscores the arguments for using the magnitude of sensor data made above, with recognition rates dropping to about 30%. There is, however, still an average gain of 3.7% when adding the additional features.

In conclusion, the results of the first evaluation step show that both in cases with and without sensor displacement, using the magnetic/gyro difference features improves recognition rates. Table 1 summarizes these results.

Table 1. Comparison of recognition rates w/o and with angular velocity comparison features for gyroscope and magnetic data

Modality	Recognition Rate w/o ang. vel. comp. features	Recognition Rate w. ang. vel. comp. features	Gain
3D, undisplaced	90.0%	93.0%	3.0%
magnitude, undisplaced	74.6%	83.9%	9.3%
magnitude, displaced	52.5%	59.7%	7.3%

Comparison: gyro data to gyro + difference in angular velocities. Classifying the gym dataset with features derived solely from gyroscope sensor data yields an average recognition rate of 80.1% in the 3D case and 48.5% in the magnitude case as long as the sensor has not been displaced. In both cases, adding the features resulting from our difference of angular velocities approach net improvements: average recognition rates rise to 90.4% in the 3D case, a gain of 10.3%. Adding the additional features to features calculated from the magnitude gyro signal vector, recognition can be boosted to 79.9%, showing a tremendous gain of 31.4%. This suggests that adding the new features manages to recoup at least part of the information lost by switching from 3D data to magnitude of data. In the case of displaced sensors, recognition rates using only gyroscope data drop by 11.8% to 36.7% using the features calculated on the magnitude of the signal. Once again, adding the information resulting form our comparison of gyro angular velocity and the estimated magnetic one yields significant improvements, raising average recognition rates amongst the four different groups of displaced sensors to 53.1%, a gain of 16.4%.

Table 2 summarizes these results.

Comparison: gyro + acceleration data to same + difference in angular velocities. In the third section of the evaluation, we combined gyroscope and acceleration data for the feature set; these modalities provide a realistic

Table 2. Comparison of recognition rates w/o and with angular velocity comparison features for gyro data

Modality	Recognition Rate w/o ang. vel. comp. features	Recognition Rate w. ang. vel. comp. features	Gain
3D, not displaced	80.1%	90.4%	10.3%
magnitude, not displaced	48.5%	79.9%	31.4%
magnitude, displaced	36.7%	53.1%	16.4%

scenario for activity recognition. In the case of a single, non displaced sensor, recognition rates are very good, especially when using the 3D signal for feature calculation. Averaged over the 5 different sensors and both test persons yields a recognition of 95.7%. This is very close to perfect already and leaves little room for improvement. Even so, however, adding the features derived from the angular velocity comparison raises accuracy slightly to 96.8%, a gain of 1.1%. Using the magnitude of acceleration and gyroscope as basis for feature calculation expectedly drops recognition rates to 70.0% when there is no sensor displacement. Adding the information derived from our approach manages to increase accuracy significantly again to 87.6%, a gain of 17.6%.

Finally, in the case of displaced sensors, recognition rates drop down to an average of 52.0% over all 4 groups of sensors, using just acceleration and gyroscope data. Adding in the gyroscope - magnetic comparison features manages to improve accuracy by 5.6% to 57.6%.

Overall, as in the other cases presented in our evaluation, there is measurable information contained in the features derived from comparing actual and approximated angular velocity. Table 3 summarizes these results.

Table 3. Comparison of recognition rates w/o and with angular velocity comparison features for combined gyro and acceleration data

Modality	Recognition Rate w/o ang. vel. comp. features	Recognition Rate w. ang. vel. comp. features	Gain
3D, not displaced	95.7%	96.8%	1.1%
magnitude, not displaced	70.0%	87.6%	17.6%
magnitude, displaced	52.0%	57.6%	5.6%

Comparison: gyro + acceleration data with and without angular velocity difference between subjects one and two. In activity recognition, an added difficulty may present itself: when classifiers trained on data obtained from one person are used to classify activities performed by a different person, recognition accuracy usually drops, sometimes severely. Thus, while user independence is of course a desirable quality of any recognition system, it is usually hard to achieve. E.g., different users may perform activities slightly differently and sensor placement may diverge slightly. In theory, magnetic distortions should be quite person independent, as they depend mostly on the environment. To test this hypothesis, we trained with acceleration and gyroscope data from subject one and tested on data from subject two and vice versa. We obtained these results: Training with upper arm data without our magnetic distortion features resulted in a recognition rate of 40.05%, while lower arm data performed slightly better at 46.51%. On average, recognition accuracy was 43.28%. Adding the features calculated from angular velocity comparison resulted in noticeable gains; upper arm accuracy increased to 46.86%, a gain of 6.81%; lower arm recognition was boosted to 56.08%, an increase of 9.57%; finally, on average, recognition rose to 51.47%, an improvement of 8.19%. Table 4 summarizes these results.

Table 4. Comparison of recognition rates w/o and with angular velocity comparison features between different subjects

Modality	Recognition Rate w/o ang. vel. comp. features	Recognition Rate w. ang. vel. comp. features	Gain
upper arm	40.05%	46.86%	6.81%
lower arm	46.51%	56.08%	9.57%
both	43.28%	51.47%	8.19%

Since sensor placement on subject one was different from sensor placement on subject two, it is not possible to divide total gains from the additional features into gains from displacement resistance and gains from person independence. Overall, while recognition expectedly performs worse than in the case of same subject training and testing (51.47% compared to 57.6%), the gain from the magnetic distortion features is actually 46% higher (8.19% compared to 5.6%). This indicates that our initial supposition of the added value for achieving a measure of subject independence is correct.

4 Conclusion

The results presented in the previous section are a strong indication that magnetic field disturbances computed from the difference between gyro signals and angular velocity derived from magnetic field vector changes are useful for wearable activity recognition. The inclusion of the corresponding features has lead to an improvement of performance in all experiments. This included cases of shifted sensors and person independent recognition.

Despite the promising results, our evaluation, however, should only be considered a starting point for further work. The technique presented here should be tested on a wider range of datasets. It would also be interesting to apply it to data gathered from mobile devices like the iPhone 4. Another aspect deserving of more research is further refining the estimation approach itself in order to be able to apply it to individual axes with sufficient precision. It would then be possible to characterize distortions along three individual dimensions instead of just their magnitude, possibly making the technique even more valuable. In that context, it might also be worth exploring how much sensor displacement is necessary before the gains from the greater invariance to displacement outweigh the accuracy penalties in case of little displacement when switching from 3D signal data to magnitude of signal data.

References

1. Amft, O., Bannach, D., Pirkl, G., Kreil, M., Lukowicz, P.: Towards wearable sensing-based assessment of fluid intake. In: 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops), pp. 298–303 (2010)
2. Benbasat, A., Paradiso, J.: An inertial measurement framework for gesture recognition and applications. Gesture and Sign Language in Human- (January 2002)
3. Fels, S., Lyons, M.: Creating new interfaces for musical expression: introduction to nime. In: SIGGRAPH 2009: SIGGRAPH 2009 Courses (August 2009)
4. Förster, K., Roggen, D., Tröster, G.: Unsupervised classifier self-calibration through repeated context occurences: is there robustness against sensor displacement to gain? In: ISWC 2009 (2009)
5. Hook, J., Taylor, S., Butler, A., Villar, N., Izadi, S.: A reconfigurable ferromagnetic input device. In: Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology, UIST 2009 (October 2009)
6. Junker, H., Amft, O., Lukowicz, P., Tröster, G.: Gesture spotting with body-worn inertial sensors to detect user activities. Pattern Recognition (January 2008)
7. Kern, N., Schiele, B., Schmidt, A.: Recognizing context for annotating a live life recording. Personal and Ubiquitous Computing (January 2007)
8. Kunze, K., Bahle, G., Lukowicz, P., Partridge, K.: Can magnetic field sensors replace gyroscopes in wearable sensing applications? To be published in ISWC (October 2010)
9. Kunze, K., Lukowicz, P.: Dealing with sensor displacement in motion-based onbody activity recognition systems. In: Proceedings of the 10th International Conference on Ubiquitous Computing, UbiComp 2008 (September 2008)
10. Ladetto, Q., Gabaglio, V.: Combining gyroscopes, magnetic compass and gps for pedestrian navigation. Systems in Geodesy (January 2001)
11. Lester, J., Choudhury, T., Borriello, G.: A practical approach to recognizing physical activities. In: Proceedings of Pervasive (January 2006)
12. Levi, R., Marshall, R.: Gyro aided magnetic compass. US Patent App. 10/618,549 (January 2003)
13. Minnen, D., Starner, T., Essa, I., Isbell, C.: Improving activity discovery with automatic neighborhood estimation. In: International Joint Conference on ... (January 2007)
14. Raya, R., Ceres, R., Roa, J., Rocon, E.: Assessment of the involuntary motion of children with motor impairments to improve the accessibility of an inertial interface. In: Proceedings of the 9th International Conference on Interaction Design and Children, IDC 2010 (June 2010)
15. Stiefmeier, T., Roggen, D., Troster, G., Ogris, G.: Wearable activity tracking in car manufacturing. IEEE Pervasive ... (January 2008)
16. Young, A., Ling, M., Arvind, D.: Distributed estimation of linear acceleration for improved accuracy in wireless inertial motion capture. In: Proceedings of the 9th ACM/IEEE International Conference on Information Processing in Sensor Networks, IPSN 2010 (April 2010)
17. Zappi, P., Stiefmeier, T., Farella, E., Roggen, D.: Activity recognition from onbody sensors by classifier fusion: sensor scalability and robustness. In: Proceedings of ... (January 2007)

Feature Weighting for CPM-Based Context Validation

Nermin Brgulja, Rico Kusber, and Klaus David

Chair for Communication Technology, University of Kassel, Germany
{nbrgulja,rkusber,david}@comtec.eecs.uni-kassel.de

Abstract. Context Pattern Method (CPM) is a statistical method that is used to quantify the validity of contextual information based on dependent contexts using previous knowledge about the system. The method exploits the interdependencies in a context aware system among entities and the environment in which they reside in order to calculate the Probability of Correctness (PoC) for a context under investigation. PoC expresses the level of confidence, that the contextual information sensed, are in fact correct or not. Obviously, each of the dependent contexts has a different importance to the context that is under investigation. Therefore its influence to the PoC measure needs to be weighted accordingly. In this paper we discuss the concept of feature weighting and show how feature selection algorithms can be applied for this purpose. We apply chi², relief-f and mutual information, algorithms to the CPM method in order to weight the influence of the individual dependent contexts to the overall PoC measure and evaluate the method's performance.

Keywords: feature weighting; context pattern method; CPM; relief-f; mutual information; chi square.

1 Context Validation Using CPM Method

One of the main challenges in pervasive and context-aware computing is the ability to identify and detect context information uncertainty that emerge in a context aware system as the result of inherent inaccuracy and unreliability of many types of context providers and sensors which by their nature are imperfect and prone to errors and failures [3] [4] [5]. In our previous work [1] the Context Pattern Method (CPM) has been developed that calculates the Probability of Correctness (PoC) of contexts and as such provides a mechanism to assess the risk of using the contextual information. CPM method exploits the interdependencies that exist in a context aware system among entities and the physical environment in which they reside in order to affirm or contradict the context under scrutiny. The CPM method defines PoC as:

$$PoC(ce_i) = \frac{1}{(n+m)}\left(\sum_{j=1}^{n} p(ce_i \mid ce_j) + \sum_{k=1}^{m} QoCp(ce_i)_k\right);\ ce_j \in DC_i \tag{1}$$

Where **ce** is the abbreviation for context element. It is the basic context information entity, which provides evidence about what has been observed, when and where it has

P. Lukowitz, G. Kortuem, and K. Kunze (Eds.): EuroSSC 2010, LNCS 6446, pp. 82–93, 2010.

been observed and the observation value itself. A context element can be both, a sensor reading as well as a high level context inferred out of multiple sensor readings. Every ce is characterized by a set of Quality of Context (QoC) parameters. The **PoC(ce_i)** is the probability of correctness of the context element that is under investigation ce_i. It is a value between 0 and 1 where 0 means complete disapproval and 1 complete approval of the value provided by the context element that is under investigation ce_i. The **p($ce_i|ce_j$)** is the conditional probability between the context element under investigation ce_i and context element it depends on ce_j. The value is calculated based on previous observations about the system. **DC_i** is seen as the totality of dependent context elements ce_j which a context element under investigation ce_i depends on. **QoCp(ce_i)** is the QoC parameter of the context element under investigation ce_i. **n** is the number context elements which ce_i depends on (n = DC_i size) and **m** is the number of used QoC parameters.

The CPM method has been purposely developed to calculate the context information's PoC which is a QoC parameter that expresses the probability that the provided context information is correct. It is a value between 0 and 1 where 1 describes the case where the system is fully confident that the evaluated contextual information is correct and 0 denotes the opposite case where the system is fully confident that the evaluated contextual information is wrong. The closer the PoC value is to one of these two extremes, the higher is the probability that the provided context information is valid (when PoC→1) and invalid (when PoC→0). The PoC value can be used as the basis for context validation purposes. As shown in [2], the context information is classified valid/invalid by comparing its PoC value against a predefined threshold.

According to Equation (1), the CPM method considers all dependent contexts $ce_j \in DC_i$ being equally important to the context under investigation ce_i. This equal weight bias handicaps CPM method since it allows less relevant dependent contexts $ce_j \in DC_i$ to have as much effect to PoC calculation as others. Thus, CPM can perform poorly when many irrelevant dependent contexts $ce_j \in DC_i$ are used for calculating the PoC of the context element under investigation ce_i. In this paper we enhance the CPM method by applying weights to the individual conditional probabilities and the QoC parameters. The weighted CPM method's performance and characteristics are summarized using experimental evaluation.

The paper is organized as follows; Section 2 discusses the concept of feature weighting using feature selection algorithms. Section 3 proposes a solution for the CPM method. Section 4 describes the experiments and presents the evaluation results. Section 5 concludes the paper also outlining ongoing and future work.

2 Feature Weighting

In machine learning, feature weighting is used with the purpose of reflecting how much a particular feature reveals about class membership of instances. The weights of features are determined from their distribution across training classes. In the context of context validation using the CPM method, this procedure can be formalized as follows. As presented in Equation (1), the context element under investigation ce_i is validated against each dependent context element $ce_i \in DC_i$ and its QoC parameters

QoCp(ce_i). For feature weighting purposes, a function w(ce_j,ce_i) is required that computes the relevance score of a dependent context element ce_j to the context element under investigation ce_i. This score can be used directly as its weight within the CPM method. The same applies to the individual QoC parameters QoCp(ce_i). However, in this paper we assume that the weights for the individual QoC parameters $wq(ce_i)_k$ are specified by the user.

A key decision in the dependent context weighting procedure is to choose a function computing w(ce_j,ce_i). Such functions typically try to capture the intuition that the best dependent context elements ce_j are the ones that best reflect the context element under investigation ce_i. They determine w(ce_j,ce_i) from the correlation between ce_j and ce_i, attributing greater weights to those ce_j that most correlate with ce_i. In this study we introduce three such functions: mutual information, relief-f and chi².

2.1 Mutual Information

Mutual Information (MI) is a theoretic measure of association between two variables that measures the mutual dependence of the two variables. The MI of two variables is the reduction in uncertainty of one variable's value given knowledge of the other's value. From that point, MI between ce_j and ce_i measures how much information presence of ce_j contains about ce_i.

$$MI(ce_j;ce_i) = \sum_{ce_i,cei} p(ce_j \cap ce_i) \cdot \log\left(\frac{p(ce_j \cap ce_i)}{p(ce_j)p(ce_i)}\right) \tag{2}$$

According to Equation (2), in case ce_i is totally dependent on ce_j, the $p(ce_j \cap ce_i)$ is equal to $p(ce_j)$. As the consequence MI(ce_j,ce_i) = H(ce_j) which is the Shannon entropy, or amount of information contained in ce_j. On the opposite, if ce_i is totally independent from ce_j, the $p(ce_j \cap ce_i)$ would be equal $p(ce_j)p(ce_i)$ and MI(ce_j;ce_i) = 0.

2.2 Relief-F Algorithm

In [6] a statistical feature selection algorithm called relief has been described, that uses instance based learning to assign relevance weight to each feature. Relief algorithm is a filter method based on the feature weighting approach that estimates features according to their performance in distinguishing near instances. Relief searches the two nearest neighbours for each instance: one from the same class (nearest hit) and one from the different class (nearest miss).

There are many variations of the relief algorithms, whereas only the relief-f offers weight update schemes in a multiclass problem. Relief-f was tested as the feature weighting method in lazy learning [7] and was found to be very useful. The outcome of the relief-f algorithm is a list of weights assigned to the individual features according to its relevance. From the CPM method point of view, the relief-f will calculate a weight for every dependent context element $ce_j \in DC_i$ according to its relevance to the context element under investigation ce_i. Weights are values between 0 and 1 where 1 describes the case where ce_j is very relevant to ce_i and therefore important for validating the ce_i and 0 the case where ce_j is irrelevant to ce_i and therefore completely unimportant for ce_i-s validation.

2.3 Chi² (χ^2) Algorithm

The χ^2 statistic measures the lack of independence between two features. The χ^2 coefficient is a measure for association between two features. From the CPM method point of view, χ^2 denotes the strength of the association between ce_j and ce_i.

$$\chi^2 = \sum_{i=1}^{r}\sum_{j=1}^{c}\frac{(O_{ij} - E_{ij})^2}{E_{ij}} \tag{3}$$

Where r is the number of rows and c is the number of columns of the contingency table. O_{ij} is the observed frequency and E_{ij} is the expected (theoretical) frequency. Both values are calculated based on the values from the contingency table. The resulting χ^2 is equal to 0 if ce_j and ce_i are independent. The χ^2 can take any value larger than 0 based on the dependency strength between ce_j and ce_i. The χ^2 test says that there is a significant relationship between variables but it does not say how significant and important this is. Cramer's V can be used as post-test to determine strengths of association after χ^2 has determined the significance.

$$V = \sqrt{\frac{\chi^2}{n \cdot (\min[r, c] - 1)}} \tag{4}$$

Cramer's V varies between 0 and 1. Close to 0 it shows little or no association between ce_j and ce_i whereas when close to 1 it indicates a strong association between ce_j and ce_i.

3 Feature Weighting for the CPM Method

Integrating the concept of feature weighting into the CPM method requires incorporating the weights of the individual features. From the Equation (1) we can state that the weighting concept is required for two types of parameters: Firstly, the conditional probabilities $p(ce_i|ce_j)$ has to be weighted using the $w(ce_j,ce_i)$ weights. The $p(ce_i|ce_j)$ probabilities are calculated based on the reference context patterns which reflect the previous observation about the system (the system knowledge). The methods described in Section 2 can be used for calculating the $w(ce_i,ce_j)$ weights based on the same reference context patterns that are used for calculating the individual $p(ce_i|ce_j)$. Generally, the $w(ce_j,ce_i)$ represents the importance of the dependent context element ce_j for validation of the context element under investigation ce_i. Secondly, the QoC parameters $QoCp(ce_i)$ needs to be weighted, which describe the quality of the contextual information provided by the context under investigation ce_i. We specify $QoCp(ce_i)$ weights as $wq(ce_i)_k$ which represents the weight of the k-th $QoCp(ce_i)$ parameter. It is assumed that the $wq(ce_i)$ weights are specified in the range 0..1. The closer the $wq(ce_i)$ to one of the two extremes, the more important a QoC parameter is for the validation of ce_i (case $wq(ce_i)\rightarrow 1$) and respectively less important for the ce_i-s validation (case $wq(ce_i)\rightarrow 0$). The new $PoC(ce_i)$ using feature weighting is defined as:

$$PoC(ce_i) = \frac{\left(\sum_{j=1}^{n} w(ce_j, ce_i) p(ce_i \mid ce_j) + \sum_{k=1}^{m} wq(ce_i)_k QoCp(ce_i)_k \right)}{\sum_{j=1}^{n} w(ce_j, ce_i) + \sum_{k=1}^{m} wq(ce_i)_k} \quad (5)$$

$w(ce_j, ce_i)$ is the weight for the dependent context element ce_j which reflects the importance of the ce_j for validation of the context element under investigation ce_i. The value is calculated using one of the methods described in Section 2 based on the same training data set (reference contexts) used for calculating the conditional probabilities $p(ce_i|ce_j)$ in the CPM method.

$wq(ce_i)_k$ is the weight for the k-th QoC parameter of the context element under investigation $QoCp(ce_i)_k$. It is a value provided by the user which represents the QoC parameter's importance for the $PoC(ce_i)$ calculation.

For the Equation (5), we have to consider that two different weight calculation types using different methodologies are applied: one for weighting the individual dependent contexts $w(ce_j,ce_i)$ and another one for weighting the individual QoC parameters $wq(ce_i)_k$. In order to make the Equation (5) consistent it is required to normalize the weights $w(ce_j,ce_i)$ and $wq(ce_i)$ into the same weight scale. In this research, the QoC parameters weighting as well as the feature weighting methods described in Section 2 are using the 0..1 weight scale.

4 Feature Weighting for the CPM Method

4.1 Smart Home Scenario

The smart home scenario envisions a person living in a smart environment where devices and services autonomously adapt based on changes in the environment and in the context of the user. It is an adapted version of the scenario developed in [1]. In the scenario, the user location is sensed using the mobile phone's WiFi interface and is validated against a set of dependent contexts elements $ce_j \in DC_i$ which, as shown in Figure 2, comprise: loudness of the user's surrounding, activity detection on the user's Personal Computer (PC), motion detection in the individual rooms, temperature of the user's surrounding and the list of discovered Bluetooth devices. The floor plan for the selected scenario is presented in the Figure 1 and can be easily extended.

The smart home scenario has been simulated using the Siafu context simulator (http://siafusimulator.sourceforge.net). The user behavior is modeled based upon the typical behavior of a person living in such an environment. The model contains several activities like cooking, watching TV, relaxing, reading a book, working on the PC, sleeping etc. The user model contains a high degree of freedom, which means that there is no strict user behavior in the model. The model roughly specifies the user's daily live activities, but the time and the duration of the individual activity varies. The user movement itself, and in particular, the path along which a user will move around is modeled and entirely handled by the Siafu simulator. The estimated user location is sensed through the user's mobile phone, using its WiFi interface. The weighted CPM method which is specified in the Section 3 through the Equation (5) is applied to calculate the probability of correctness for the sensed user location, which suffers from the inaccuracies and external influences to the WiFi radio channel. The feature weighting methods described in Section 2 are used for calculating the individual weights $w(ce_i,ce_j)$. For that purpose we have used the WEKA framework (www.cs.waikato.ac.nz/ml/weka/), which provides the implementation for the three feature selection algorithms described in Section 2.

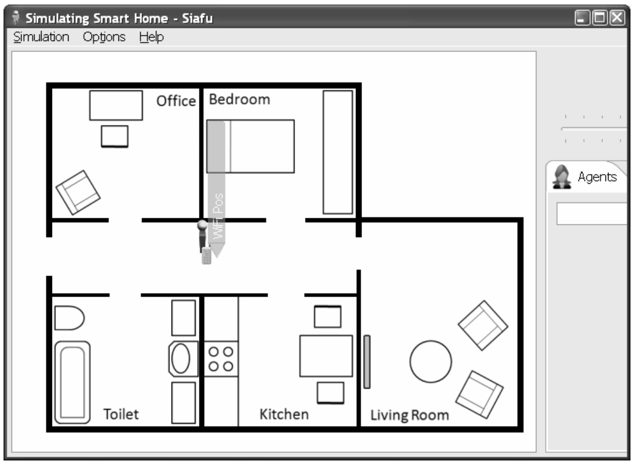


Fig. 1. Smart Home floor plan

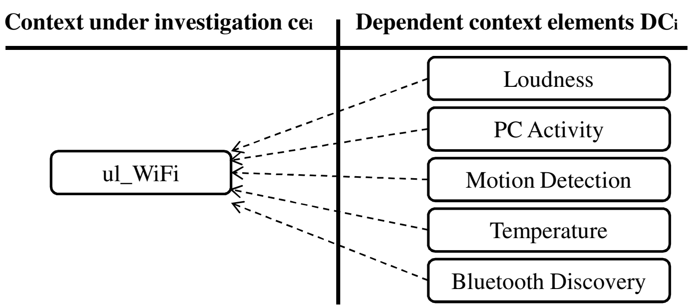


Fig. 2. User location and the list of dependent context elements used for its validation

In the scenario we distinguish between two locations:

- the actual user's location (ul_actual) which is the real location of the user in the apartment obtained via the Siafu simulator and
- the user location derived using the mobile phone's WiFi interface (ul_WiFi)

The ul_WiFi is calculated using the WiFi grid model indoor positioning method described in [9]. For failure injection purposes, the WiFi signal strength is modified. The signal strength modification is modeled based on real world problems, where the instability of the signal strength is introduced due to changing environmental dynamics caused by for example open/closed doors and the availability of other devices using the same frequency. The CPM methods, using the feature weighting types described in Section 2, are applied in order to calculate the probability of correctness for the sensed user locations PoC(ul_WiFi). Each time a change in user location ul_WiFi or in one of the dependent context elements $ce_j \in DC_i$ is sensed, the system calculates four PoC(ul_WiFi) values, one using non-weighted CPM method and three values using the feature weighting methodologies described in Section 2. The ul_WiFi is validated by comparing its PoC(ul_WiFi) value against a predefined threshold:

if: PoC(ul_WiFi) > 0.5 than: classify as VALID
if: PoC(ul_WiFi) < 0.5 than: classify as INVALID
if: PoC(ul_WiFi) = 0.5 than: classify as UNKNOWN

The special case PoC(ul_WiFi) = 0.5 denotes the situation where the method could not decide upon context's validity The outcome of the ul_WiFi validation process is compared to the ul_actual. In case ul_WiFi is classified as valid and ul_WiFi = ul_actual, the system will count a correct validation. The same applies for the case when ul_WiFi is classified as invalid and ul_WiFi ≠ ul_actual. All other cases are counted as wrong validation except the case when PoC(ul_WiFi) = 0.5 which is counted as unknown. The method's accuracy is calculated as the ratio between the number of correct validations and the number validation attempts.

4.2 Experiments and Evaluation Results

In the first set of experiments, the performance of the weighted CPM methods, using different feature weighting concepts described in Section 2, are compared against each other and against the non-weighted CPM method. The methods accuracies are evaluated with regard to the number of reference patterns (knowledge base size) and the number of dependent contexts $ce_j \in DC_i$ which is expressed by the size of DC_i. Context pattern is defined as a set of interdependent entities describing a contextual situation through their values [1]. For example, context elements: user location = "in the office", loudness = "quiet" PC activity = "active", motion = "office", temperature = "warm", Bluetooth = "PC" is a context pattern which collectively describe the contextual situation "user in the office". The reference patterns are context patterns observed in the past during the "system observation phase". They are used as the basis for the CPM method and the weights calculation during the "learning and validation phase". In the system observation phase, all dependent context elements $ce_j \in DC_i$ and the context element under investigation ce_i are continuously monitored. Each time a change in one of the context elements happens, a snapshot is made and the new context pattern is recorded. The reference context patterns represent the normal/ regular system state which means that the knowledge base data is error-free.

In the experiment, the used knowledge base size starts at 50 reference contexts patterns and increases in steps of 50 till 1000. For every new set of reference context patterns, the conditional probabilities $p(ce_i|ce_j)$ are learned, and the $w(ce_j,ce_i)$ weights for the individual dependent contexts $ce_j \in DC_i$ are calculated using chi^2, relief-f and the mutual information. Once the new references are applied, the user behavior is simulated for a full day (24h) and the validation outcomes are observed. The ul_WiFi position is validated based on the PoC(ul_WiFi) values which are provided: firstly, by the three variants of the weighted CPM method defined in Equation (5), using chi², relief-f and the mutual information for calculating $w(ce_j,ce_i)$ and secondly, by the non-weighted CPM method defined in Equation (1). Each method's accuracy is calculated as explained previously. The five dependent contexts $ce_j \in DC_i$, that are used for validating the ul_WiFi are listed in Figure 2 and are applied in the following order: loudness, PC activity, motion detection, temperature and the Bluetooth discovery. In the first run all five dependent contexts have been used for the PoC(ul_WiFi) calculation (DC_i size = 5) while in the second run the loudness is excluded (DC_i size = 4) and so on until only the Bluetooth discovery remained (DC_i size = 1). The evaluation results presented in Figures 3, 4, 5 and 6 shows the individual method accuracies with respect to knowledge base size and the number of available dependent contexts (DC_i size).

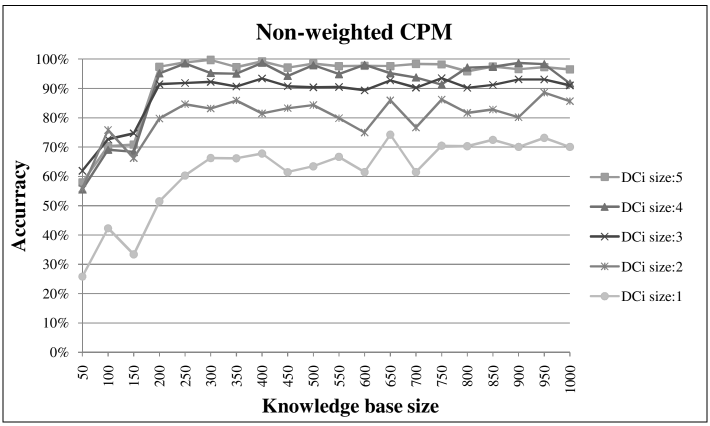


Fig. 3. Non-weighted CPM method's accuracy

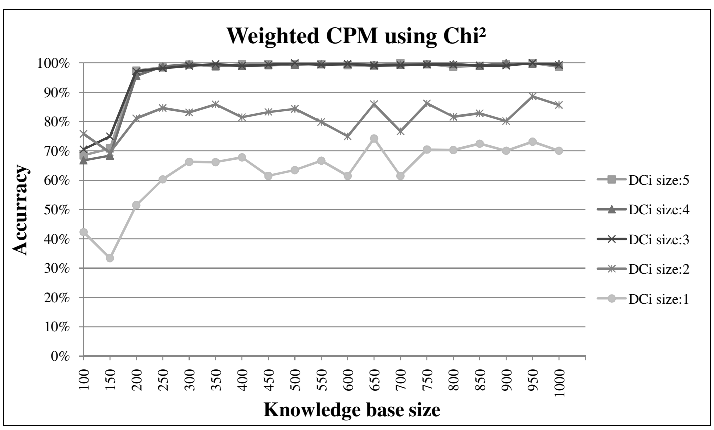


Fig. 4. Chi² weighted CPM method's accuracy

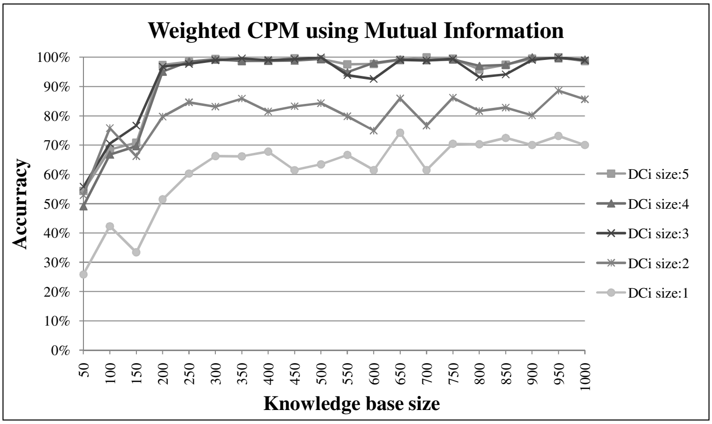


Fig. 5. MI weighted CPM method's accuracy

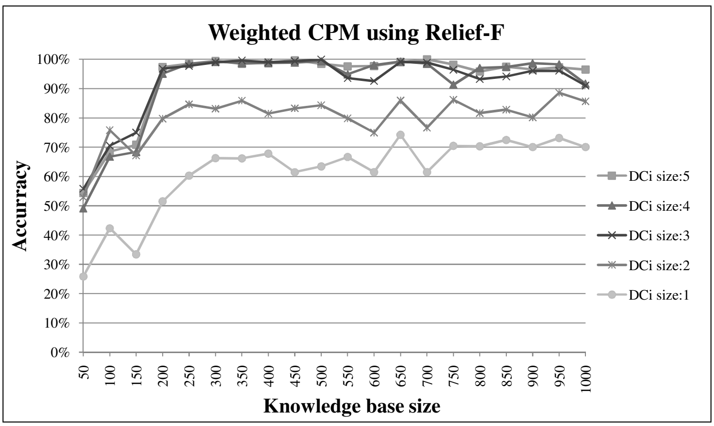


Fig. 6. Relief-f weighted CPM method's accuracy

Looking at the Figures 3, 4, 5 and 6, we can state that the CPM methods accuracy tends to increase with increasing DC_i size. Furthermore, the method requires a sufficient amount of reference context patterns (knowledge base size) in order to properly validate the context element under investigation ce_i. These evaluation results prove our previous conclusions about the CPM method characteristics [1] [2].

In principle, the weighted CPM methods perform better or at least equally as good as the non-weighted one, but the improvement in accuracy varies among different concepts of feature weighting. The strength of the improvement depends on various aspects: Firstly, on quality of the dependent context elements $ce_j \in DC_i$, that are used for validation of the context element under investigation ce_i. This directly depends on correlation between ce_i and the individual ce_j-s and expresses the amount of information about ce_i that can be derived from a ce_j. The stronger the variance in quality among $ce_j \in DC_i$, the higher is the importance for the weighted CPM method.

This is because the low quality $ce_j \in DC_i$ will corrupt the $PoC(ce_i)$ value of the non-weighted CPM method, since they are considered as equally important. Secondly, when there are less than 3 dependent context elements $ce_j \in DC_i$, the weighted CPM method has almost no advantage over the non-weighted one. From the Equation (5) one can see that in case when $ce_j \in DC_i$ contains only one context element, the $w(ce_j,ce_i)$ weights play no role for the calculation of $PoC(ce_i)$ when there are no QoC parameter used and a minor role when QoC parameters in use. Thirdly, when a low number of reference contexts are available (in our example knowledge base size < 150), the weighted CPM method performs worse than the non-weighted one. This behavior is due to the disability of the feature weighting to provide adequate weights for the dependent context elements when insufficient amount of reference data is available. This is reflected in lower accuracies of the weighted CPM method. However, as shown in [1], the CPM method requires sufficient amount of reference contexts in order to work properly.

From the Figures 4, 5 and 6 one can further see that different types of feature weighting used for calculating $w(ce_j,ce_i)$ result in different performances of the weighted CPM method. Our evaluation results show the advantage of weight calculation $w(ce_j,ce_i)$ using chi² over the relief-f and mutual information. The deviation of the mutual information weighted CPM method shown in Figure 5 from the chi² weighted one shown in Figure 4 is rather surprising since the major difference between chi² and mutual information is that chi² is a normalized value. However the reason for this is seen in the way the rare cases are handled. In case of the mutual information, rare cases will have higher score than the common ones, while in case of chi² the scores are calculated according to case frequency in the contingency table [8]. In our experiment, the relief-f weighted CPM method has performed worse than the chi² and mutual information weighted ones. From the Figures 5 and 6 one can see that continuous increase in knowledge base size does not necessarily means continuous increase in weighted CPM method accuracy. For every new set of reference context patterns, the weights are calculated new, which has a direct affect to the validation accuracy.

The first experiment has shown that the weighted CPM method performs better or equally as good as the non-weighted one. Does this also hold for cases when there are errors in dependent contexts? In the second set of experiments the performance of the weighted and non-weighted CPM methods are evaluated with regard to the number of errors in dependent context elements that are used for validating the ul_WiFi. It is assumed that the knowledge base data, which is used for learning the CPM method and calculating the weights $w(ce_i,ce_j)$, is correct. The errors are simulated in dependent contexts $ce_j \in DC_i$, which in real applications are provided by sensors that are by their nature imperfect and prone to errors and failures [3][4]. In the experiment setting, all 5 dependent contexts (DC_i size = 5) are used for validating ul_WiFi. The errors are modeled based on real world problems. The Bluetooth discovery error is modeled in the way that no Bluetooth devices are found in the entire apartment. Mobile phone's Bluetooth discovery delivers "no Bluetooth found" for all rooms. The PC activity error is modeled in the similar way so that the PC activity was set to "inactive" for all cases (also when user working on PC). The loudness and temperature errors are modeled such that the respective sensor will continuously measure the same loudness and temperature in all rooms. Motion detection error is modeled in the way that the

Table 1. Comparison of weighted and non-weighted CPM methods for different amount of failures in $ce_j \in DC_i$

Nr. Errors in $ce_j \in DC_i$	**Average validation accuracy using CPM method at DC_i size = 5 and knowledge base size > 200 context patterns**			
	Non-weighted CPM	***Chi² weighted CPM***	***Mutual Information weighted CPM***	***Relief-f weighed CPM***
0	97,7%	**99,3%**	98,1%	98,8%
1	93,6%	**96, 7%**	94,5%	95,4%
2	**81,2%**	68,5%	74,1%	77,6%
3	**65,1%**	59,7%	60,1%	62,4%
4	36,1%	32,0%	36,8%	34,4%
5	35,2%	31,5%	32,4%	33,6%

system will sense motion in the floor for all cases, also when the user is not in floor but in one of the rooms. The errors are applied accumulatively, starting at loudness and following the DC_i sequence depicted in Figure 2.

As shown in Table 1, the accuracy decreases for all methods with increasing number of errors in dependent contexts. In case when no errors in dependent context elements $ce_j \in DC_i$ are available, all weighted CPM methods perform better than the non-weighted one. This applies also for the case when only 1 erroneous dependent context element available. The table shows that further increase in the number of errors in dependent contexts cause a stronger decrease in accuracy for the weighted CPM methods than for the non-weighted one. This behavior can be explained by looking at the Equation 5. Due to the fact that in case of weighted CPM method the dependent context elements are weighted differently, the errors in the more important dependent contexts, which also have higher weights, will have strong influence to the $PoC(ce_i)$ value. As the consequence, the weighted CPM methods accuracy decreases more rapidly than in case of the non-weighted one. The Table 1 also shows that when there are more erroneous dependent contexts $ce_j \in DC_i$ in reference context elements than the valid ones, in our example this was the case when 3 and more dependent contexts are erroneous, the validation using CPM method is not possible.

5 Summary and Conclusion

In this paper we have discussed the concept and the applicability of feature weighting for the Context Pattern Method (CPM). Our proposed solution contains two types of weights: one for weighting the dependent context elements which the context element under investigation depends on, and another for the Quality of Context (QoC) parameter of the context under investigation. We see the QoC parameter weighting as rather subjective and therefore controlled by the user. The weighting of the dependent context elements, on the other hand, are learned from the reference context patterns (knowledge base) using well known feature weighting methods: chi², mutual information and relief-f.

Our experimental results show a clear advantage of the weighted-CPM method over the non-weighted one. We explicitly explored parameterization of the weighted functions, finding that the choice of certain dependent contexts is critical to the CPM method. We find that the most optimal weighting procedure improves the validation accuracy of the CPM method, achieving up to 10% improvement on the non-weighted one. We further find that the feature weighting procedure does not cope well with errors in dependent contexts. Our results show that with the increase in number erroneous dependent contexts, the weighted CPM methods accuracy decreases more rapidly than the non-weighted one.

For the future work we plan to apply the feature weighting method for dependent contexts selection which should allow autonomic selection of dependent contexts at runtime.

Acknowledgment. VENUS is a research cluster at the interdisciplinary Research Center for Information System Design (ITeG) at Kassel University. We thank Hesse's Ministry of Higher Education, Research, and the Arts for funding the project as part of the research funding program "LOEWE – Landes-Offensive zur Entwicklung Wissenschaftlich-ökonomischer Exzellenz". http://www.iteg.uni-kassel.de/venus

References

1. Brgulja, N., Kusber, R., David, K., Baumgarten, M.: Measuring the Probability of Correctness of Contextual Information in Context Aware Systems. In: Proc. of the IEEE International Conference on Pervasive Intelligence and Computing (PICom 2009), pp. 246–253 (December 2009)
2. Brgulja, N., Kusber, R., David, K.: Validating Context Information in Context Aware Systems. In: 6th International Conference on Intelligent Environments (IE 2010), Kuala Lumpur Malaysia (July 2010) (to appear)
3. Henriksen, K., Indulska, J., Rakotonirainy, A.: Modeling Context Information in Pervasive Computing Systems. In: Mattern, F., Naghshineh, M. (eds.) PERVASIVE 2002. LNCS, vol. 2414, pp. 167–180. Springer, Heidelberg (2002)
4. Dey, A.K., Salber, D., Abowd, G.D.: A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications. Human-Computer Interaction (HCI) Journal 16(2-4), 97–166 (2001)
5. Padovitz, A., Zaslavsky, A., Loke, S.W., Burg, B.: Maintaining Continuous Dependability in Sensor-Based Context-Aware Pervasive Computing Systems. In: Proc. of the 38th Annual Hawaii International Conference on System Sciences, HICSS 2005 (2005)
6. Kira, K., Rendell, L.A.: A practical approach to feature selection. In: Proc. of the Ninth International Conference on Machine Learning, pp. 249–256. Morgan Kaufmann Publishers, San Francisco (1992)
7. Wettschereck, D., Aha, D.W., Mohri, T.: A Review and Empirical Evaluation of Feature Weighting Methods for a Class of Lazy Learning Algorithms. In: Artificial Intelligence Review, vol. 11, pp. 273–314 (1997)
8. Yang, Y., Pedersen, J.O.: A Comparative Study on Feature Selection in Text Categorization. In: Proc. of the 14th International Conference on Machine Learning, pp. 412–420 (1997)
9. Muttitanon, W., Kumar, N., Souris, M.: An Indoor Positioning System (IPS) using Grid Model. Journal of Computer Science 3, 907–913 (2007)

A Formal Model of Reliable Sensor Perception

Tobe Toben

OFFIS e.V., 26121 Oldenburg, Germany
toben@offis.de

Abstract. The safety of autonomously acting systems depends on a reliable assessment of the systems' context. We propose a framework to formalise and analyse both qualitative and quantitative measures of the context quality in terms of safety and precision. The measures are based on order-theoretic arguments by relating the ground truth (as given by the real environment) with the context information (as inferred by the sensors). We derive a hierarchy of qualitative notions that can serve as a high-level requirement description for the sensor and controller implementation of the system. The quantitative part of the framework then allows for an evaluation of a probabilistic variant of the sensor regarding its safety and precision. We in particular treat the analysis of sensor fusion based on evidence theory.

1 Introduction

Context-awareness is an important prerequisite for enabling pervasive computing applications and smart behaviour of systems. Any autonomously acting system needs to have reliable information about the relevant part of its environment in order to exhibit a safe and effective behaviour. Consider for example an autonomously driving vehicle. Here, an important context information is whether some obstacle is located within the planned driving trajectory. This context information is typically acquired by the use of sensors, which monitor the area in front of the vehicle. There exists a number of techniques and algorithms that infer relevant context information from the raw sensor data. In almost every scenario however, the context information is only an abstraction of the real environment of the system. For example, while the sensor may properly detect the presence or absence of an obstacle, the exact kind of obstacle (e.g. whether it is car or a lorry) can often not be detected.

Assessing the Quality of Context (QoC) is relevant for any kind of ubiquitous computing. In general, the quality is influenced by a number of parameters [9,1], like the timeliness, trust-worthiness, completeness and significance of information. Now we are particular interested in autonomous behaviour of safety-critical systems, i.e. systems whose malfunction may cause severe damage or even harm to persons. For these systems, QoC plays an important role [18,12,8] and a thorough investigation of the safety aspects of context becomes indispensable.

In our proposal, we derive two fundamental parameters for determining the quality of context for such kind of systems, namely *safety* and *precision*. When

P. Lukowitz, G. Kortuem, and K. Kunze (Eds.): EuroSSC 2010, LNCS 6446, pp. 94–107, 2010.

sensing a real environment and mapping it to its context representation, a safe context means that the information is not "wrong" with respect to the reality, while a precise context means that the information is not "coarser than necessary". An unsafe context may obviously lead to hazards, and an imprecise context may prevent any meaningful behaviour of the system.

We will formalise both notions of safety and precision by the use of order-theory. To this end, we regard a context perception as an abstraction relation between two "worlds", namely between the real environment and the systems' context. Properties of this relation can then be expressed in terms of two functions going back and forth between these two worlds. While the first function corresponds to the act of *sensing*, the second function reflects how the system *interprets* the context in terms of possible environment situations. Our claim is that in fact the relationship of *both* functions has to be considered in order to define a meaningful notion of safety and precision (Sect. 2). The results of this investigation can be seen as a formal representation of requirements on both the sensors (expressed in terms of the sensing function) and the control algorithms of the system (expressed in terms of the interpretation function).

However, as real-world sensors are typically not fully reliable, the adherence to the sensing function cannot be guaranteed in all cases. To this end, we also propose in Section 3 an extension towards a *quantitative* notion of safety and precision. We do so by relating a probabilistic variant of the sensing function to our formalisation of a (qualitative) safe and precise perception. In Section 4 we discuss related work, and Section 5 concludes.

2 Sensor Perception

As a running example, we consider a simple environment comprising three possible environment situations $\mathcal{E} = \{car, person, nothing\}$, namely the presence of a car, of a person and the absence of any obstacle. A context is represented by means of a set of exclusive context predicates [19]. For example, in order to represent a sensor that detects the presence of obstacles, we use the two predicates $\mathcal{C} = \{occ, free\}$, which allows us to represent four different contexts:

$$C_0 = \emptyset \qquad C_1 = \{occ\} \qquad C_2 = \{free\} \qquad C_3 = \{occ, free\}$$

C_0 represents the impossible context where no predicate is satisfied. C_1 and C_2 represent definite context information, stating that the area is occupied or free, respectively. C_3 represents an indefinite context information where both predicates are possibly true.

As already sketched above, our notion of context quality involves two mappings, namely a sensing ς from an environment situation to a context, and an interpretation ι of a context back to the environment. Then, a safe perception requires a safe interpretation of a sensing $\varsigma(e)$ of an environment situation e, that is, we have to state

$$\text{How does } \iota(\varsigma(e)) \text{ relate to } e?$$

In general, one does not require an equality relation here, because that would imply that the sensing is exact and does not loose any information. Intuitively, to guarantee a safe but not necessarily exact perception, the combination $\iota(\varsigma(e))$ should represent "at least e, but possible more". In the running example, it would be meaningful to require

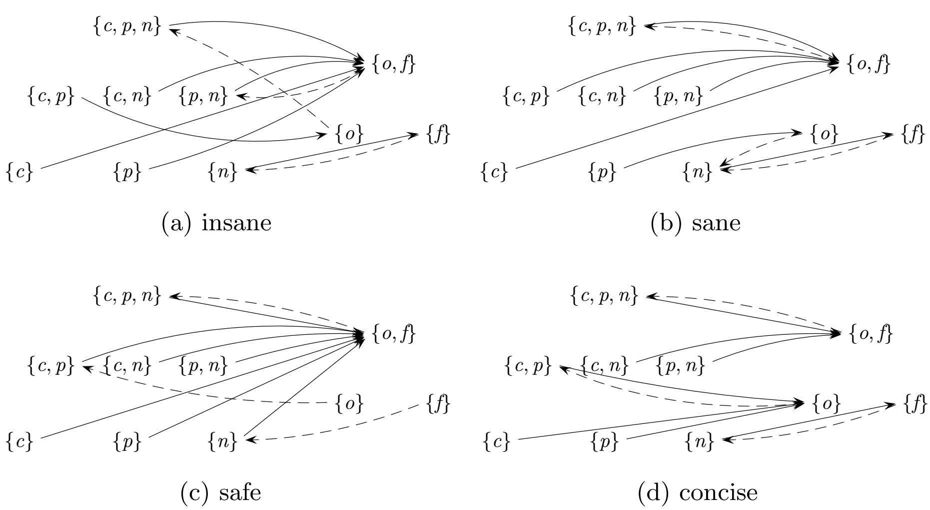
$$\iota(\varsigma(car)) = \{car, person\}$$

if $\varsigma(car)$ yields $\{occ\}$ and the fact that the sensor has detected the presence of an object should be interpreted as having some object, either a car or a person, in the sensed environment. We will formalise this intuition in the following section.

2.1 Classification of Sensor Perception

The powerset of a set S is defined as $\mathcal{P}(S) := \{S' \mid S' \subseteq S\}$. Powersets are a particular instance of a complete lattice [4], that is, a partially ordered set $(S, \leq)$ where any subset has both a least upper bound (lub) and a greatest lower bound (glb). Lattices exhibit a canonical notion of information order induced by the partial order: the relation $S_1 \leq S_2$ states that S_1 represents at least as precise information as S_2. Likewise, $S_1 \lneq S_2$ indicates strictly more precise information of S_1 wrt. S_2. For powersets, the order '$\leq$' is given by set inclusion '$\subseteq$' and the lub and glb are obtained by set union and intersection, respectively. Note that in our example, we in particular have $C_1 \subsetneq C_3$ and $C_2 \subsetneq C_3$.

We now define and characterise a perception based on the properties of its sensing and interpretation. We will first motivate these properties, then formally define them in Def. 2 and finally give examples for some properties in Fig. 1.



(a) insane (b) sane

(c) safe (d) concise

Fig. 1. Characterisation of perceptions by analysing the composition of sensing (solid arrows) and interpretation (dashed arrows) functions

Definition 1 (Perception). *A tuple* $(\mathcal{E}, \varsigma, \iota, \mathcal{C})$ *where*

- $\mathcal{E}$ *is a finite set of environment situations,*
- $\mathcal{C}$ *is a finite set of exclusive context predicates,*
- $\varsigma : \mathcal{P}(\mathcal{E}) \to \mathcal{P}(\mathcal{C})$ *is a sensing function, and*
- $\iota : \mathcal{P}(\mathcal{C}) \to \mathcal{P}(\mathcal{E})$ *is an interpretation function,*

is called a perception. ♢

A first criterion is whether both functions preserve the ordering of information, that is, whether they are monotone functions. The second criterion addresses the combination of sensing and interpretation. Starting with an argument e of the sensing function, a suitable characterisation for being safe is that the interpretation of the obtained sensing yields e or some less precise information than e. However, we should not obtain information that is more precise or unrelated to e. The third criterion considers the combination of first applying the interpretation and then the sensing function. Again, we will demand a proper ordering of information, however in the opposite direction as for safety. As we will see below, this property allows us to characterise the (qualitative) precision of a perception. Moreover, we can strengthen the previous two criteria by requiring equality instead of orderedness.

Definition 2 (Perception Classification). *A perception* $(\mathcal{E}, \varsigma, \iota, \mathcal{C})$ *is called*

- *sane if* ς *and* ι *are monotone functions, i.e.*

$$E_1 \subseteq E_2 \implies \varsigma(E_1) \subseteq \varsigma(E_2) \quad \textit{and} \quad O_1 \subseteq O_2 \implies \iota(O_1) \subseteq \iota(O_2)$$

- *safe if it is sane and* $\forall E \in \mathcal{P}(\mathcal{E}) : E \subseteq \iota(\varsigma(E))$*,*
- *precise if it is safe and* $\forall O \in \mathcal{P}(\mathcal{C}) : \varsigma(\iota(O)) \subseteq O$*,*
- *concise if it is precise and* $\forall O \in \mathcal{P}(\mathcal{C}) : \varsigma(\iota(O)) = O$*,*
- *exact if it is concise and* $\forall E \in \mathcal{P}(\mathcal{E}) : \iota(\varsigma(E)) = E$*.* ♢

A number of perceptions for the running example are given in Figure 1. For space reasons, all elements are abbreviated by their first letter, and the relation between the empty sets, $\varsigma(\emptyset) = \emptyset$ and $\iota(\emptyset) = \emptyset$, are omitted.

Perception 1a is not sane since neither ι nor ς are monotone. For example, we have $\iota(\{o, f\}) = \{c, p\}$ and $\iota(\{o\}) = \{c, p, n\}$, that is, ι derives more precise information from $\{o, f\}$ than from $\{o\}$. Perception 1b is sane but not safe. We have $\varsigma(\{p\}) = \{o\}$ and $\iota(\{o\}) = \{n\}$, hence $\{p\} \not\subseteq \iota(\varsigma(\{p\}))$. In fact, interpreting the sensing of a person as the empty situation should not be considered to be a safe perception. Perception 1c is safe as the sensing of every non-empty environment is interpreted to be the universal environment $\mathcal{E}$. This coarseness however leads to, e.g., $\iota(\{f\}) = \{n\}$ and $\varsigma(\{n\}) = \{o, f\}$, hence $\varsigma(\iota(\{f\})) \not\subseteq \{f\}$. Thus, the perception is not precise according to Def. 2. Perception 1d is concise, hence also precise. It matches with our intuition that both $\{c\}$ (car) and $\{p\}$ (person) are merged to $\{o\}$ (occ), and $\{n\}$ (nothing) to $\{f\}$ (free). The indefinite context occurs if a combination of n with c or p is sensed. Also, the interpretation yields the most precise value with respect to the given sensing function, e.g. $\iota(\{o\}) = \{c, p\}$. Note that precision is to be understood with respect to $\mathcal{C}$, that is, a precise perception preserves as much information as possible, given the set of context predicates. In our example, the kind of obstacle is dismissed as both situations c and p are sensed to the same context. In contrast, an exact perception has no loss of information at all, which is only possible if $|\mathcal{E}| = |\mathcal{C}|$.

The most relevant notions are that of *precise* and *concise* perceptions. A perception that is not precise does not effectively use its potential. Conciseness yields additional benefits by ensuring that ς becomes surjective (hence every possible context is actually used by the perception) and ι becomes injective (hence we have meaningful interpretations for all non-empty contexts, see below). To require an exact perception seems not to be realistic for real-world applications.

2.2 Relation to Galois Connections

From an order-theoretic perspective [4], the notion of a precise perception corresponds to the definition of a *Galois connection*, and a concise perception to that of a *Galois insertion*. A number of results from the theory of Galois connection immediately transfers to our characterisation of precise and concise perceptions. The proofs of these results can be found in [11]. The first proposition states that a precise perception is fully determined by either one of the sensing and interpretation function.

Proposition 1 (Determination). *Let $(\mathcal{E}, \varsigma, \iota, \mathcal{C})$ be a precise perception. Then ς uniquely determines ι by $\iota(O) = \bigcup\{E \mid \varsigma(E) \subseteq O\}$ and ι uniquely determines ς by $\varsigma(E) = \bigcap\{O \mid \iota(O) \subseteq E\}$ for $E \subseteq \mathcal{E}$ and $C \subseteq \mathcal{C}$.* ♢

Related to this observation, we can automatically obtain a precise perception via a basic sensor function that maps from $\mathcal{E}$ to $\mathcal{P}(\mathcal{C})$.

Proposition 2 (Basic sensor function). *Let $\varsigma_0 : \mathcal{E} \to \mathcal{P}(\mathcal{C})$ be a function. Then the perception $(\mathcal{E}, \varsigma, \iota, \mathcal{C})$ with*

$$\varsigma(E) := \bigcup_{e \in E} \varsigma_0(e)$$

$$\iota(C) := \{e \in \mathcal{E} \mid \varsigma_0(e) \subseteq C\}$$

for $E \subseteq \mathcal{E}$ and $C \subseteq \mathcal{C}$ is precise. ♢

This yields a methodology for constructing precise perceptions by first choosing the sets $\mathcal{E}$ and $\mathcal{C}$ according to the real-word scenario, then defining a basic sensor function $\varsigma_0 : \mathcal{E} \to \mathcal{P}(\mathcal{C})$ which describes for each environment situation a suitable context, and apply proposition 2. We illustrate this construction on the running example, by using a basic sensor function ς_0 with

$$\varsigma_0(c) = \{o\} \qquad \varsigma_0(p) = \{f\} \qquad \varsigma_0(n) = \{f\}$$

This yields a perception $(\mathcal{E}, \varsigma, \iota, \mathcal{C})$ with $\varsigma(\{e\}) = \varsigma_0(e)$ for $e \in \mathcal{E}$ and

$$\begin{array}{lll} \varsigma(\{c,p\}) = \{o,f\} & \varsigma(\{c,p,n\}) = \{o,f\} & \iota(\{f\}) = \{p,n\} \\ \varsigma(\{c,n\}) = \{o,f\} & \varsigma(\emptyset) = \emptyset & \iota(\{o\}) = \{c\} \\ \varsigma(\{p,n\}) = \{f\} & \iota(\emptyset) = \emptyset & \iota(\{o,f\}) = \{c,p,n\} \end{array}$$

We obtain a rather unintuitive perception as e.g. both the environments '$\{p\}$' (person) and '$\{n\}$' (nothing) are sensed to the context '$\{f\}$' (free). As this context however is appropriately interpreted by ι, the perception as a whole is safe and precise. This again underlines the importance of considering both mappings to obtain a meaningful characterisation of context quality. Moreover, this example demonstrates that there is not a unique precise perception for a given pair of environment and context. Note that this construction also ensures that only the empty environment is sensed to the empty (impossible) context. Additionally, if every environment situation e is in the image of the basic sensor function, we obtain a concise perception in which ι becomes injective, and hence the same holds for the interpretation function (that is, $\iota(C) = \emptyset$ implies $C = \emptyset$).

Finally, the theory of Galois connections equips us with a notion of compositionality. This is of practical relevance whenever the context recognition cannot be performed in one sweep but requires multiple stages of sensing. In this case, the following result allows us to analyse each stage in isolation and still conclude properties for the whole perception chain.

Proposition 3 (Composition). *Let $(\mathcal{E}, \varsigma, \iota, \mathcal{C})$ and $(\mathcal{E}', \varsigma', \iota', \mathcal{C}')$ be precise perceptions with $\mathcal{C} = \mathcal{E}'$. Then $(\mathcal{E}, \varsigma' \circ \varsigma, \iota \circ \iota', \mathcal{C}')$ is a precise perception.* $\diamondsuit$

2.3 Discussion

The presented classification of perceptions allows for a qualitative assessment in terms of safety and precision. In particular, our framework can be seen as a formalisation of system requirements, separated into requirements on the sensor part (given by the sensing function) and requirements on the controller part (given by the interpretation function). These requirements express the desired mappings for each environment situation and context. Still, the sensing function can in particular map a singleton environment situation to a *set* of exclusive context predicates, thereby expressing an inherent uncertainty in observing this environment situation. We will see that this mapping affects the notion of quantitative safety and precision as defined below.

All in all, the characterisation of a precise or concise perception as a *reference* can help to exclude systematic errors in the design of the perception process. However, a perfect realisation of the sensing function based on hardware sensors and corresponding software algorithms is in general not possible. This is as real-words sensors comprise an inherent amount of inaccuracy and may be disturbed by a number of external influences. Also, software algorithms typically employ heuristics for deriving context information out of raw data and thus cannot be fully reliable in general. In order to account for this fact, we propose in the following a quantitative extension of our approach in order to be able to measure the overall quality (both in terms of safety and precision) of a probabilistic variant of the sensing function with respect to a given perception.

3 Uncertainty

As motivated above, real-world sensors are in general not fully reliable and hence cannot be faithfully formalised as a function that maps each environment situation to one dedicated context. Rather, a sensor corresponds to a probabilistic mapping to different contexts. For example, while a sensor will most of the time (e.g. with probability 0.99) properly detect a car, in some rare cases (e.g. with probability 0.01) it fails and erroneously yields e.g. the value *free*.

Given a finite set S, a basic probability assignment (bpa) over S is a function $p : \mathcal{P}(S) \rightarrow [0, 1]$ with $p(\emptyset) = 0$ and $\sum_{S' \subseteq S} p(S') = 1$. The set of all basic probability assignments over S is denoted by $BPA(S)$. With this, a basic sensor function can be generalised into its probabilistic variant as follows.

Definition 3 (Uncertain Sensor). *Let $P = (\mathcal{E}, \varsigma, \iota, \mathcal{C})$ be a perception. A function*

$$\tilde{\varsigma} : \mathcal{E} \rightarrow BPA(\mathcal{C})$$

that maps each environment situation to a bpa over $\mathcal{C}$ is called an uncertain sensor *over P. The set of all uncertain sensors over P is denoted by $US(P)$.* ♢

By using a bpa, an uncertain sensor can map dedicated probability masses in particular to non-singleton contexts, thereby expressing a certain amount of inconclusive observations and even uncommitted belief (assignments to $\mathcal{C}$).

3.1 Sensor Quality

An uncertain sensor expresses to which amount it yields different outcomes than the sensing function as given by the perception. For the running example, an uncertain sensor with

$$\tilde{\varsigma}(\{c\})(\{o\}) = 0.7 \qquad \tilde{\varsigma}(\{c\})(\{f\}) = 0.1 \qquad \tilde{\varsigma}(\{c\})(\{o, f\}) = 0.2$$

states that with a probability of 70% the sensor coincides with ς but with 10% it yields $\{f\}$ and with 20% $\{o, f\}$. By taking the interpretation function into account we can classify this amount of incorrect sensing into unsafety and imprecision: As $\{c\} \not\subseteq \iota(\{f\})$, $\tilde{\varsigma}$ is 10% unsafe, and as $\{c\} \subsetneq \iota(\{o, f\})$ it is 20% imprecise. We generalise this observation in the following definitions.

Definition 4 (Environment Classification). *Let $P = (\mathcal{E}, \varsigma, \iota, \mathcal{C})$ be a perception and $e \in \mathcal{E}$ an environment situation. Then*

$$\begin{aligned} Cor_P(e) &= \{E \in \mathcal{P}(\mathcal{E}) \mid \iota(\varsigma(\{e\})) = \iota(\varsigma(E))\} \\ Imp_P(e) &= \{E \in \mathcal{P}(\mathcal{E}) \mid \iota(\varsigma(\{e\})) \subsetneq \iota(\varsigma(E))\} \\ Uns_P(e) &= \mathcal{P}(\mathcal{E}) \setminus \big(Cor_P(e) \cup Imp_P(e)\big) \end{aligned}$$

denote the correct, imprecise and unsafe environments for e, respectively. ♢

To motivate this definition, we introduce a more elaborated example which is inspired by [17]. The task is to measure the distance to an object within a radius of zero to nine units. The sensor has a rather low resolution and can only distinguish between the distances *(c)lose*, *(n)ear* and *(f)ar*. More specifically, setting $\mathcal{E} = \{0, \ldots, 9\}$ and $\mathcal{C} = \{c, n, f\}$, the basic sensor function is

$$\varsigma_0(e) = \begin{cases} \{c\} & \textit{if } 0 \leq e \leq 2 \\ \{n\} & \textit{if } 3 \leq e \leq 5 \\ \{n, f\} & \textit{if } e = 6 \\ \{f\} & \textit{if } 7 \leq e \leq 9 \end{cases}$$

and ς and ι are obtained by proposition 2. For example, we obtain $\varsigma(\{2, 3\}) = \{c, n\}$, and $\iota(\{n\}) = \{3, 4, 5\}$, and $\iota(\{n, f\}) = \{3, \ldots, 9\}$. We will refer to this perception by the name D in the following.

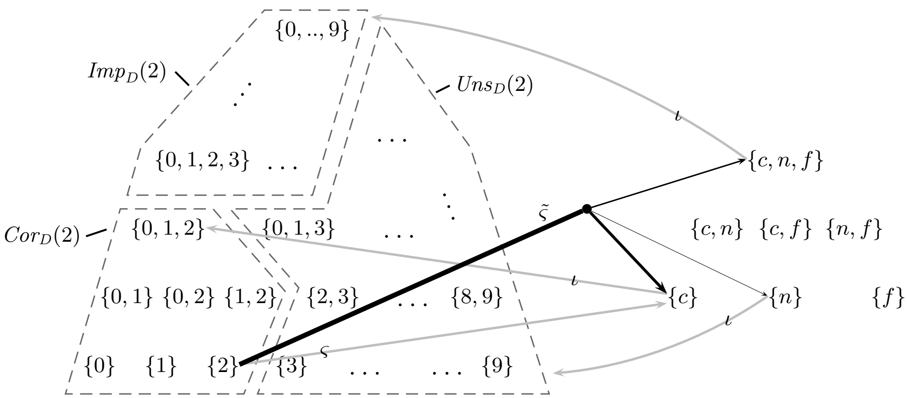


Fig. 2. Illustration of Def. 4 and Def. 5 for $e = 2$ with $\iota(\varsigma(\{2\})) = \{0, 1, 2\}$

Figure 2 illustrates the concepts of Def. 4. For the environment situation 2, the sensing function derives the context $\{c\}$ which is interpreted as $\{0, 1, 2\} =: E$. This induces a partitioning into correct environments $Cor_D(2)$ (subsets of E and E itself), imprecise environments $Imp_D(2)$ (proper supersets of E), and unsafe environments $Uns_D(2)$ (neither subsets nor supersets of E). Note that in particular $\{0\}$ is considered to be a correct environment for $\{2\}$, basically as the perception cannot distinguish them anyway.

We obtain a measure of unsafety and imprecision by summing the probability masses of those contexts which are interpreted to environments of the corresponding partition. This is illustrated in Fig. 2 by the $\tilde{\varsigma}$-labelled arrow which branches the sensing into different contexts with certain probabilities.

Definition 5 (Environment Quality). *Let* $P = (\mathcal{E}, \varsigma, \iota, \mathcal{C})$ *be a perception and* $\tilde{\varsigma} \in \mathit{US}(P)$ *an uncertain sensor. The environment quality of* $\tilde{\varsigma}$ *for an environment situation* $e \in \mathcal{E}$ *is determined by*

$$\mathit{Uns}_{\tilde{\varsigma}}(e) := \sum_{O \in \mathcal{P}(\mathcal{C}),\, \iota(O) \in \mathit{Uns}_P(e)} \tilde{\varsigma}(e)(O)$$

$$\mathit{Imp}_{\tilde{\varsigma}}(e) := \sum_{O \in \mathcal{P}(\mathcal{C}),\, \iota(O) \in \mathit{Imp}_p(e)} \tilde{\varsigma}(e)(O)$$

which quantify the amount of unsafety and imprecision, respectively.

The average values over all environment situations are denoted by $\mathit{Uns}_{\tilde{\varsigma}}(\mathcal{E}) := 1/|\mathcal{E}| \cdot \sum_{e \in \mathcal{E}} \mathit{Uns}_{\tilde{\varsigma}}(e)$ *and* $\mathit{Imp}_{\tilde{\varsigma}}(\mathcal{E}) := 1/|\mathcal{E}| \cdot \sum_{e \in \mathcal{E}} \mathit{Imp}_{\tilde{\varsigma}}(e)$. ♢

We can obtain a similar notion of quality from the *perspective of a context*. This is of particular relevance for practical applications as the system only has information about the context as given by its sensor implementation, and does not know the real environment. A notion of context quality then allows for an estimation how reliable this context information with respect to the actual environment situation is. Formally, the quantity can be computed as the conditional probability of unsafe and imprecise sensings with respect to the probability for all possible environment situations that map to a certain context.

Definition 6 (Context Quality). *Let* $P = (\mathcal{E}, \varsigma, \iota, \mathcal{C})$ *be a perception and* $\tilde{\varsigma} \in \mathit{US}(P)$ *an uncertain sensor. The context quality of* $\tilde{\varsigma}$ *for a context* $O \in \mathcal{P}(\mathcal{C})$ *is determined by*

$$\mathit{Uns}_{\tilde{\varsigma}}(O) := \frac{\sum_{e \in \mathcal{E},\, \iota(O) \in \mathit{Uns}_P(e)} \tilde{\varsigma}(e)(O)}{\sum_{e \in \mathcal{E}} \tilde{\varsigma}(e)(O)}$$

$$\mathit{Imp}_{\tilde{\varsigma}}(O) := \frac{\sum_{e \in \mathcal{E},\, \iota(O) \in \mathit{Imp}_P(e)} \tilde{\varsigma}(e)(O)}{\sum_{e \in \mathcal{E}} \tilde{\varsigma}(e)(O)}$$

which quantify the amount of unsafety and imprecision, respectively. ♢

For example, we consider a sensor $\tilde{\varsigma}_1 \in \mathit{US}(D)$ that is well suited to detect close objects, but tends to make errors for larger distances. The probability function of this sensor is given in Table 1. The last two columns given the amount of unsafety and imprecision for each environment situation for $\tilde{\varsigma}_1$ with respect to D according to Def. 5, and the last two rows gives the corresponding values for the contexts according to Def. 6.

In the example, sensing the environment 7 yields with a probability of 0.3 an unsafe context. This is as $\iota(\varsigma(\{7\}))$ is $\{7, 8, 9\}$ and hence both sets $i(\{c\}) = \{0, 1, 2\}$ and $i(\{n\}) = \{3, 4, 5\}$ are in $\mathit{Uns}_D(7)$. Thus $\mathit{Uns}_{\tilde{\varsigma}_1}(7) = \tilde{\varsigma}(7)(\{c\}) + \tilde{\varsigma}(7)(\{n\}) = 0.1 + 0.2 = 0.3$. Analogously, we obtain a probability of 0.2 for being imprecise as both $\iota(\{n, f\})$ and $\iota(\{c, n, f\})$ are elements of $\mathit{Imp}_D(7)$. The average amount of unsafety and imprecision of $\tilde{\varsigma}_1$ is 0.09 and 0.23, respectively.

As an example for context quality, we consider the context $\{c, n\} =: O$. The sum of probabilities over all environments is $\sum_{e \in \mathcal{E}} \tilde{\varsigma}(e)(O) = 0.55$. We

Table 1. Uncertain sensor $\tilde{\varsigma}_1$ together with its quality masses. The gray cells visualise the sensing function given by the perception D.

e \ O	$\{c\}$	$\{n\}$	$\{f\}$	$\{c,n\}$	$\{n,f\}$	$\{c,f\}$	$\{c,n,f\}$	$Uns_{\tilde{\varsigma}_1}(e)$	$Imp_{\tilde{\varsigma}_1}(e)$
0	1.0							0	0
1	0.95			0.05				0	0.05
2	0.9			0.1				0	0.1
3		0.8		0.2				0	0.2
4		0.9		0.1				0	0.1
5		0.65	0.05		0.3			0.05	0.3
6		0.15	0.15		0.6		0.1	0.3	0.1
7	0.1	0.2	0.5		0.15		0.05	0.3	0.2
8		0.1	0.4		0.2		0.3	0.1	0.5
9			0.3	0.1			0.7	0.1	0.7
$Uns_{\tilde{\varsigma}_1}(O)$	0.03	0.11	0.14	0.18	0	-	0	$Uns_{\tilde{\varsigma}_1}(\mathcal{E})$	$Imp_{\tilde{\varsigma}_1}(\mathcal{E})$
$Imp_{\tilde{\varsigma}_1}(O)$	0	0	0	0.82	0.35	-	1	0.09	0.23

have $\iota(O) = \{0, \ldots, 5\} \in Imp_D(e)$ for $e \in \{1, 2, 3, 4\}$. This yields $Imp_{\tilde{\varsigma}_1}(O) = (0.05 + 0.1 + 0.2 + 0.1)/0.55 = 0.82$. Analogously, as $\iota(O) \in Uns_D(9)$ we obtain $Uns_{\tilde{\varsigma}_1}(O) = 0.1/0.55 = 0.18$. Note that we have $Uns_{\tilde{\varsigma}_1}(O) + Imp_{\tilde{\varsigma}_1}(O) = 1$ due to the fact that $\{c, n\}$ is not in the image of ς_0, and hence there is no environment which *correctly* maps to it.

All in all, the quality values reflect the intuition that the sensor is rather safe when delivering a (c)lose distance (0.03), but becomes unsafer for distances (n)ear and (f)ar (0.11 and 0.14). Additionally, from $Imp_{\tilde{\varsigma}_1}(\mathcal{E}) = 0.23$ we see that the sensor gives an imprecise reading in almost every forth case (assuming that all environment situations occur with an equal probability).

3.2 Properties of the Quality Mass

We present a number of properties regarding our quality mass. The first observation is a direct consequence of the definition of an uncertain sensor $\tilde{\varsigma}$ based on basic probability assignments, and of the fact that the sets of correct, imprecise and unsafe environment indeed form a partition of all possible environments.

Proposition 4 (Quality Mass). *Let $P = (\mathcal{E}, \varsigma, \iota, \mathcal{C})$ be a perception and $\tilde{\varsigma} \in US(P)$ an uncertain sensor. Then both the environment and the context quality lies in $[0, 1]$, that is,*

(i) $0 \leq Uns_{\tilde{\varsigma}}(e) + Imp_{\tilde{\varsigma}}(e) \leq 1$
(ii) $0 \leq Uns_{\tilde{\varsigma}}(O) + Imp_{\tilde{\varsigma}}(O) \leq 1$

for any environment $e \in \mathcal{E}$ and context $O \in \mathcal{P}(\mathcal{C})$. ♢

For the context quality, we have certain probabilities fixed to zero for singleton contexts and the universal context. When working with *safe* perceptions, the context $\mathcal{C}$ cannot be unsafe as is represents all possible environments. Similarly, the imprecision of singleton contexts is zero as, due to the monotonicity, there is no smaller context which is potentially more precise.

Proposition 5 (Definite Context Quality). *Let* $P = (\mathcal{E}, \varsigma, \iota, \mathcal{C})$ *be a safe perception and* $\tilde{\varsigma} \in US(P)$ *an uncertain sensor. Then* $Uns_{\tilde{\varsigma}}(\mathcal{C}) = 0$ *and* $Imp_{\tilde{\varsigma}}(\{o\}) = 0$ *for any* $o \in \mathcal{C}$. ♢

For an *exact* perception, we observe a relationship of our quality mass to the classical notion of belief and plausibility in the Dempster-Shafer framework [5,14]. Given a finite set Ω and a bpa $m \in BPA(\Omega)$, the two functions

$$Bel_m(X) := \sum_{X' \subseteq X} m(X') \qquad\qquad Pl_m(X) := \sum_{X' \cap X \neq \emptyset} m(X')$$

define the belief and the plausibility regarding a subset $X \subseteq \Omega$, respectively. Assuming an exact perception $P = (\mathcal{E}, \varsigma, \iota, \mathcal{C})$ and an uncertain sensor $\tilde{\varsigma} \in US(P)$ with $\tilde{\varsigma}(e) = m$, we have

$$Pl_m(\{e\}) = 1 - Uns_{\tilde{\varsigma}}(e) \tag{1}$$

and

$$Bel_m(\{e\}) = 1 - (Uns_{\tilde{\varsigma}}(e) + Imp_{\tilde{\varsigma}}(e)) \tag{2}$$

for any environment situation e. In particular, the difference between belief and plausibility corresponds to our notion of imprecision.

The proof for this relationship exploits that $\iota(\varsigma(\{e\}))$ equals $\{e\}$ in an exact perception. Then the belief measure exactly corresponds to our notion of correct environments according to Def. 4, which entails equation 2. Moreover, as

$$Cor_P(e) \cup Imp_P(e) = \{E' \in \mathcal{E} \mid E' \cap \{e\} \neq \emptyset\}$$

we have $Bel_m(\{e\}) + Imp_{\tilde{\varsigma}}(e) = Pl_m(\{e\})$, which entails equation 1.

3.3 Sensor Fusion

The relation to the Dempster-Shafer framework also allows us to evaluate the effects of sensor fusion. To this end, we choose another uncertain sensor $\tilde{\varsigma}_2$ which gives good results as long as the obstacle is near or far, but is very imprecise on smaller distances. The bpa are given in Table 2. We see that the sensor does not give any unsafe output, but produces imprecise results on about every second reading. Combining the information of both sensors $\tilde{\varsigma}_1$ and $\tilde{\varsigma}_2$ will hopefully yield a sensor that gives safe and precise results for the complete set of environments.

Dempster's rule of combination [5] merges two bpa $m_1, m_2 \in BPA(S)$ by setting $(m_1 \oplus m_2)(\emptyset) = \emptyset$ and

$$(m_1 \oplus m_2)(X) = \frac{\sum_{A \cap B = X} m_1(A) \cdot m_2(B)}{1 - \sum_{A \cap B = \emptyset} m_1(A) \cdot m_2(B)}$$

for $X \in \mathcal{P}(S) \setminus \{\emptyset\}$. Given a set of hypotheses X, this rule combines individual probability masses whenever there is an agreement (in the sense of intersection)

Table 2. Uncertain sensor $\tilde{\varsigma}_2$ together with its quality masses

e \ O	$\{c\}$	$\{n\}$	$\{f\}$	$\{c,n\}$	$\{n,f\}$	$\{c,f\}$	$\{c,n,f\}$	$Uns_{\tilde{\varsigma}_2}(e)$	$Imp_{\tilde{\varsigma}_2}(e)$
0							1.0	0	1.0
1							1.0	0	1.0
2	0.1			0.2			0.7	0	0.9
3		0.2		0.2			0.6	0	0.8
4		0.5		0.2			0.3	0	0.5
5		0.4			0.5		0.1	0	0.6
6					0.6		0.4	0	0.4
7			0.7		0.1		0.2	0	0.3
8			0.9				0.1	0	0.1
9			1.0					0	0
$Uns_{\tilde{\varsigma}_2}(O)$	0	0	0	0	0	-	0	$Uns_{\tilde{\varsigma}_2}(\mathcal{E})$	$Imp_{\tilde{\varsigma}_2}(\mathcal{E})$
$Imp_{\tilde{\varsigma}_2}(O)$	0	0	0	1	0.5	-	1	0	0.56

Table 3. Fusion of $\tilde{\varsigma}_3 = \tilde{\varsigma}_1 \oplus \tilde{\varsigma}_2$ together with its quality masses

e \ O	$\{c\}$	$\{n\}$	$\{f\}$	$\{c,n\}$	$\{n,f\}$	$\{c,f\}$	$\{c,n,f\}$	$Uns_{\tilde{\varsigma}_3}(e)$	$Imp_{\tilde{\varsigma}_3}(e)$
0	1.0							0	0
1	0.95			0.05				0	0.05
2	0.91			0.09				0	0.09
3		0.84		0.16				0	0.16
4		0.95		0.05				0	0.05
5		0.75	0.03		0.18			0.03	0.18
6		0.15	0.15		0.66		0.04	0.3	0.04
7	0.03	0.08	0.82		0.06		0.01	0.11	0.07
8		0.01	0.94		0.02		0.03	0.01	0.05
9			1.0					0	0
$Uns_{\tilde{\varsigma}_3}(O)$	0.01	0.09	0.06	0	0	-	0	$Uns_{\tilde{\varsigma}_3}(\mathcal{E})$	$Imp_{\tilde{\varsigma}_3}(\mathcal{E})$
$Imp_{\tilde{\varsigma}_3}(O)$	0	0	0	1	0.28	-	1	0.05	0.07

of both bpa regarding the set X. Conflicting evidence where the intersection is empty is distributed over all values by a normalisation factor.

Under the assumption that both sensors are independent sources, we can use this rule to combine the individual bpa for each environment situation. The result for the example is given in Table 3. We see that the safe reading of $\tilde{\varsigma}_2$ for (in particular) near and large distances is able to reduce the amount of unsafety for the corresponding contexts. Also, the overall amount of imprecision is reduced to 0.07. Note that the unsafety of environment situation 6 can not be reduced by the rule of combination as given above. This is however of no particular surprise as $\varsigma_0(6) = \{n, f\}$ violates the Dempster-Shafer assumption of working with *exclusive* hypotheses. A possible solution would either be to focus on basic sensor functions that map to singleton contexts only, or to integrate the modified theory of Dempster-Shafer (DSmt, see [6]) where the assumption of exclusive hypotheses is removed. Technically, this requires to go from powersets to hyperpowersets in order to represent disjunctive versus conjunctive hypotheses.

4 Related Work

Context-awareness [13,1], context reasoning [16,1] and Quality of Context [9,10] have been studied as general concepts and methods for pervasive computing applications. Without a good "understanding" of their working context, such devices cannot realise the vision of smart and ubiquitous computing. As soon as smart devices start to act autonomously in a heterogeneous environment, a new "quality of quality" has to be considered as wrong context information can lead to severe hazards. This problem has in particular been recognised for autonomous vehicles [18,12,8], however no widely accepted formal framework for a thorough investigation of safety-critical context information exists.

Lattice theory is already heavily used in context modelling and reasoning. For example, formal concept analysis [7] derives a structured view of large amounts of raw context data by means of ordered structures (concept lattices). This is used both as a machine learning technique and for data analysis. Situations are considered as a semantic interpretation of context information, and lattices of situations [19] capture the dependence between situations in a structured way. Run-time learning on the basis of lattices is for example done in [20].

In the framework of Abstract Interpretation [3,11], Galois connections are used for constructing safe approximations of program semantics. There, the credo is to "analyse the abstract and conclude to the concrete behaviour" of (in particular safety-critical) software by means of formal methods. The same principle applies to our setting as we "decide on the context and act in the environment". In fact, this "analogy" was the starting point for our investigation.

5 Conclusion

Based on the relation between the real environment and the derived context, we presented a hierarchy of quality notions which are inspired by the theory of Galois connections. Using these notions as a basic qualitative assessment, we developed an evaluation method for the intrinsic uncertainty of real-world sensor applications. This method derives a measure of safety and precision, which are the two relevant parameters for safety and liveness aspects [15] of systems. We showed that our approach generalises the classical belief/plausibility measures to non-exact perceptions, i.e. perceptions with an inherent loss of information.

To keep this paper focussed on the main ideas, we used a rather simple context and environment representation. However, our approach is extendable to any lattice-structured representation, and in future work we will transfer the ideas to more sophisticated context modelling approaches [1]. The integration of techniques like DSmt [6] or conflict metrics [2] is possible and desirable. Also, an evaluation with realistic sensor data will investigate the applicability of our high-level formalisation to a real-world implementation of a sensor perception.

References

1. Bettini, C., Brdiczka, O., Henricksen, K., Indulska, J., Nicklas, D., Ranganathan, A., Riboni, D.: A survey of context modelling and reasoning techniques. Pervasive and Mobile Computing 6(2), 161–180 (2010)
2. Carlson, J., Murphy, R.R., Chistopher, S., Casper, J.: Conflict Metric as a Measure of Sensing Quality. In: ICRA, pp. 2032–2039. IEEE, Los Alamitos (2005)
3. Cousot, P., Cousot, R.: Abstract Interpretation: A Unified Lattice Model for Static Analysis of Programs. In: POPL, pp. 238–252 (1977)
4. Davey, B.A., Priestley, H.A.: Introduction to Lattices and Order. Cambridge University Press, Cambridge (1990)
5. Dempster, A.P.: A generalization of Bayesian inference. Journal of the Royal Statistical Society 30(B), 205–247 (1968)
6. Dezert, J., Smarandache, F.: DSmT: A new paradigm shift for information fusion. CoRR abs/cs/0610175 (2006)
7. Ganter, B., Wille, R.: Formal concept analysis: Mathematical foundations. Springer, Heidelberg (1999)
8. Gehrke, J.D.: Evaluating Situation Awareness of Autonomous Systems. In: Madhavan, R., Messina, E. (eds.) Performance Metrics for Intelligent Systems Workshop. pp. 206–213. NIST (2008) Published on CD-ROM
9. Manzoor, A., Truong, H.L., Dustdar, S.: On the Evaluation of Quality of Context. In: Roggen, D., Lombriser, C., Tröster, G., Kortuem, G., Havinga, P.J.M. (eds.) EuroSSC 2008. LNCS, vol. 5279, pp. 140–153. Springer, Heidelberg (2008)
10. McKeever, S., Ye, J., Coyle, L., Dobson, S.: A Context Quality Model to Support Transparent Reasoning with Uncertain Context. In: Rothermel, K., Fritsch, D., Blochinger, W., Dürr, F. (eds.) QuaCon 2009. LNCS, vol. 5786, pp. 65–75. Springer, Heidelberg (2009)
11. Nielson, F., Nielson, H.R., Hankin, C.: Principles of Program Analysis. Springer, New York (1999)
12. Savi, R., Limbourg, P.: Aggregating uncertain sensor information for safety related systems. In: European Safety and Reliability Conference, ESREL (2006)
13. Schilit, B., Adams, N., Want, R.: Context-aware computing applications. In: IEEE Workshop on Mobile Computing Systems and Applications, Santa Cruz, US (1994)
14. Shafer, G.: A Mathematical Theory of Evidence. Princeton University Press, Princeton (1976)
15. Sistla, A.P.: Safety, liveness and fairness in temporal logic. Formal Asp. Comput. 6(5), 495–512 (1994)
16. Sun, J., Wu, Z.: Context Reasoning Technologies in Ubiquitous Computing Environment. In: Sha, E.H.M., et al. (eds.) EUC 2006. LNCS, vol. 4096, pp. 1027–1036. Springer, Heidelberg (2006)
17. Voorbraak, F.: Combining Unreliable Pieces of Evidence. Tech. rep., University of Amsterdam (1995)
18. Wardzinski, A.: The Role of Situation Awareness in Assuring Safety of Autonomous Vehicles. In: Górski, J. (ed.) SAFECOMP 2006. LNCS, vol. 4166, pp. 205–218. Springer, Heidelberg (2006)
19. Ye, J., Coyle, L., Dobson, S., Nixon, P.: Using Situation Lattices in Sensor Analysis. In: PerCom, pp. 1–11. IEEE Computer Society, Los Alamitos (2009)
20. Zenou, E., Samuelides, M.: Galois Lattice Theory for Probabilistic Visual Landmarks. J. UCS 10(8), 1014–1033 (2004)

Effect of Caching in a Broker Based Context Provisioning System

Saad Liaquat Kiani[1], Michael Knappmeyer[1,2], Eike Steffen Reetz[1,2], Nigel Baker[1], and Ralf Tönjes[2]

[1] University of the West of England, Bristol, BS16 1QY, United Kingdom
[2] University of Applied Sciences Osnabrück, 49009 Osnabrück, Germany

Abstract. Caching is a well-established mechanism used in distributed systems for improving overall performance. In this paper, we analyse the effect of using a context cache in a broker-based context provisioning system. An experiment is carried out using a simulation based on our framework architecture of context consumers, context broker and context providers exchanging different types of context data over time. The results show notable improvement in the context query response time.

Keywords: Context Broker, Context Cache, Performance Evaluation.

1 Introduction

Communicating context data is one of the fundamental functional tasks of context provisioning systems in ubiquitous environments. A number of context-aware systems have been developed that perform this task using different architectures [5][1][24]. A common architectural design used in the development of a context-aware system is the centralised server model where a central server or broker aids context consumers in acquisition of context data. Our architecture has been developed in the C-CAST project whose main objective is to evolve mobile multimedia multicasting to exploit the increasing integration of mobile devices with everyday physical world and environment. A Context Provisioning Framework has been designed as a subsystem to support multifaceted context-aware services and applications. New context types can be added gradually during runtime. Moreover, components can be implemented for and deployed over heterogeneous networks and devices. Specifically, service scenarios of mall shopping, guided commuting to work, and social party participation have been demonstrated successfully [22]. The prototype Context Provisioning Framework is based on a central broker model but improves upon existing architectures by providing a context caching mechanism. In this paper, we analyse the performance gains achieved by enabling the caching of context at the broker.

Common caches used in computing systems are CPU, disk and web caches. Caches utilise the principal of locality which dictates that an object is influenced directly only by those in its immediate surroundings. In a temporal sense, it means that some resources will be referenced repeatedly within small time

P. Lukowitz, G. Kortuem, and K. Kunze (Eds.): EuroSSC 2010, LNCS 6446, pp. 108–121, 2010.

duration. In spatial sense, it means that data elements will be referenced within relatively close storage locations. Disk and memory caches utilise spatial and temporal locality by fetching a complete block (or more) around a particular disk address when data at that address is read. Temporal locality is widely exploited in web caches by storing static, not-changing resources in caches, e.g. images and Cascading Style Sheets (CSS) files. Content within web pages is commonly tagged with temporal information about the duration of its validity. The use of caching mechanism in HTTP based web communication is well established and is characterized by its inclusion in the HTTP/1.1 specification (see [10], section 13).

Context data also has some properties that make it an ideal caching candidate. Context is any information that can be used to characterise the situation of an entity [7]. An entity's situation usually includes the state of its environment, its location and other physical and conceptual properties. The state and properties change with time and hence make context information temporal in nature, i.e. its validity and hence usefulness is time dependant. Though context of all entities changes with time, it does remain valid for a particular duration. The validity duration is dependent on the type of context. For example the users' location may remain valid from a few seconds to a few hours depending on the activity. The temperature of a certain location may be valid for a few hours. The users' shopping preferences may remain valid for a few weeks, etc. This validity property of context information can be attached to the context data as metadata and exploited in a context-aware system. Moreover, access to context data by context consuming applications also demonstrates temporal locality similar to requests to web servers. We have taken advantage of these properties in our context provisioning architecture.

An implementation of our architecture has been developed and deployed but due to the limited number of real users available for testing, it is currently impossible to carry out high-load performance testing with the actual deployment. We have in parallel developed a simulation model that is used for testing and validation of the design based on Discrete Event Simulation. Our study on the effects of context caching is performed using this simulation model. After discussing related work in this area in Section 2, we describe our architecture in Section 3. The simulation model, experiment and results are described in Section 4. Concluding remarks and future directions are presented in Section 5.

2 Related Work

A number of centralised server based context provisioning systems have been developed, e.g. CoBrA [6], SOCAM [13], CASS [9], JCAF [2], PACE [14], Akogrimo [23] and MobiLife [15] employing a central server/broker to provide context from context producing components to context consuming components. Caching context has not been targeted in most of these systems explicitly. MobiLife made some progress in this regard by caching context at the context provider components. This approach gives rise to distributed context caches at each context provider with the potential to save computational load at the providers but does not reduce the communication cost. The query from the context consumer has to traverse the complete round trip from the context provider via the context broker. This mechanism can be improved by building a collective cache based on the smaller caches at context provider level.

Buchholz et al. [4] discuss the importance of caching context in improving its quality. Ebling et al. [8] also highlight caching of context data as an important issue in designing context-aware services and this point is reiterated in [19]. Caching context requires that the validity of a context data instance can be specified. This can be achieved by the inclusion of temporal information in the context representation format. MobiLife is one of the few context-provisioning systems that specify a caching component at the architecture level. However, its context representation format [11] [18] contains no metadata that specifies its temporal properties. A similar system is the Context Based Service Composition (CB-SeC) [21] that employs a Cache Engine for providing context based service composition. However, the CB-SeC system does not store context information but the whole context service in the cache. A Caching Service is demonstrated in the SCaLaDE middleware architecture [3] for use with Internet data applications. The focus of this Caching Service is on providing disconnected operation to mobile devices by keeping a mobile device cache consistent with a central cache in the network. However, no performance metrics are reported regarding the gains achieved by the use of this cache.

Despite the established significance and usability of caching components in distributed systems, context aware systems have not demonstrated their use. Some researchers have highlighted the importance of caching context information but no study has reported any results on the empirical gains of employing a context cache in a context provisioning system and this deficiency has served as the main motivation of our study. Furthermore, to the authors' best knowledge this is one of the first attempts of applying Discrete Event Simulation in the domain of Context Provisioning. Interest in such simulations is expected to increase, specifically if the performance of large-scale systems is to be evaluated.

3 C-CAST Architecture

Our reference context provisioning system, Context Casting (C-CAST), consists of three core components (Fig. 1). Context producing components are titled Context Providers and components that use context are Context Consumers. A central brokering component, Context Broker, facilitates the flow of context data between these providers and consumers. The definition of these basic roles allows the Context Provisioning Framework to support evolving and emerging ubiquitous applications. New context types can be added gradually during runtime in a plug and play manner by utilising an advertisement (registration) procedure. A detailed description of the C-CAST system is provided in [25], [20] and [16]. The following paragraphs presented a brief overview.

A *Context Consumer (CxC)* is a component that asks for and uses context data, e.g. for a context-aware application. A CxC can retrieve context information by asking the Context Broker (CxB) for a particular type of context and the CxB tries to retrieve such context from the relevant Context Providers (CxP).

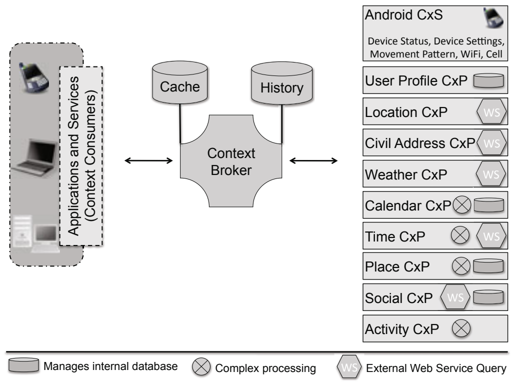


Fig. 1. C-CAST System Architecture

A *Context Provider (CxP)* is a component whose task is to provide context data of a certain type, e.g. weather, location, activity, etc. A CxP gathers data from a collection of sensors, networks, web services or other relevant sources and may apply various aggregation and reasoning mechanisms depending on the type of context it provides. A CxP provides context data in response to a specific invocation and is specialised in a particular context domain. A Context Source (CxS) is a special CxP, only offering asynchronous mode of communication. Hence a CxS directly pushes context to the CxB without being queried.

The *Context Broker (CxB)* acts as a handler and aggregator of context related communication. Primarily, the CxB controls the context flow among all attached components by allowing CxCs to query or subscribe to context information and CxPs to deliver notifications. For facilitating synchronous context queries of CxC, the CxB also provides a CxP lookup service and proxy query service by maintaining the entries of context providers registered with the broker, their communication endpoints, and their capabilities.

3.1 Context Representation

An XML based schema, ContextML [17], is used in our system to model context data, context queries, subscription/notification and some control messages as well. The following paragraphs describe its core elements.

– **Entity:** Every exchange of context data is associated with a specific entity, which can be a complex group of more than one entity. An entity is the subject of interest (e.g. a user or group of users), which context data refers

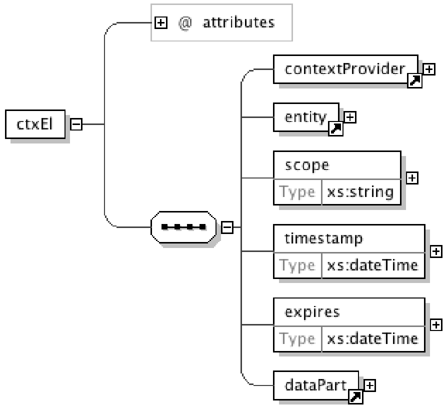


Fig. 2. Context Element part of ContextML schema

to, and it is composed of two parts: a type and an identifier. The type refers to the category of entities; exemplar entity types are *username* (for human users), *imei* (for mobile devices), *SIP uri* (for SIP accounts), *room* (for a sensed room) and *group* (for groups of other entities e.g. usernames or IMEI numbers). The entity identifier specifies a particular item in a set of entities belonging to the same type. Every human user of C-CAST system could be related to many entities in addition to the obvious type *username*, therefore a component that provides identity resolution is necessary. In our system a user profile context provider performs identity resolution for the broker.

– **Scope:** Specific context data in ContxtML is defined as scope (type of context) and comprises a set of related context parameters, for example the x, y, and z parameter of a position [17]. Every context parameter has a name and belongs to only one scope. Using scope as context exchange unit is useful because all parameters in that scope are requested, updated, provided and stored at the same time; data creation and update within a scope are always atomic is always consistent.

Whenever a context consumer requests or subscribes to a context *scope*, it receives a response encoded in the ContextML element 'ctxEl' when context is available. *ctxEl* contains information about where the context has been detected and encoded (*contextProvider*), which entity it is related to (*entity*), what scope it belongs to, and the actual context data in the *dataPart* element (Fig. 2). The elements *par*, *parS* and *parA* are simple constructs to store name-value pairs and attributed collections (arrays and structs), respectively. Every context instance that is exchanged is tagged with a specific timestamp and an expiry time. The timestamp and expiry tags specify the validity period of the scope after which context information is considered invalid.

3.2 Context Caching

The context broker has a caching component that saves retrieved context data at the time of responding to a query. The context data remains in the cache for the validity period unless it is replaced by more recent context of the same scope or the entity has to be removed to free the cache due to cache size limits. Similar broker architectures, e.g. [6], do not employ such a caching mechanism. Two cache replacement policies have been implemented along with a trivial policy of having an unlimited cache. Though this trivial policy is impractical, it provides us with a measure of the maximum cacheable context in the system at any given time and the pattern of cache growth/shrinkage. The first practical cache replacement policy is to remove the 'oldest context first', i.e. the context data stored prior to other. The second policy is to remove the 'least used context' from the cache. Only one policy is used at a time.

From an architectural point of view the central cache also serves as essential sink for asynchronous CxS components directly pushing their context. This is especially useful for mobile devices and smartphones suffering from temporary connectivity loss. Mobility is implicitly supported and the well-known challenge of pushing data directly to mobile devices (often problematic due to firewalls, Network Address Translation and port restrictions) is avoided.

3.3 Context Querying

CxC entities query context about an entity over HTTP from a RESTful interface exposed by the broker. Queries are specified in ContextML and contain scope, entity Id, entity type and constraints over certain parameters of scope, e.g. 'getContext?scope=weather&entity=username|saad'. The Context Broker tries to satisfy the query by searching the cache, providing context originating from context sources, or forwarding it to a context provider that can provide context of the scope specified in the query.

Some types of scopes are dependent on other scopes and this dependency graph is known to the context broker. For example, the weather context of a user is dependent on his location. Proxy query is a mechanism in which the broker queries all required providers consecutively on behalf of a consumer when the consumer has requested a context type which is dependent on some other scopes. Hence, if a context consumer requests a user's weather context, it does not have to query the location context first; it simply queries the weather context and the broker queries a location provider on behalf of the context consumer, gets the user's location and then queries the weather provider about the weather of that location.

4 Performance Evaluation

In addition to the development and real-world deployment of the C-CAST system, a simulation model has been developed to evaluate the system under various conditions. The simulator based on OMNET++, a Discrete Event Simulator toolkit, employs a model of the actual C-CAST context management system, including the context, components and communication model. The following paragraphs discuss this simulation model and the experiment performed to analyse the effect of employing a context cache component in the broker.

4.1 Simulation Model

The OMNET++ simulation model consists of a context broker module, context sources, context providers and context consumers connected by communication channels. The simulator comprises the core functionalities of Context Caching, Proxy Context Query Service, Provider Registration and Lookup Service. Furthermore, all occurring types of ContextML messages are modelled. CxS modules produce context data about pre-configured entities and send it to the CxB in regular intervals. CxPs are configured to provide context on invocation by the CxB and provide context about one particular scope only.

The simulation model comprises various input parameters which can be set individually for each simulation run allowing several scenarios to be evaluated and compared against each other. The parameters for each scope contain:

- *Scope ID*: For simplicity and allowing probability distributions to be applied we use numerical identifiers $(0, 1, ..., maxScopeID)$ for each scope instead of human readable names.
- *Scope Dependencies*: List of ScopeIDs which are required as input
- *Context Validity [s]*: The validity of a context instance.
- *ContextMLsizeMean [Bytes]*: The amount of data required to represent a context instance (including protocol overhead).

The parameters for each Context Provider comprise:

- *Context Provider ID*: For simplicity and allowing probability distributions to be applied we use unique numerical identifiers $(0, 1, ..., maxProviderID)$ for each CxP instead of Strings.
- *Scope ID*: see above; serves for scope and provider associations here.
- *Average Provisioning Time [ms]*:The average time delta between synchronous provider invocation and context provisioning. In reality this value depends on database access, sensor reading time, external web queries and applied context processing, respectively.

Each Context Source is described by:

- *Update Interval [s]*: The interval between two context updates, usually corresponding to the context validity time.
- *Scope ID*: The scope of the generated context instances

The Context Broker module takes the following parameters into account:

- *Lookup Time [ms]*: Average time required for Context Provider lookup
- *Cache Access Time [ms]*: Average time required for finding context in cache
- *Caching Enabled*: A boolean value for enabling/disabling cache
- *Maximum Cache Size*: Number of context instances allowed in cache
- *Cache Strategy*: Cache replacement policy (Oldest first or Least Used first)

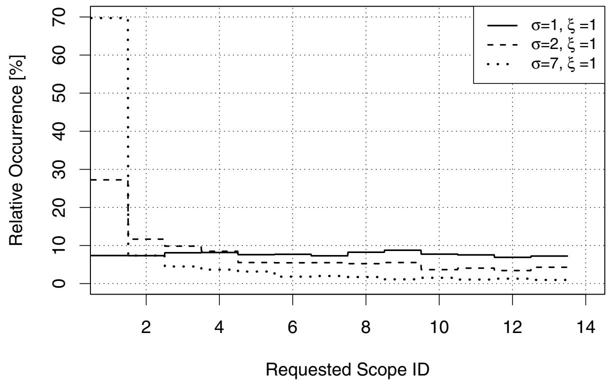


Fig. 3. Distribution of scope IDs in simulation of context queries

A Context Consumer is described by the following set of attributes:

– *Context Query Rate [/s]*: The number of context requests per second. An exponential distribution is applied with a selectable scale λ.
– *Requested Scope*: The requested scope of interest. A Pareto distribution is applied with a selectable shape σ and scale ξ (cp. Eq. 1 and Fig. 3).

$$scopeID = \left\lceil maxScopeID \cdot \left(\frac{\xi}{randomUniform\,(0,1]} \right)^{-\sigma} \right\rceil \quad (1)$$

The discretised Pareto distribution has been selected because in a real world scenario only a few primitive scopes are requested by applications very often (e.g. location, userProfile) and high-level scopes are required rarely (e.g. mood). This pattern has been reinforced in actual trials of the C-CAST system. 13 different scopes are used in this experiment and their distribution in context queries is shown in Fig. 3 with various Pareto shapes σ and scale ξ=1.

The CxC modules query the CxB at a rate that is defined by an exponential distribution. The process of the arrival of new request can be seen as a Poisson process in which events (queries) occur continuously and independently of each other with an average inter-arrival time of λ. This process is analogous to page view requests to websites, a well-modelled Poisson process [12]. The exponential distribution defines the time between events (queries in this case) in the Poisson process and allows us to observe the effect of the rate of queries on cache performance. To acquire realistic results, our testbed deployment has been measured and the parameters have been set accordingly. Table 1 summarises selected scope and CxP/CxS parameters used in the simulation.

The output parameters collected during each run are:

– *Overall Context Traffic [Bytes]*
– *Average Context Query Response Time [ms]*

Table 1. Simulation Parameters

Scope ID	CxP ID/ CxS ID	Real Scope	Processing /Update Time [ms]	Scope Dependencies	Average size of Context [Bytes]	Validity
1	CxP#1	userProfile	100		1946	60
2	CxP#2	calendar	120		40000	300
3	CxP#3	location	180	9,13	750	300
4	CxP#4	civilAddress	180	3	874	300
5	CxP#5	place	280	3	3515	300
6	CxP#6	time	90	3	1105	60
7	CxP#7	activity	350	1,5,6	913	60
8	CxP#8	weather	180	4	1817	900
9	CxS#1	cell	600		832	600
10	CxS#2	devStatus	600		1079	600
11	CxS#3	devSettings	600		883	600
12	CxS#4	movement	30		812	30
13	CxS#5	wifi	60		2157	60

– *Average Age of Context Age [s]*: The average delta between generation of a context instance and context response, i.e. how long a context instance has existed before it was finally received at the querying consumer

4.2 Simulation Results

In each simulation run 10,000 context requests distributed across 100 entities are instantiated. After all the replies have been received by consumers, the simulation is terminated. Scopes and CxPs are initialised using the values from Table 1. The CxB cache access time and provider lookup time are assumed to be 10ms. As main parameters, the CxC request rate (λ) and the shape of Pareto distribution for selecting the requested scopes (σ) are varied. Hence, the benefit of the selected cache policies is investigated with varying context request behaviour. If Pareto shape parameter σ is varied, we set the request rate parameter λ=20s^{-1}. If λ is varied, we assume a constant σ=2 and ξ=1.

Fig. 4 and Fig. 5 show the effect on the average query response delay and reveal significant benefit of caching, regardless of the policy and the size applied. This influence increases the faster requests are sent (due to the limited context expiry time) and the stronger the Pareto behaviour (long tail) is, i.e. if only a few scopes are favoured. Naturally, the age of the context instance received in the response decreases with more frequent requests. However, consider the influence of the scope distribution shape in Fig. 6. The larger the interest in only a few scopes, the fresher is the context. This appears contrary to the normal expectation that a long tailed scope distribution, which favours one scope very highly over others, should result in higher average context age because the context related to the favoured context scope faces little competition for a space in the cache.

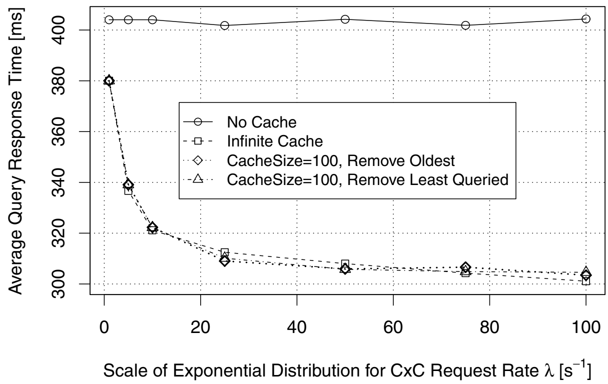


Fig. 4. Query Response Time depending on Request Rate

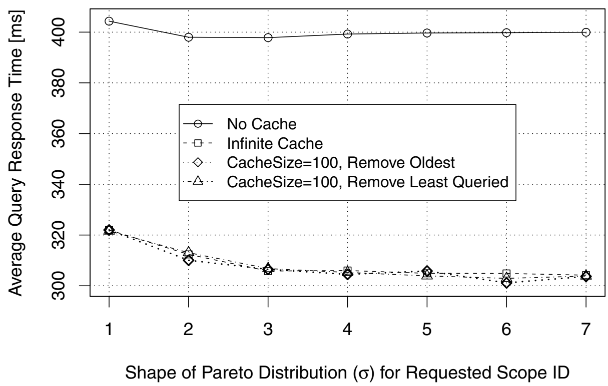


Fig. 5. Query Response Time depending on Scope Pareto Distribution Shape

Average context age should logically increase as more context is served from cache. But in our setup, the favoured scope (forming the long tail) has very short validity period and therefore even if it is served from the cache, it fails to influence the average context age considerably. The resultant decrease in average context age in Fig. 6 is due to the fact that other context scopes are also queried, though less frequently. Since they are queried infrequently, they are almost always retrieved from CxPs instead of from the cache. Hence the resultant average context age is reduced due to the freshness of less frequently queried context. As compared to the scenarios where no cache is used, caching always results in higher average context age and this fact can be interpreted as a trade-off of caching. Higher average context age is not a detrimental factor (and vice versa) in overall performance of the system because as long as the context data is within its validity period, its utility remains the same. However, the knowledge of average context age (time spent in existence before a context datum is accessed and reaches a consumer) of each type of context can aid in identifying access patterns by the consumers and tuning cache replacement policies.

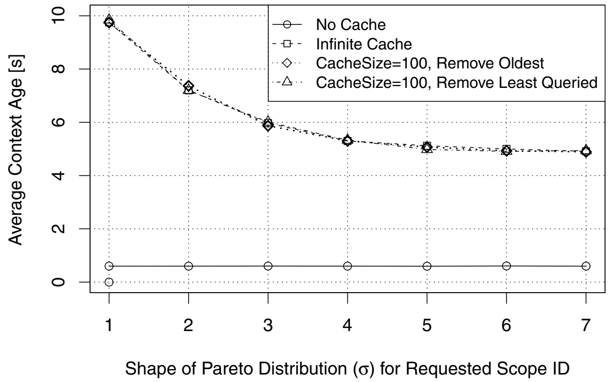


Fig. 6. Context Age depending on Scope Pareto Shape

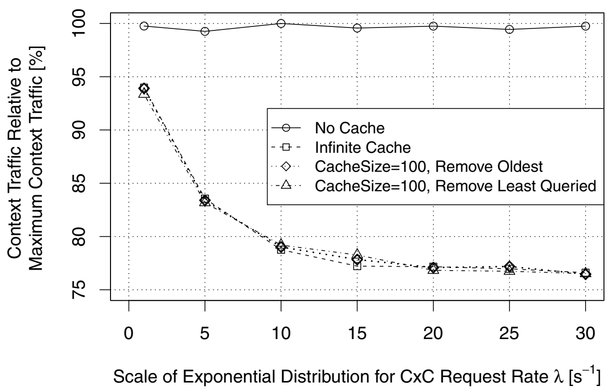


Fig. 7. Context Traffic depending on Request Rate

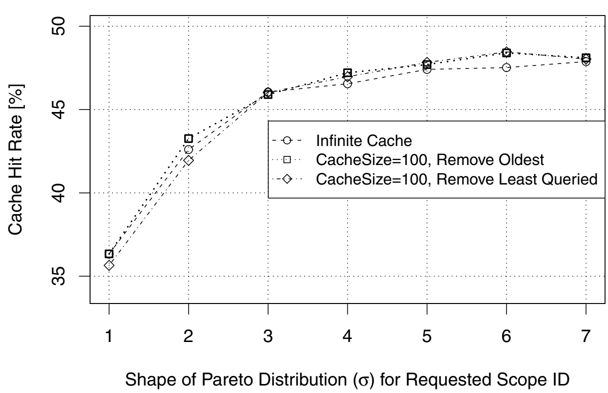


Fig. 8. Context Hit Rate depending on Scope Pareto Shape

Clearly, the overall context traffic is reduced if caching is applied (Fig. 7). More frequent requests result in higher cache hits because validity of context data remains the same. The influence of the scope distribution on the cache hit rate is shown in Fig. 8. Note that for higher values of σ, a higher hit rate is a consequence of most of the requests being focussed around one or two scopes.

Overall, our performance analysis proves that caching can aid in accelerating a large Context Provisioning Framework. The selected policies for removing entries from cache achieve similar results. This similarity can be attributed to the fact that all context data is dynamic and has limited validity. Therefore the cache does not grow beyond a certain limit under reasonable context query rates, number of entities and scope distributions. It is expected that the differences in effects of cache replacement policies may become more pronounced when the number of the entities and context scopes increase considerably.

5 Conclusion and Future Work

In this paper we have presented the C-CAST Context Provisioning Framework which is able to support emerging and evolving ubiquitous applications and services by offering multifaceted types of context. The framework is based on a modular design and an XML-based context representation schema also used for encoding brokering messages (e.g. CxP registration). Components are invoked by RESTful HTTP. The key element is a central broker which facilitates communication between heterogeneous producers and consumers of contextual information. In addition to the development of a prototype test-bed implementation, we have used Discrete Event Simulation for estimating the system performance in a larger scale scenario. Input parameters for the simulator have been based on measurement results taken from deployment and testing real world components.

The simulation model has been introduced and assumptions argued. We demonstrated that caching can significantly speed up context provisioning and reduce the context traffic. We can also conclude that the simulation and in particular Discrete Event Simulation is a powerful means for estimating the behaviour of context-aware systems when real users and applications are not available in large numbers. We observed that context distribution, particularly related to communication paradigms, has not received much attention in the community if compared to other areas (e.g. reasoning, security, privacy, context modelling, HCI design). Even less focus was put on simulating large-scale systems. Relevant efforts in this area are often limited in their evaluation of demonstrating selected scenarios in a prototype with a small number of users and devices. Ubiquitous computing is becoming a reality these days and both everyday smartphones and web services can provide rich context information. At the same time, Google Market and Apple AppStore show the trend of rapid user driven application development. It can be safely estimated that not only the number of context sources but also of context sinks will grow tremendously. It is essential to take such deliberations into account and target more sophisticated performance analysis on large context distribution and acquisition systems. Hence, we encourage further studies in this direction with the aid of simulations.

Designing a simulation model usually comes along with the challenges of finding the right abstraction level and identifying reasonable real world assumptions. This challenge heats up with the OSI level and sub-layered systems (e.g. operating system, JavaEE application server). Since, even after two decades of active research, context-aware applications appear rarely in the real-world, currently it is not a trivial task to estimate user and application behaviour correctly. Besides providing finer granulated simulation models, we will attempt to determine more realistic situation parameters. In addition, our model will be extended to consider publish/subscribe context querying and distributed caching.

References

1. Baldauf, M., Dustdar, S., Rosenberg, F.: A survey on context-aware systems. Int. J. Ad Hoc Ubiquitous Comput. 2(4), 263–277 (2007)
2. Bardram, J.E.: The java context awareness framework (jcaf) - a service infrastructure and programming framework for context-aware applications. In: Gellersen, H.-W., Want, R., Schmidt, A. (eds.) PERVASIVE 2005. LNCS, vol. 3468, pp. 98–115. Springer, Heidelberg (2005)
3. Bellavista, P., Corradi, A., Montanari, R., Stefanelli, C.: A mobile computing middleware for location-and context-aware internet data services. ACM Transactions on Internet Technology (TOIT) 6(4), 380 (2006)
4. Buchholz, T., Küpper, A., Schiffers, M.: Quality of context: What it is and why we need it. In: Workshop of the HP OpenView University Association (2003)
5. Chen, G., Kotz, D.: A survey of context-aware mobile computing research. Tech. rep., Hanover, NH, USA (2000)
6. Chen, H., Finin, T., Joshi, A.: An intelligent broker for context-aware systems. In: Proceedings of Adjunct Ubicomp 2003, pp. 183–184 (October 2003) (poster paper)
7. Dey, A.K.: Understanding and using context. Personal and Ubiquitous Computing 5, 4–7 (2001)
8. Ebling, M., Hunt, G., Lei, H.: Issues for context services for pervasive computing. In: Workshop on Middleware for Mobile Computing, Heidelberg (2001)
9. Fahy, P., Clarke, S.: Cass-middleware for mobile context-aware applications. In: Workshop on Context Awareness, MobiSys, pp. 304–308 (2004)
10. Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., Berners-Lee, T.: Rfc 2616: Hypertext transfer protocol – http/1.1, http://tools.ietf.org/html/rfc2616
11. Floreen, P., Przybilski, M., Nurmi, P., Koolwaaij, J., Tarlano, A., Wagner, M., Luther, M., Bataille, F., Boussard, M., Mrohs, B., et al.: Towards a context management framework for MobiLife. In: 14th IST Mobile & Wireless Summit (2005)
12. van Gelder, P., Beijer, G., Berger, M.: Statistical analysis of pageviews on web sites, Information and Communication Technologies, vol. 28. Witpress, Southampton (2002)
13. Gu, T., Pung, H.K., Zhang, D.Q.: A service-oriented middleware for building context-aware services. J. Netw. Comput. Appl. 28, 1–18 (2005), http://portal.acm.org/citation.cfm?id=1053030.1053031
14. Henricksen, K., Indulska, J., McFadden, T., Balasubramaniam, S.: Middleware for distributed context-aware systems. In: Meersman, R., Tari, Z. (eds.) OTM 2005. LNCS, vol. 3760, pp. 846–863. Springer, Heidelberg (2005)

15. Kernchen, R., Bonnefoy, D., Battestini, A., Mrohs, B., Wagner, M., Klemettinen, M.: Context-awareness in mobilife. In: Proc. of the 15th IST Mobile Summit. IST Mobile Summit, Mykonos, Greece (2006)
16. Knappmeyer, M., Tönjes, R., Baker, N.: Modular and extendible context provisioning for evolving mobile applications and services. In: 18th ICT Mobile Summit (2009)
17. Knappmeyer, M., Kiani, S.L., Frá, C., Moltchanov, B., Baker, N.: A light-weight context representation and context management schema. In: Proceedings of IEEE International Symposium on Wireless Pervasive Computing (May 2010)
18. Koolwaaij, J.: MobiLife Context Representation Framework. MobiLife, http://www.lab.telin.nl/~koolwaaij/showcase/crf/crf.html
19. Lei, H., Sow, D., Davis, I., John, S., Banavar, G., Ebling, M.: The design and applications of a context service. ACM SIGMOBILE Mobile Computing and Communications Review 6(4), 55 (2002)
20. Moltchanov, B., Knappmeyer, M., Licciardi, C., Baker, N.: Context-Aware Content Sharing and Casting. In: ICIN (2008)
21. Mostéfaoui, S., Tafat-Bouzid, A., Hirsbrunner, B., et al.: Sing context information for service discovery and composition. In: 5th Intl. Conference on Information Integration and Web-based Applications and Services, iiWAS, vol. 3, pp. 15–17
22. Mota, T., Gomes, D.: C-cast deliverable d11: Specification of test scenario and selected experimental environment. Tech. rep. (April 2009), http://www.ict-ccast.eu/files/C-Cast_D11.pdf
23. Osland, P.O., Viken, B., Solsvik, F., Nygreen, G., Wedvik, J., Myklbust, S.E.: Enabling context-aware applications. In: Proceedings of ICIN 2006 Convergence in Services, Media and Networks (June 2006)
24. Truong, H., Dustdar, S.: A survey on context-aware web service systems. International Journal of Web Information Systems 5(1), 5–31 (2009)
25. Zafar, M., Baker, N., Moltchanov, B., Goncalves, J.M., Kiani, S.L., Knappmeyer, M.: Context management architecture for future internet services. In: ICT Mobile Summit 2009, Santander, Spain (June 2009)

A Logic Based Context Query Language

Ling Feng[1], Junhui Deng[1], Zhanjiang Song[2], and Wenwei Xue[2]

[1] Department of Computer Science & Technology, Tsinghua University, Beijing, China
{fengling,deng}@tsinghua.edu.cn
[2] Nokia Research Center, Beijing, China
{zhanjiang.song,wayne.xue}@nokia.com

Abstract. Context is a basic ingredient in pervasive computing. It refers to user's both external environment and internal status, implying user's information needs during the interaction between the user and computer systems. By context-awareness, a variety of context information from heterogeneous sources is sensed, inferred, and exploited to facilitate information systems to better understand users' needs and deliver the right context-aware service to the right user at the right time and place. Fundamental to context-awareness is the context-aware computing infrastructure support. Among its components, context processing engine is the core of such an infrastructure, whose power is reflected from its context query language. In this paper, we first overview typical context query languages developed in the literature, and then present the design of our temporal description logic based context query language. We implement such a query language on our context-aware mobile services development platform. To the best of our knowledge, this is the first attempt on a temporal description logic based language for context querying in the literature. As the provided temporal relevant predicates can accommodate both time interval and time point semantics, such a language is flexible and powerful enough to represent and query complex context information in time-critical context-aware applications.

Keywords: context, context query language, Temporal Description Logic.

1 Introduction

In the pervasive computing era, the usability of a system supplied service to its ambient users is not an absolute value intrinsic to the underlying information systems, but rather a factor that can vary depending on the goal and user, in other words, the context of use. In fact, context can serve as hints on which service to offer based on the past and current usage context, since it carries a kind of semantics related to when, where, and how to use that service. By means of context-awareness, service providers can better understand users' real needs. Thanks to the rapid development of sensing and pervasive computing technologies, a lot of context information is able to be sensed, inferred, and exploited to enhance services' usability. Here, to detect context for context-aware information services delivery, a powerful context query language is necessary for representing and acquiring diverse context from heterogeneous resources, where context reasoning element shall also be incorporated in context querying [4].

P. Lukowitz, G. Kortuem, and K. Kunze (Eds.): EuroSSC 2010, LNCS 6446, pp. 122–134, 2010.

In this paper, after a brief overview of typical context query languages developed in the literature, we present our novel context query language. It is based on the formal temporal description logic extended with concrete domains [1, 2]. This language serves as the interface for application developers to interact with the context query engine, which has been implemented in our mobile context-aware services development platform.

The reason for us to turn to a logic based context query language is to seek consistency with our description logic-based context modeling method [18]. It is our hope to set up a solid theoretical foundation for context querying and optimization, rather than processing context by putting together some query operators, unaware of logic soundness and completeness. To the best of our knowledge, this is the first work on a DL-based language particular used for context querying. The presented language can naturally facilitate context inference, as well as complex context representation and querying through its time interval and time point based temporal predicates.

The reminder of the paper is organized as follows. We overview four typical kinds of context query languages in the literature in Section 2, and present our temporal description logic based solution in Section 3. The use of the language in a context-aware mobile services development platform is demonstrated in Section 4. We conclude the paper and point out further work in Section 5.

2 Typical Context Query Languages

There exist four typical kinds of context query languages in the literature, which are *SQL-based, XML-based, ontology-based,* and *API*-based, respectively. A more comprehensive discussion of the existing context query languages can be found in good survey papers [8, 12].

2.1 SQL-Based Context Query Language

SQL (Structured Query Language) is the standard database query language, which has formed the backbone of most modern database systems. Formulating context queries in an SQL-like language is a straightforward solution for context querying. Such kind of context queries typically views context information as traditional database tables with auxiliary context-associated attributes like accuracy, confidence, update time, sample interval, and so on. For example, [13] takes both dynamic sensor readings and static user profiles as normal database tables, and poses queries directly upon sensors. It phrases a query for the average people's car speed within period *w* at the “WangFu” road as follows:

```
Select     avg(speed, w)
From       sensorReadings as s
Where      s.segment∈{"WangFu"}
```

[11] considers four types of context entities (i.e., device, access point, people, and space), as well as relationships among context entities, and uses them to create a

relational database model. For example, it queries the Alice's location context as follows:

```
Select      Location
From        PersonLocation
Where       PersonID = Alice's UID
Require     location.updateTime  within  2  minutes  of
            present time
            location.accuracy within 500 meters of actual
            location
TimeLimit   1 minute
```

The advantage of extending SQL for context querying is that both users and system developers are already familiar with SQL, and thus incurring little training time to write context queries.

2.2 XML-Based Context Query Language

Apart from SQL, XML (eXtensible Markup Language) has also been employed to represent context queries and query results [9, 10, 17]. For example, a context query for the location of a user named "John Leung" can be stated in the following XML fragment [4]:

```
<factory_node id="getUserLocation" class="location">
  <entity class="user">
     <identity class = "name"> John Leung </identity>
  </entity>
</factory_node>
```

where Factory-Node is the core node to represent queries, and the execution of this query will return the following context information:

```
<context_information class="location">
 <entity class="user">
   <identity class="name"> John Leung </identity>
 </entity>
</context_information>
```

2.3 Ontology-Based Context Query Language

[11, 6, 7] use existing ontology languages such as RDF, DAML+OIL, and OWL to query context. For instance, the context information "*John's location is bathroom*" can be easily represented in a RDF syntax: *Location* (*John*, *bathroom*), and label x in (*locatedIn John ?x*) can be used for querying *John's location*. [16] adopts a 5-ary tuple (O, σ, θ, $\sum$, τ) to express a context query, where O is a set of used ontologies, σ is a selection list, θ is a filtering statement, $\sum$ is the cardinality, and τ is temporal constraints. The benefit of ontology-based query languages lies in its reasoning capability which can facilitate the derivation of high-level context information from low-level raw context data.

2.4 API-Based Context Query Language

Some researchers design specific APIs or even new programming languages for context querying and processing. For example, [14] provides APIs in Java in their context management tools. [5] proposes a programming language called *iQL* to define context related data types and compose context data into high-level one of complex types. In *iQL*, context data sources can include active context sources and passive context sources. Passive context sources supply context data only upon request. Active context sources get an initiative signal and then push the context data in a stream to the subscriber. Hybrid data sources are those that not only supply context data upon request, but also push context data stream.

3 A Temporal Description Logic Based Context Query Language

To accommodate temporal and spatial characteristics of context, as well as the intelligence implied by context-awareness, we exploit the well-developed temporal description logic extended with concrete domains ALCF(D) [6], where features like time and place can be uniformly treated as concrete domains. Before presenting our design of context query language, let's first review the Temporal Description Logic ALCF(D), which serves as the logic foundation for our context query language and further context querying.

3.1 A Brief Review of Temporal Description Logic

Our previous good experiences with Description Logics (DL) in context-aware computing [18] lead us to turn to DL again for more complex context querying. DL [3] is a family of knowledge representation languages tailored for expressing knowledge about concepts and concept hierarchies, and are considered the most important knowledge representation formalism, unifying and giving a logical basis to the well-known traditions of frame-based systems, semantic networks, and KL-ONE-like languages, object-oriented representations, semantic data models, and type systems, etc. They can be used for representing the ontology of a domain, and thus form the basis of such ontological languages as OWL. Many reasoning and editing tools currently exist for easily and efficiently dealing with DL knowledge bases. As known, a DL knowledge base contains two components, i.e., the TBox and the ABox. The TBox introduces the terminology, i.e., the vocabulary of an application domain, while the ABox contains assertions about named individuals in terms of this vocabulary. The vocabulary consists of *concepts* representing sets of individuals, and *roles* representing binary relationships between individuals. In addition to atomic concepts and roles, DL allows to build complex descriptions of concepts and roles using such operators as *intersection* ($\sqcup$), *union* ($\sqcap$), and *complement* ($\neg$). Besides, roles can have full ($\forall$) and existential ($\exists$) quantifications. In the Temporal Description Logic ALCF(D) with concrete domains, a concrete domain D is a pair ($\triangle$D,ΦD), where $\triangle$D is a set called the domain, and ΦD is a set of predicate names. Each predicate name *P* in ΦD is associated with *n* parameters [1, 2, 15].

3.2 A Description Logic Based Context Query Language

The design of our context query language is in line with the temporal description logic with concrete domain ALCF(D) with an easy mapping. A context query *Q* typically involves some predicate symbols connected via boolean operator $\wedge$ (AND), $\vee$ (OR), or $\neg$ (NOT). Each predicate symbol is associated with some arguments, denoted as $p(a_1, a_2, \ldots, a_n)$ ($n \geq 1$). Different predicate symbols can have different numbers of arguments, and we call the number of arguments *n* of predicate *p* the *arity* of *p*. The arguments can be either constants or variable with name identifiers. The meaning (or answer) of query *Q* is to deduce the different constant combinations that, when bound (assigned) to the variables, can make all the involved predicates true. Simply, a context query condition is a conjunction of disjunctions of query predicates of the form: $(x_1, \ldots, x_s) \leftarrow (q_{1,1} \vee \ldots \vee q_{1,n1}) \wedge \ldots \wedge (q_{m,1} \vee \ldots \vee q_{m,nm})$, where $q_{i,j}$ can be either positive or negative predicate literal, and $x_1, \ldots, x_s$ are the query output.

I. *Basic Notations*

The basic constituents of the query language are *identifiers* made up of alphabetic, digital, and other visible characters, and *data types* such as string, number (`int`, `float`, `double`), and time. *Variables* and *constants* are supported. An Extended Backus–Naur Form (EBNF) syntax specification is below.

```
identifier = alphabetic character ,
             { alphabetic character | digit } ;
alphabetic character =
   "A"|"B"|"C"|"D"|"E"|"F"|"G"|"H"|"I"|"J"|"K"|"L"|"M"|
   "N"|"O"|"P"|"Q"|"R"|"S"|"T"|"U"|"V"|"W"|"X"|"Y"|"Z";
digit = "0"|"1"|"2"|"3"|"4"|"5"|"6"|"7"|"8"|"9" ;
variable = identifier;
constant = number | string | time-constant |
           due-time-constant;
time-constant = time-point-constant |
time-interval-constant | dur-time-constant;
time-point-constant = clock-time-constant;
time-interval-constant =
           [clock-time-constant, clock-time-constant] |
           (clock-time-constant, clock-time-constant] |
           [clock-time-constant, clock-time-constant)|
           (clock-time-constant, clock-time-constant);
dur-time-constant = number, time-granularity;
time-granularity = "Second"|"Minute"|"Hour"|
                   "Day"|"Month"|"Year";
time-format = time-point-format | time-interval-format;
time-point-format = clock-time-format;
time-interval-format =
           [Clock-Time-Format, Clock-Time-Format] |
           (Clock-Time-Format, Clock-Time-Format] |
           [Clock-Time-Format,Clock-Time-Format) |
```

```
            (Clock-Time-Format, Clock-Time-Format)
 clock-time-format =
           [Year::Month::Day::] Hour::Minute::Second;
clock-time-constant = [number:: number:: number::]
                      number:: number:: number
```

II. *DL Concepts and Roles*

DL concepts and roles are meaningful entities, and each of them may contain one or more attributes.

For example, the following statement

```
DEFINE DL CONCEPT
      GPS-Stream(personID: int, timestamp: Clock-Time-Format,
                 x-val: float, y-val: float);
```

defines a DL concept `GPS-Stream` with four attributes `(personID, timestamp, x-val, y-val)`, where `(x-val, y-val)` signifies the coordinate location of a person of ID `personID` at `timestamp` sensed by a GPS device.

The following statement

```
DEFINE DL ROLE  Work(personID: int, employer: string);
```

defines a DL role `Work` with two attributes `(personID, employer)`, representing person of ID `personID` works for a certain employer. An EBNF specification of DL concepts and roles is as follows.

```
DL-concept-define = "DEFINE DL CONCEPT",
                   DL-concept-name, DL-concept-structure;
DL-concept-name = identifier;
DL-concept-structure = DL-concept-member-element,
                        {"," DL-concept-member-element};
DL-concept-member-element =
                 DL-concept-element-name, ":", data-type;
DL-concept-element-name = identifier;
data-type = "int" | "float" | "double" | "string";
DL-role-define = "DEFINE DL ROLE",
      "(", DL-concept-name, ",", DL-concept-name, ")";
```

III. *Predicates*

Predicates are used to declare context query conditions, and can be categorized into three types, which are 1) *DL concept/role associated predicates,* 2) *system-built-in predicates,* and 3) *user-defined predicates*.

1) A DL *concept* can be designated by an unary predicate, while a DL *role* can be designated by a binary predicate. In other words, the logic term C(x) is true if variable x is an instance of the DL concept C. Similarly, R(x, y) is true if variable x and y are two

concept instances observing the DL role relation R. For example, assume that `e` is an instance of the DL concept `GPS-Stream(personID, timestamp, x-val, y-val)`, we have `GPS-Stream(personID, timestamp, x-val, y-val) (e) = TRUE`. The instance of the DL role `Work(e.personID, "Nokia") = TRUE`, given that the person of ID `e.personID)` works for `"Nokia"` in reality.

2) System-built-in predicates include the five classical comparison predicates, i.e., EQUAL (equal), GE (greater-than), GEQ (not-less-than), LE (less-than), and LEQ (not-greater-than), as well as time related comparison predicates, which are *before, meets, overlaps, during, starts, finishes, equal, after, met-by, overlapped-by, started-by, contains, finished-by*, developed by Allan [1, 2]. Here, two types of time measurements are supported, i.e., ***time interval*** and ***time point***. A time related function `DUR(e.timestamp)` returns the durability of an event (i.e., instance of a DL concept/role). When `DUR(e.timestamp)=0`, it indicates that `e` is a point-wise event, and when `DUR(e.timestamp)=30minute`, it indicates that `e` happens during a time interval of 30 minutes.

3) User-defined predicates are defined by users. One such predicate example is `Near(x, y, placeName)`. It returns `TRUE` if a position at coordinate `(x, y)` is near the place named `placeName`.

Aggregate functions (count, average, max, min, and sum) and other user-defined functions can also be defined in the language to enforce the operation capability against context information.

```
predicate = predicate-name, " (", arg {"," arg}, ")";
arg = variable | variable, ".", concept-element-name |
      constant | half-constant;
predicate-name = DL-concept-name | DL-role-name |
                 system-built-in-predicate-name |
                 user-defined-predicate-name;
system-built-in-predicate-name =
       comparison-predicate-name | time-predicate-name;
comparison-predicate-name = "EQ"|"GE"|"GEQ"|"LE"|"LEQ";
time-predicate-name = point-point-predicate-name |
                      interval-interval-predicate-name |
                      point-interval-predicate-name |
                      interval-point-predicate-name;
point-point-predicate-name = "EQUAL"| "BEFORE"|"AFTER"
interval-interval-predicate-name = "EQUAL" | "BEFORE"|
           "MEETS" | "OVERLAY" | "STARTS" | "DURING" |
           "FINISHES"| "FINISHED-BY" | "CONTAINS" |
           "STARTED-BY" | "OVERLAPPED-BY" | "MET-BY" |
           "AFTER";
point-interval-predicate-name = "BEFORE" | "MEETS" |
    "STARTS" | "DURING" | "FINISHES"|"MET-BY" | "AFTER";
interval-point-predicate-name = "BEFORE" | "MEETS" |
```

```
    "FINISHED-BY" | "CONTAINS" | "STARTED-BY"|"MET-BY" |
    "AFTER";

function = function-name, "(", data-type,
                     {", " data-type}")"  ["AS", url];
function-name = aggregate-function-name |
     user-defined-function-name | time-function-name;
aggregate-function-name = "COUNT" | "AVERAGE" |
             "MAX" | "MIN" | "SUM" | "FIRST" | "LAST";
time-function-name = "DUR";
```

IV. *Context Queries*

A context query declares four parts, i.e., *query condition, query output, query quality*, and *output channel*. Query condition is a conjunction of disjunctions of query predicates of the form $(q_{1,1} \vee \ldots \vee q_{1,n1}) \wedge \ldots \wedge (q_{m,1} \vee \ldots \vee q_{m,nm})$, where $q_{i,j}$ can be either positive or negative predicate defined above. Query output is a list of variables of instances or instance attributes. Output quality constrains query execution time and result uncertainty level, since most context information sensed or inferred is uncertain in nature. Query output can be delivered to users as a file or message through pipe or socket. Query output, query quality, and output channel can be omitted. When there is no query output, a Boolean TRUE or False will be returned based on the evaluation of query condition. We have the following EBNF specification for the context query language.

```
context-query = query-output , "←", query-condition [,"QoS:",
               QoS-Format] [, output-format];
query-output = "(", output-item {"," output-item}, ")";
output-item = variable |
                variable, ".", concept-element-name;
boolean-operator = ∧ | ∨ | ¬;
query-condition = " (",conjunctive-condition, " )"
                 {"∧", " (",conjunctive-condition, " )"};
conjunctive-condition  =  predicate  |  "¬",  predicate  |
                         predicate, {"∨", predicate};
QoS-Format  =  "[  exec-time",  Time-Interval,  "]"  |
                 "[uncertainty", number, "]";
output-format = "file", (filename) | "pipe" |
                "message" | "socket";
```

3.3 Context Query Examples

We give some examples to show how to use the proposed description logic based query language to query context. Let `GPS-Stream(ID, timestamp, x-val, y-val)` be a DL concept and `Work` be a DL role.

Example 1: *Who working in Nokia has been near XiDan for 1 hour?*

```
(e.ID) ← GPS-Stream(ID, timestamp, x-val, y-val)(e) ∧
          Work(e.ID, "Nokia") ∧ Near(e.x-val,e.y-val,"XiDan") ∧
          EQUAL(DUR(e.timestamp), "1 Hour");
```

Here, `e` is declared as an instance of the DL concept `GPS-Stream`. `Near` is a user-defined predicate. `EQUAL(DUR(e.timestamp), "1 Hour")` requires that `e` happen within a time interval (i.e., `1 hour`).

Example 2: *Who working in Nokia appeared near XiDan in the period of 7:00 am to 8:00 am?*

```
(e.ID) ← GPS-Stream(ID, timestamp, x-val, y-val)(e) ∧
          Work(e.ID, "Nokia") ∧ Near(e.x-val,e.y-val,"XiDan") ∧
          EQUAL(DUR(e.timestamp), 0)  ∧
          DURING (e.timestamp, [7::00::00,  8::00::00]);
```

Here, `e.timestamp` is a time point, since `DUR(e.timestamp)=0`.

Example 3: *Who goes to XiDan first and 30 minutes later goes to WangFuJing?*

```
(e1.ID) ←GPS-Stream(ID, timestamp, x-val, y-val)(e1) ∧
          GPS-Stream(ID, timestamp, x-val, y-val)(e2) ∧
          EQ(e1.ID, e2.ID) ∧
          Near(e1.x-val, e1.y-val, "XiDan") ∧
          Near(e2.x-val, e2.y-val, "WangFuJing") ∧
          EQUAL(DUR(e1.timestamp), 0) ∧
          EQUAL(DUR(e2.timestamp) , 0) ∧
          BEFORE(e1.timestamp, e2.timestamp, "30 minute");
```

Note that `BEFORE` operator is the reverse of `AFTER` operator.

Example 4: *Which shopping mall does a user go to after s/he visited XiDan?*

```
(x) ←GPS-Stream(ID, timestamp, x-val, y-val)(e1) ∧
     GPS-Stream(ID, timestamp, x-val, y-val)(e2) ∧
     EQ(e1.ID, e2.ID) ∧
     Near(e1.x-val, e1.y-val, "XiDan") ∧
     Near(e2. x-val, e2.y-val, x) ∧ ShopMall(x) ∧
     EQUAL(DUR(e1.timestamp), 0) ∧
     EQUAL(DUR(e2.timestamp), 0) ∧
     BEFORE (e1.timestamp, e2.timestamp,*);
```

Lable"*"in `Before(e1.timestamp, e2.timestamp, *)` represents any time value.

Example 5: *Who appears near XiDan, and later his/her colleague from the same working place appears near XiDan?*

```
(e1.ID) ←GPS-Stream(ID, timestamp, x-val, y-val)(e1) ∧
          GPS-Stream(ID, timestamp, x-val, y-val)(e2) ∧
```

```
           Near(e1.x-val, e1.y-val, "XiDan") ∧
           Near(e2. x-val, e2.y-val, X) ∧
          ¬Equal(e1.ID, e2.ID) ∧ Work(e1.ID, x) ∧
           Work(e2.ID, y) ∧ Equal(x,y) ∧
           EQUAL(DUR(e1.timestamp), 0) ∧
           EQUAL(DUR(e2.timestamp) , 0) ∧
           BEFORE(e1.timestamp, e2.timestamp,*);
```

Example 6: *Two persons stay closely together at the weekend. Who are they?*

```
(e1.ID,e2.ID)← GPS-Stream(ID,timestamp, x-val, y-val)(e1)∧
                GPS-Stream(ID,timestamp, x-val,y-val)(e2) ∧
               ¬EQ(e1.ID, e2.ID) ∧
                Near(e1.x-val,e1.y-val,e2.x-val,e2.y-val) ∧
               EQUAL(DUR(e1.timestamp), 0) ∧
               EQUAL(DUR(e2.timestamp) , 0) ∧
               EQUAL(e1.timestamp, e2.timestamp) ∧
               Weekend(e1.timestamp);
```

Here, `Weekend` is a user-defined function.

Example 7: *Who goes to XiDan three times in one day?*

```
(e.ID) ← GPS-Stream(ID, timestamp, x-val, y-val)(e) ∧
          Near(e.x-val, e.y-val, "XiDan") ∧
          EQUAL(DUR(e.timestamp), 0) ∧
          EQUAL(Count(e.timestamp), 3) ∧
          DURING(e.timestamp, "[0::0::0, 24:00::00]");
```

Example 8: *A user goes to XiDan first and then goes to WangFuJing. S/he does not go to the Great Wall between these two visits. Who is it?*

```
(e1.ID) ← GPS-Stream(ID, timestamp, x-val, y-val)(e1) ∧
           GPS-Stream(ID, timestamp, x-val, y-val)(e2) ∧
           EQ(e1.ID, e2.ID) ∧
           Near(e1.x-val, e1.y-val, "XiDan") ∧
           Near(e2.x-val, e2.y-val, "WangFuJing") ∧
           EQUAL(DUR(e1.timestamp), 0) ∧
           EQUAL(DUR(e2.timestamp) , 0) ∧
           BEFORE(e1.timestamp, e2.timestamp, *) ∧
           EQUAL(e1.ID, e3.ID) ∧
           ¬Near(e3.x-val, e3.y-val, "GreatWall") ∧
           ¬DURING(e3.timestamp, [e1.timestamp, e2.timestamp]);
```

4 Implementation

The above logic-based context query language has been implemented in our context-aware mobile services development platform. The platform has five major components, including context service registry, context supplier, context manager, context consumer, and context-aware service registry. The bottom context suppliers (like mobile phones, GPS, RFID devices, external databases, etc.) provide context information via the context service registry to the internal system. The middle context manager is responsible for quality-aware context acquisition and context querying (detection) at the high-performance backend. Necessary context is stored into a context database for later retrieval and analysis. Two types of context queries are supported, which are ad-hoc context queries or continuous context queries. The upper context consumer monitors context and execute context-triggered information services. Users configure and requests information services via the context-aware service registry.

A context-aware information service takes the form *Event-Condition-Action* extended with the execution *Mode* and *Time Extent*. When a certain event (context) happens (holds), and the condition is satisfied, the corresponding action will be executed. Such a checking and execution can be done either *once* only, *periodically* with a certain frequency, or *continuously* in the specified *time extent*. Via the platform interface (as shown in Figure 1), users can add their new context-aware mobile services via the interface. They can also re-use or modify existing services to tailor to their changing requirements. For ease of operation, type-ahead input suggestions are provided by the platform.

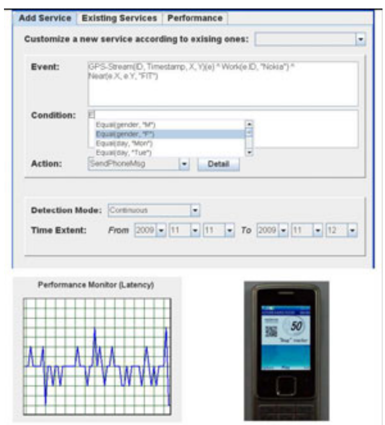


Fig. 1. Context-aware services development interface

To be scalable for heavy workload in terms of query numbers and context volumes, we adopt parallel context processing strategy by partitioning incoming context data and operators. The whole system is implemented with OSGi and C++/MPI.

Taking the service request ("*if a female Nokia employee appears at the front door of the FIT building on November 11 and 12, 2009, then send the shopping mall Doug's voucher to her mobile phone*") for example, users fill in the event, condition, action, detection mode (continuous), and time extent items, and then press the Finish button. The platform monitors the context in the period of November 11 and 12, 2009. If it detects a Nokia employee's mobile phone GPS is near the *FIT* building, the platform will check the static *employee* database. If the *female* condition is satisfied, a

SendPhoneMsg action will be initiated right away which sends an electronic voucher to this person's mobile phone. Users can also view the system response times (elapsing from the happening of a certain event/context to the detection of the event by interactively entering different event/context queries.

5 Conclusion

Context is an essential element in pervasive computing. Users' information needs can be better understood and supplied by means of context-awareness. Context data may be sensed, inferred, or directly input by users, etc., which calls for specific query mechanisms to acquire context information. In order to deliver context-aware services, we propose a temporal description logic based context query language. We are currently researching parallel query processing and optimization techniques with respect to the proposed logic-based context query language. It is also important to transform such a logic-based query language to a simple user-friendly interface language specifically for end users.

Acknowledgments. The authors would like to thank Hao Wang, Lin Qiao, Yuanping Li, and Tangjian Deng for the excellent implementation work. The work is jointly funded by Nokia Research Center, China, National Natural Science Foundation of China (60773156), Chinese National 863 Advanced Technology Program (2008AA01Z132), and Chinese Major State Basic Research Development 973 Program (2011CB302200).

References

1. Allen, J.: Maintaining Knowledge about Temporal Intervals. Communications of the ACM 26(11) (1983)
2. Artale, A., Franconi, E.: A Survey of Temporal Extensions of Description Logics. Annals of Mathematics and Artificial Intelligence 30(1), 171–210 (2000)
3. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P. (eds.): The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press, Cambridge (2003)
4. Buchholz, T., Krause, M., Linnhoff-Popien, C., Schiffers, M.: CoCo: Dynamic composition of context information. In: 1st Intl. Conf. on Mobile and Ubiquitous Systems: Networking and Services, pp. 335–343 (2004)
5. Cohen, N.H., Lei, H., Castro, P., Davis II, J.S., Purakayastha, A.: Composing Pervasive Data using iQL. In: 4th IEEE Workshop on Mobile Computing Systems and Applications, pp. 94–104 (2002)
6. Gu, T., Pung, H.K., Zhang, D.Q.: Toward an OSGi-based Infrastructure for Context-Aware Applications. IEEE Pervasive Computing 3(4), 66–74 (2004)
7. Gu, T., Pung, H.K., Zhang, D.Q.: A Middleware for Building Context-Aware Mobile Services. In: 59th IEEE Vehicular Technology Conference, pp. 2656–2660 (2004)
8. Haghighi, P.D., Zaslavsky, A.B., Krishnaswamy, S.: An Evaluation of Query Languages for Context-Aware Computing. In: Bressan, S., Küng, J., Wagner, R. (eds.) DEXA 2006. LNCS, vol. 4080, pp. 455–462. Springer, Heidelberg (2006)

9. Heer, J., Newberger, A., Beckmann, C., Hong, J.I.: Liquid: Context-aware distributed queries. In: Dey, A.K., Schmidt, A., McCarthy, J.F. (eds.) UbiComp 2003. LNCS, vol. 2864, pp. 140–148. Springer, Heidelberg (2003)
10. Hönle, N., Kappeler, U.P., Nicklas, D., Schwarz, T., Grossmann, M.: Benefits of integrating meta data into a context model. In: 2nd IEEE PerCom Workshop on Context Modeling and Reasoning at PerCom 2005, pp. 25–29 (2005)
11. Judd, G., Steenkiste, P.: Providing Contextual Information to Pervasive Computing Applications. In: 1st IEEE Intl. Conf. on Pervasive Computing and Communications, pp. 133–142 (2003)
12. Li, Y., Feng, L., Zhou, L.: Context-Aware Database Querying: Recent Progress and Challenges. In: The Book Context-Aware Mobile and Ubiquitous Computing for Enhanced Usability: Adaptive Technologies and Applications, IGI Publisher (2009)
13. Madden, S., Franklin, M.J.: Fjording the stream: An Architecture for Queries Over Streaming Sensor Data. In: 18th ICDE, pp. 555–566 (2002)
14. McFadden, T., Henricksen, K., Indulska, J.: Automating Context-Aware Application Development. In: UbiComp Workshop on Advanced Context Modelling, Reasoning and Management, pp. 90–95 (2004)
15. Lutz, C.: Reasoning with concrete domains. In: 16th IJCAI, pp. 90–95 (1999)
16. Perich, F., Joshi, A., Yesha, Y., Finin, T.: Collaborative Joins in a Pervasive Computing Environment. Intl. J. on Very Large Data Bases 14(2), 182–196 (2005)
17. Reichle, R., Wagner, M., Khan, M.U., Geihs, K.: A Context Query language for Pervasive Computing Environments. In: 6th Intl. Conf. on Pervasive Computing and Communications, pp. 434–440 (2008)
18. van Bunningen, A., Feng, L., Apers, P.: A Context-Aware Preference Model for Database Querying in an Ambient Intelligent Environment. In: Bressan, S., Küng, J., Wagner, R. (eds.) DEXA 2006. LNCS, vol. 4080, pp. 33–43. Springer, Heidelberg (2006)

Sensor Abstractions for Opportunistic Activity and Context Recognition Systems

Marc Kurz and Alois Ferscha

Institute for Pervasive Computing
Johannes Kepler University, 4040 Linz, Austria
Phone: +43(0)732/2468/8527; Fax: +43(0)732/2468/8426
{kurz,ferscha}@pervasive.jku.at

Abstract. Pervasive environments are inherently characterized to draw from sensor infrastructures in order to become situation aware. Very recent technological evolutions of sensor hardware (e.g. for geoposition, acceleration, orientation, noise, light, humidity, chemical properties, etc.) have fertilized an explosive growth of sensor infrastructures, introducing whole new challenges for sensor software architectures like heterogeneity, redundancy and replaceability, fault tolerance, mobility, massive deployment, but most of all frequency of change and spontaneous availability. In this paper we address the very fundamental issue of exploiting spontaneous sensor configurations by introducing mechanisms for sensor self-description, goal-oriented sensing missions and dynamic sensor ensemble management. The concept of "sensor abstractions" is introduced, making the use of physical as well as immaterial sensors transparent from any technical sensor properties. An opportunistic sensor software architecture has been implemented, reversing former architecture principles (e.g. fusion of all available sensors) into a spontaneous, selective, utility driven involvement of sensors based on their sensing mission contribution potentials. The framework is implemented in OSGi, and demonstrated for activity recognition missions.

1 Introduction and Motivation

Activity and context recognition systems are an interesting research field in the area of pervasive and ubiquitous computing [13]. By applying (wireless) sensor networks which deliver data of the physical world and the persons living and acting in it [23] such systems interpret the data in terms of inferring activities and more generally the context [1,5,9] of persons and subjects in real world environments.

All such activity and context recognition systems have shared one major problem so far: the sensor deployment is application-specific and thus the mapping from sensor signals to context and activities has to be known and defined at *design time* [26,27]. This results in a static and predefined sensor infrastructure and provides no flexibility in the sensor deployment, the sensors actual positions (e.g. body-worn sensors have to be placed *exactly* at predefined positions) or the sensors availability (e.g. sensors can fail due to various reasons). Due to advancements

P. Lukowitz, G. Kortuem, and K. Kunze (Eds.): EuroSSC 2010, LNCS 6446, pp. 135–148, 2010.

in electronics and wireless communication enabling technologies, the development of low-cost, low-power, small and multifunctional sensor nodes is possible [3]. As sensor systems nowadays are getting smaller and smaller and due to their capability of communicating with wireless interfaces, their mobility is increasing [7]. Due to this node-mobility on the one hand and the limited power resources resulting from the fact that the small and wireless devices are not permanently attached to a power socket [2,30] on the other hand, the infrastructure of a wireless sensor network can change permanently and cannot assure that a probably predefined sensor setup is static and stable over a certain amount of time.

Therefore we are working on the development of mobile opportunistic activity and context recognition enabling technologies. The term *opportunistic* in this context means that no predefined sensor infrastructure has to be defined at design time, the system rather makes best use (by applying a utility-driven approach [18]) of the available sensor devices according to a recognition goal, whereas when the system once started to operate, this sensor-setup is also not presumed to be fixed. The system reacts on changes in the sensor network topology at *runtime*. For example, new sensor nodes can be added to a running system, or existing nodes may be disconnected (e.g. they may run out of power). An opportunistic activity and context recognition system has to react properly at runtime to such dynamics in the sensor infrastructure. Furthermore, neither the network typology of sensor devices is fixed, nor the types of sensors that can be used within an opportunistic system are predefined, nor is the recognition goal hard-coded in the application at design time, it can be dynamically stated by users and/or applications in an abstract manner at *runtime*. Based on this high-level recognition goal, which has to be further processed and translated by the system to a coordinated machine-readable sensing mission, the available sensing devices configure themselves to so-called *ensembles* by applying self-* capabilities, like *self-description*, *self-organization* and *self-management* [12,17]. Different algorithms, methodologies and mechanisms have to be assessed on the way to a fully functioning system that enables opportunistic goal-driven activity and context recognition in the various steps of the recognition chain (like goal processing [18], formation of coordinated sensor ensembles and cooperative sensing [15,19], sensor signal acquisition, ensemble reconfiguration, data processing, feature extraction, etc.). The *OPPORTUNITY framework* that is currently being developed is (i) a prototypical implementation for evaluating and testing the approaches and (ii) a first step towards a ready-to-use middleware for building opportunistic activity and context recognition applications for different domains.

Figure 1 shows the concept and general activity and context recognition chain for an opportunistic architecture, whereat in this illustration two different goals (*"I need"-Request*) are passed on to the system. On top, the users and applications are located who formulate a recognition goal in an abstract manner that is handed to the system. This request is translated into a machine readable translation (the *sensing mission*). According to this mission, the available sensor nodes (at the very bottom of Fig. 1) organize and configure themselves to ensembles which are the best available set of sensors to execute this very sensing

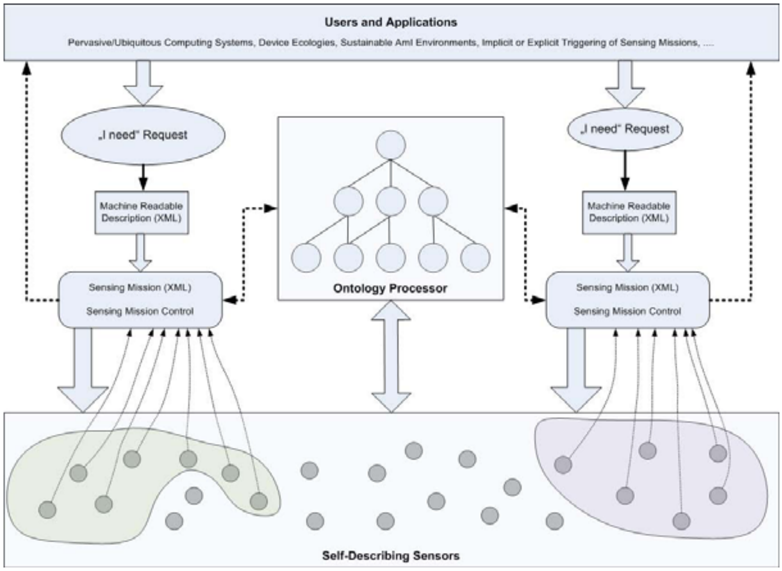


Fig. 1. Architectural concept for an opportunistic activity and context recognition system illustrating the general recognition chain and key-units in an opportunistic system

mission. If possible, according to the available sensor devices, a result for the recognition goal is returned to the requesting entity on top. During the execution of a sensing mission, the system reacts at runtime on topological changes (disconnects, connects or re-connects) in the sensor infrastructure. In the middle of the figure a knowledge-base and knowledge-processing unit (the *Ontology Processor*) is located which is indispensable in an opportunistic system for (i) goal processing and translation, (ii) for describing semantic relations from the sensor's capabilities to the capabilities required for a sensing mission and (iii) to configure coordinated sensor ensembles.

As an opportunistic activity and context recognition system does not specify its initial sensor configurations at design time and thus which types of sensor systems can be used in general, it neither does specify which hard- and software prerequisites that the sensor nodes should be capable of and it also does not specify which type of data the sensors should deliver. Therefore, we need easy to use and commonly accessible software abstractions of different types of sensors for enabling the prototypical implementation of the OPPORTUNITY framework. The purposes of the rest of this paper are to reason and explain (i) why we need sensor abstractions in an opportunistic activity and context recognition enabling framework, (ii) which types of sensors we want to abstract and how these abstractions and implementations can be done from a technical point of view, and (iii) how far we are currently with the implementations and what the next steps in the future are.

The remainder of the paper is structured as follows. The following Section 2 describes the reasons for developing sensor abstractions in detail and which different abstractions are valuable for an opportunistic system by providing examples for the identified types of sensors. Section 3 provides the identified sensor types for the OPPORTUNITY framework including implementation details and an example of a graphical visualization. In Sect. 4 related work is presented and in the last Sect. 5 you can find our conclusion and an outlook to future work.

2 Sensor Abstractions

Different sensor systems that can be roughly classified in the groups physical, logical or virtual devices [16] have different working characteristics, they measure different environmental quantities, deliver different types of data (e.g. acceleration, orientation, sound-level, temperature, humidity, etc.) and they might be accessible and controllable by a system or an application in different ways. Therefore, to have a simple and common standardized access to different sensor devices, wrappers are needed, which encapsulate hardware details and hide the low-level access details (e.g. direct memory access, data transmitting, etc.) that might be very appropriate for a specialized sensor device. Thus, a wrapper is a software abstraction of a sensor device that provides the complete functionality to a system by hiding the complexity. In the *Context Toolkit* (see [10] and [28]) the authors introduce the concept of *context widgets*, which is very similar to the aforementioned *wrappers*. *Dey et al.* define a context widget as *[...] software component that provides applications with access to context information from their operating environment.* The context widgets hide the complexity of the sensor systems, they abstract context information and they provide reusable building blocks for a system. All those characteristics can also be taken as mandatory for the sensor abstractions (the wrappers) in an opportunistic activity and context recognition system. Additionally, we further abstract the sensor systems in different types as given by their working-characteristic and their practicability and suitability, because dealing only with a set of physical sensor abstractions is not satisfactory in an opportunistic activity and context recognition system. The following Sect. 2.1 - Sect. 2.6 provide a description of the different sensor types that we have identified.

2.1 PhysicalSensor

One type that is inevitable are the common physical sensors (referred to as *PhysicalSensor*). According to [31] such sensors systems are small and low-power devices equipped with one or more sensing entities that measure and gather real-world information quantities. They are mostly composed of a core set of units, namely (i) a processing unit, (ii) a memory, (iii) a power supply and (iv) an interface for enabling (wireless) communication with a sink, other sensor nodes or another data processing unit of a system. Take a Sun SPOT (Small

Programmable Object Technology)[1] [29] and an InterSense InertiaCube3[2] as example. Both are physical devices consisting of different hardware units that are able to deliver different environmental data. The SunSPOT (working with a 180MHz 32-bit ARM920T core processor with 512K RAM and 4M Flash, using the 2,4GHz IEEE 802.15.4 radio standard for communication) is equipped with a 3-axis accelerometer, a temperature sensor and a light sensor and is comfortably accessible, programmable and adjustable by using the programming language Java. The InertiaCube3 includes a gyroscope, magnetometers and accelerometers with respect to gravity for 3DoF acceleration, angular velocity and orientation updates at a maximum of 180Hz. The InertiaCube3 sensor systems are accessible with an InterSense SDK dynamic link library (dll). To access and configure both of the sensors easily and equally in a system a wrapper abstracting the physical devices is required.

2.2 OnlineSensor

Another important source of information for an opportunistic activity and context recognition system that requires abstraction are online sources like webservices (referred to as *OnlineSensor*). This type does not rely on a physical device that is attached to a computer. It is rather some online accessible source of information. An example could be a webservice that provides regional weather information. If an opportunistic system requires weather information to successfully execute a recognition goal it could use such an online source if no physical device, that measures temperature and humidity, is currently available. Again, from the system's point of view an OnlineSensor is just another entity acting as sensor that delivers environmental data. The abstraction therefore has to deal with the connection to the remote source, the data acquisition and transmission and the data processing to the opportunistic activity and context recognition system.

2.3 PlaybackSensor

The third type of sensors is more or less a mixture of PhysicalSensors and OnlineSensors, the so called *PlaybackSensor*. Environmental data from a physical sensor device that has been recorded and stored at a previous point in time is replayed in the system and simulating this very (physical) sensor as being available. The replaying of the sensor data is being developed in the OPPORTUNITY framework to work with the same sampling rate and working characteristics as its physical pendant. In [24] and [25] the recording of a multimodal dataset in a kitchen scenario has been done using 72 sensor systems of 10 modalities. For developing, evaluating and testing opportunistic activity and context recognition enabling technologies, a rich dataset can be of high value. It is obvious that it is almost impossible to re-configure the sensor environment to be able to work

[1] http://sunspotworld.com

[2] http://www.intersense.com/InertiaCube_Sensors.aspx

with the same sensor network topology as in the initial (real-world) scenario. Therefore, the PlaybackSensors can be used to simulate a physical sensor device by replaying the recorded data by taking into account the recording characteristics. The location and nature of the (pre-recorded) data source is not fixed. A simple approach could be to use a simple text-, dat- or XML-file as data storage. The following Listing 1 shows a clipping from the content of the data file (stored as **.dat* file) for the magnetic reed switches used in the experimental scenario (recorded at a sampling rate of 100Hz). The first two columns indicate the time when the piece of data was recorded (seconds and microseconds) and the following 16 columns provide the binary (On/Off) data for each of the used magnetic reed switches. A more sophisticated approach for storing the pre-recorded data could use a database or even an online/remote source with distributed data locations. Again, from the systems point of view, the type of storing and accessing the data to enable a sensor simulation is abstracted and thus hidden.

```
1243345571 488238       0  0  0  0  0  0  0  0  1  1  1  0  0  0  0  0
1243345571 488522       0  0  0  0  0  0  0  0  1  1  1  0  0  0  0  0
1243345571 504245       0  0  0  0  0  0  0  0  1  1  1  0  0  0  0  0
1243345571 504482       0  0  0  0  0  1  0  0  1  1  1  0  0  0  0  0
1243345571 520270       0  0  0  0  0  1  0  0  1  1  1  0  0  0  0  0
1243345571 536281       0  0  0  0  0  1  0  0  1  1  1  0  0  0  0  0
1243345571 536349       0  0  0  0  0  1  0  0  1  1  1  0  0  0  0  0
. . .
```

Listing 1. A clipping from the data file from the magnetic reed switches, recorded at the experimental dataset recording sessions [24, 25]

2.4 SyntheticSensor

A *SyntheticSensor* also simulates a physical device, but the difference to the group of PlaybackSensors is the fact that such a synthetic device does not rely on a pre-recorded data source. It rather tries to autonomously generate data and simulate a certain sensors behavior and working characteristics. Different types of data generation mechanisms are applicable for this type of sensors, mainly depending on their field of application. The synthetic device could only be used for generating random (but still plausible and authentic) data that can be utilized for evaluation, testing and demonstration purposes. In this case the data generation entity in a SyntheticSensor relies on a random generator that provides a datastream following a set of given working characteristics of the sensors physical antetype (e.g. sampling rate). Another case when a synthetic device could be useful is to deal with the temporary filling of gaps in the sensor data stream. Take a physical accelerometer that is part of a configured ensemble as example. It delivers data to an opportunistic activity and context recognition system that uses this data in real-time. When this very sensor suddenly slips or disconnects (e.g.

it runs out of power, or simply moves out of the range of the system), this could make a complete reconfiguration of the running sensor ensemble according to a recognition goal necessary. In case that the sensor disappears from the network topology permanently, the reconfiguration of the system might be inevitable. But in case that the sensor reconnects after a few seconds, the reconfiguration of the ensemble might be an effort that could have been avoided. Therefore, a SyntheticSensor could act as a placeholder for at least a few seconds for a physical device to avoid the time- and power-consuming reconfiguration effort. The data generation entity in this case could generate data that is constructed from the historical data that has been delivered by the physical sensor, simply to bridge a possible gap in the datastream and to keep the (running) system stable.

2.5 HarvestSensor

A *HarvestSensor* is more or less a sub-type composition of the PlaybackSensor and the PhysicalSensor, but nevertheless worth mentioning it as independent type. A HarvestSensor in our understanding is a physical device that autonomously collects environmental data, stores this data locally on some internal memory and provides this recorded data when the system puts the sensor in replay-mode. An example for such a device is the GT3X sensor manufactured by *ActiGraph*[3]. It provides activity measures, like steps taken and energy expenditure from persons, and is equipped with a 4MB flash memory, that is capable of storing data for more than one year without having the system attached to a power supply.

2.6 ProxySensor

The last identified sensor type that requires an abstraction in an opportunistic activity and context recognition system is the *ProxySensor*. This type acts as an arbiter much like it is known in a computer network, respectively as surrogate as known from the design patterns in software engineering [14]. A proxy acts as surrogate for another sensor by providing an interface for accessing this system.

2.7 Summary

The aforementioned sensor abstractions are currently being developed in the OPPORTUNITY framework. Main focus lies on the implementation of the sensor systems that have been used in the kitchen-dataset recording [20,24]. The following Tab. 1 provides an overview and summary of the six identified sensor types.

The next Sect. 3 provides a description of the framework, an overview on implementation details of the sensor abstractions and shows exemplary screen dumps from visualizations of the different sensor types.

[3] http://www.theactigraph.com/

Table 1. Sensor Type Overview

Sensor Type	Purpose	Example
PhysicalSensor	hide complexity and hardware details of physical sensor device	use different sensor systems (e.g. SunSPOT and InterSense InertiaCube3) with a common and easy interface
OnlineSensor	hide connection details and data acquisition from online information sources	an online weather-webservice can provide valuable local weather information
PlaybackSensor	replay pre-recorded sensor data by simulating a physical device	reconfigure and simulate a test-setup virtually
SyntheticSensor	simulate a physical sensor by generating plausible data	bridge a gap of a physical accelerometer until it reconnects to the system for at leats a few seconds
HarvestSensor	provide autonomously collected environmental data to a system	sensor system collects data and stores it locally on its memory
ProxySensor	act as a surrogate for another sensor system	a new sensor can be used that is not yet implemented as one of the above types

3 Providing Sensor Abstractions

In the previous Sect. 2 we have introduced six different types of sensor abstractions that have to be differentiated in an opportunistic activity and context recognition system. This section provides implementation and technical details as the abstractions are part of the OPPORTUNITY framework that is currently being developed. Figure 2 summarizes the identified types of sensor abstractions, whereas this figure illustrates the main idea behind this concept. At the very bottom the different types of sensors are located, their software wrappers that encapsulate and hide hardware details are located in the containers directly above. All types of sensors are directly derived from the general type *Sensor*, located at the very top of the figure. The system internally only deals with data delivering entities over a common and easy accessible interface that is equal for all types of sensors (as all of them can be allocated to the general type *Sensor*). Basic and established concepts in object-oriented software design, namely *inheritance* and *abstraction*, enable a sophisticated concept of multi-typed sensor abstraction in an opportunistic activity and context recognition system. The containers hide the complete complexity, the hardware details in case of physical sensor devices, the accessing and connection methods in case of online information sources, or the data generation or replaying mechanisms in case of playback and synthetic devices. From the system's point of view a data delivering source is accessed as *Sensor* with one common interface.

The OPPORTUNITY framework is - as already mentioned - a prototypical implementation of an opportunistic activity and context recognition system used

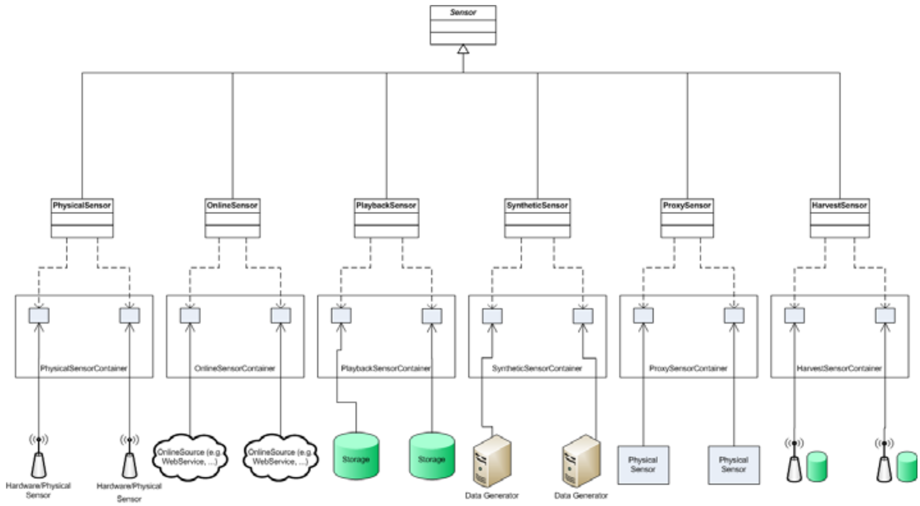


Fig. 2. The six different sensor types and the concept of realization in an opportunistic activity and context recognition system

for evaluation, testing and development. The framework is implemented using the Open Services Gateway Initiative (OSGi)[4] [4,6] and acts as a runtime environment together with a code base and libraries, able to execute autonomously on a target platform. Reasons for implementing the framework in OSGi are (i) the universality of code deployment, (ii) the life cycle management capabilities of OSGi, (iii) the modularity and component oriented paradigm, (iv) the portability and thus compatibility with various different hardware platforms, and (v) the ability to install, restore, start and stop applications at runtime. Generally, OSGi is a collection of bundles that can be interconnected using the OSGi Wire Admin service specification[5], following the producer consumer paradigm to query and propagate data internally from sensors (producers) to sensing mission(s) (consumers) or even from one sensor to another sensor if a direct communication is necessary (e.g. to share localization information, or to negotiate configuration issues).

Furthermore, for demonstrating the actions, the data flows and the ensemble (re-)configurations in an opportunistic activity and context recognition process over time, a real-time graphical visualizer is included in the OPPORTUNITY framework. It displays the currently available sensor device(s) and sensing mission(s), and how they are connected. Figure 3 shows two screen dumps of sensor configurations trying to execute sensing mission(s) by delivering environmental data to the consumer(s). Every colored ellipse illustrates one sensor device that is currently available in the environment and the red rectangle is the visualization unit for a sensing mission. The bigger, gray-colored rectangle at the very

[4] http://www.osgi.org
[5] http://www.osgi.org/About/Technology

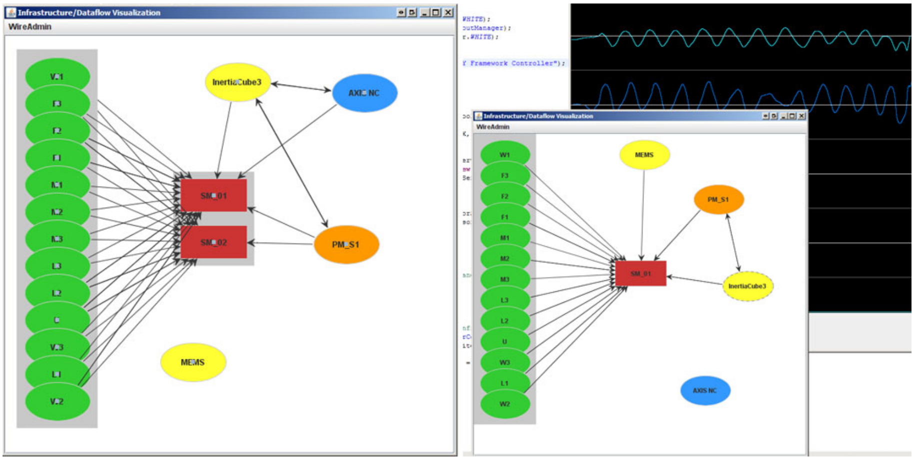


Fig. 3. Screen dumps of two examples showing the visualization of (i) Playback-, (ii) Online-, (iii) Physical- and (iv) Synthetic-Sensors in the OPPORTUNITY Framework

left side of both examples, that holds 13 sensor devices, and the gray rectangle in the left screen dump that holds the two red rectangles means that all of the included are running in the same bundle of the framework (in this case the magnetic reed switches from type PlaybackSensors, and two sensing missions). The arrows in the illustration show the datastreams from the sensors to the sensing mission and between sensors themselves. In the left example two different sensing missions are active that are currently executed. We can see 13 sensors of type *PlaybackSensor* (the green bubbles), whereas all of them are magnetic reed switches that replay prerecorded data. Furthermore, the example shows one sensor of type *OnlineSensor* ("AXIS NC"), which is a network camera and illustrated as blue ellipse, two *PhysicalSensor* ("MEMS" and "InertiaCube3") devices, namely a microphone measuring the noise level and one InertiaCube3 sensor located in a shoe sole that can be used for gait recognition [11], both visualized as yellow bubble and one *SyntheticSensor* ("PM_S1") which is a 3-axis accelerometer synthetic device that delivers random generated acceleration data (displayed as orange-colored bubble). Not every sensor in this example delivers its measured environmental quantities to every sensing mission. According to the sensors capabilities, their locations (which are included in the sensors self-descriptions) and the recognition goals requirements the ensembles are configured to have the best available set of sensors grouped. Furthermore, as we can see in both examples, sensor nodes can share information among themselves, e.g. for sharing localization data, or for configuration purposes. In the left-hand side example, the microphone (one of the yellow bubbles) is not connected to a mission or another sensor node. It would be available and ready to deliver data but it is currently not needed. The screen dump on the right side shows a similar example. There, only one sensing mission is currently active, whereas the sensor devices are the same as in the example on the left. However, as the sensing mission in the second example has other requirements, the ensemble configuration is different. Additionally, we can also see a clipping in the screen dump of the graphical visualization of the data delivered from the InterSense InertiaCube3 sensor (the 3-axis orientation diagram). In case of the types *PlayBackSensor* and *SyntheticSensor*, the OPPORTUNITY framework furthermore provides the functionality to connect, disconnect and reconnect sensor devices at runtime. This is necessary, to create flexible sensor environments that can be use to test and evaluate the ensemble (re-)configuration and more general the execution of sensing missions with different scenarios.

4 Related Work

A lot of related middleware solutions, systems and frameworks exist that enable autonomic, self-organizing computing in pervasive environments, and especially wireless sensor networks with respect to become situation aware. This section provides a summarization of the most valuable solutions and discusses their shortcomings for the opportunistic activity and context recognition problem domain.

The *TinyLime Middleware* [8] supports the development of sensor network applications. This approach departs from the traditional setting where sensor systems deliver measured environmental quantities to one centralized entity. Instead, the middleware supports multiple stations to access and collect data from different sensors. Nevertheless, the authors clearly define that the sensors used with the middleware do not need to be able to communicate with each other. Furthermore, neither the types of supported sensors, nor the integration and abstraction, nor the highly dynamic sensor infrastructure is mentioned, which are key issues in opportunistic systems.

Madden et al. present in [21] the *Tiny AGgregation Service for Ad-Hoc Sensor Networks (TAG)*. The main contribution of *TAG* is the expression of declarative queries that are distributed and executed in wireless sensor networks. This approach is developed for networks of TinyOS sensor nodes which is a limitation that does not meet the requirements of dynamic multi sensor usage by applying sensor abstractions in an opportunistic system.

In [22] the authors present a middleware suitable for dynamic network environments. The approach called *TOTA (Tuples On The Air)*, supports adaptive context-aware activities in pervasive scenarios. Basic principle is to rely on tuples, which are propagated across a network on base of application specific rules. Nevertheless, this approach does not sufficiently meet the requirements of an opportunistic system, like the spontaneous availability of resources, therefore exploiting spontaneous configurations and the goal-oriented sensing capabilities.

Some of the aforementioned approaches are capable of spreading queries to the sensor systems, others propose a non-centralized approach of sensor signal acquisition, which are all important issues in an opportunistic system. Furthermore, as we investigate in building a highly dynamic and flexible system, we cannot presume a static set of sensors, therefore we need the abstracted sensor types solution and as we also do not rely on a fixed recognition goal, the framework must be capable of goal processing and goal-oriented sensing.

5 Conclusion and Future Work

This paper identifies six (*PhysicalSensor*, *PlaybackSensor*, *OnlineSensor*, *SyntheticSensor*, *HarvestSensor* and *ProxySensor*) different types of sensors applicable in an opportunistic activity and context recognition system and describes how they are abstracted. These different sensor abstractions are necessary as an opportunistic system does not specify at design time (i) which sensor system can/shall be used, (ii) how the used sensor devices have to be placed and located and (iii) how they have to be configured according to a recognition goal as this is also not static and fixed. Furthermore, we introduce the OPPORTUNITY framework which is a flexible runtime environment and a prototypical OSGi implementation of an opportunistic activity and context recognition enabling framework. The system uses the sensor abstractions for hiding the complexity and to have a common interface for using different sensor types. For demonstration and testing purposes the framework also includes a graphical visualization of the sensor environment, displaying the currently available sensor devices, the sensing missions that are derived from an recognition goal and the data flows among the sensor devices and the sensing mission(s). With our sensor abstractions that enable the usage of different sensor types and the development of an opportunistic prototypical framework the implementation and realization of a set of a new generation activity and context recognition applications will be possible.

Future work concerns the data generation method in the synthetic devices to enable the gap bridging mechanism to act as a placeholder for a temporary disconnected physical device that is mentioned above. Another important thing that we are currently working on is the processing and translation from a recognition goal formulated in an abstract manner by a user or an application to the machine-readable sensing missions. Therefore, we are working on a goal-description language and appropriate mechanisms to be as flexible and dynamic as possible.

Acknowledgments. The project OPPORTUNITY acknowledges the financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open grant number: 225938.

References

1. Abowd, G.D., Dey, A.K., Brown, P.J., Davies, N., Smith, M., Steggles, P.: Towards a Better Understanding of Context and Context-Awareness. In: Proceedings of the 1st International Symposium on Handheld and Ubiquitous Computing, HUC 1999, pp. 304–307. Springer, London (1999)
2. Agre, J., Clare, L.: An integrated architecture for cooperative sensing networks. Computer 33(5), 106–108 (2000)

3. Akyildiz, I., Su, W., Sankarasubramaniam, Y., Cayirci, E.: A survey on sensor networks. IEEE Communications Magazine 40(8), 102–114 (2002)
4. Alliance, O.: OSGi Service Platform Release 4. architecture p. 9 (2000)
5. Baldauf, M., Dustdar, S., Rosenberg, F.: A survey on context-aware systems. Int. J. Ad Hoc Ubiquitous Comput. 2(4), 263–277 (2007)
6. Bartlett, N.: OSGi in practice. Bd (January 11, 2009)
7. Chong, C.Y., Kumar, S.: Sensor networks: evolution, opportunities, and challenges. Proceedings of the IEEE 91(8), 1247–1256 (2003)
8. Curino, C., Giani, M., Giorgetta, M., Giusti, A., Murphy, A.L., Picco, G.P.: Mobile data collection in sensor networks: The tinylime middleware. Pervasive and Mobile Computing 1(4), 446–469 (2005), http://www.sciencedirect.com/science/article/B7MF1-4H9GRV7-1/2/bf4f596735fdf93adcfd4a0ecfd6255a (special Issue on PerCom 2005)
9. Dey, A.K.: Understanding and using context. Personal Ubiquitous Comput. 5(1), 4–7 (2001)
10. Dey, A.K., Abowd, G.D.: The Context Toolkit: Aiding the Development of Context-Aware Applications, pp. 434–441. ACM Press, New York (1999)
11. Doppler, J., Holl, G., Ferscha, A., Franz, M., Klein, C., dos Santos Rocha, M., Zeidler, A.: Variability in foot-worn sensor placement for activity recognition. In: Proceedings of the 13th International Symposium on Wearable Computers (ISWC 2009), Linz, Austria, September 4-7. IEEE Computer Society Press, Los Alamitos (2009)
12. Dressler, F.: Self-Organization in Sensor and Actor Networks. John Wiley & Sons, Chichester (2007)
13. Ferscha, A.: What is Pervasive Computing, pp. 93–108. Universitätsverlag Rudolf Trauner (June 2003)
14. Gamma, E., Helm, R., Johnson, R., Vlissides, J.: Design patterns: elements of reusable object-oriented software. Addison-wesley, Reading (1995)
15. Garcia, M., Lloret, J.: A cooperative group-based sensor network for environmental monitoring. In: Luo, Y. (ed.) Cooperative Design, Visualization, and Engineering. LNCS, vol. 5738, pp. 276–279. Springer, Heidelberg (2009)
16. Indulska, J., Sutton, P.: Location management in pervasive systems. In: Proceedings of the Australasian Information Security Workshop Conference on ACSW frontiers 2003, ACSW Frontiers 2003, pp. 143–151. Australian Computer Society, Inc., Darlinghurst (2003)
17. Kephart, J.O., Chess, D.M.: The vision of autonomic computing. Computer 36(1), 41–50 (2003)
18. Kephart, J.O., Das, R.: Achieving self-management via utility functions. IEEE Internet Computing 11(1), 40–48 (2007)
19. Li, W., Kamil, Y., Manikas, A.: A wireless array based cooperative sensing model in sensor networks. In: Global Telecommunications Conference on IEEE GLOBECOM 2008, December 4-30, pp. 1–6. IEEE, Los Alamitos (2008)
20. Lukowicz, P., Pirkl, G., Bannach, D., Wagner, F., Calatroni, A., Förster, K., Holleczek, T., Rossi, M., Roggen, D., Troester, G., Doppler, J., Holzmann, C., Riener, A., Ferscha, A., Chavarriaga, R.: Recording a complex, multi modal activity data set for context recognition. In: Proceedings of the 1st Workshop on Context-Systems Design, Evaluation and Optimisation (CosDEO 2010). VDE Publishing House, Hannover (February 2010)
21. Madden, S., Franklin, M.J., Hellerstein, J.M., Hong, W.: Tag: a tiny aggregation service for ad-hoc sensor networks. In: IN OSDI (2002)

22. Mamei, M., Zambonelli, F.: Programming pervasive and mobile computing applications with the tota middleware. ACM Transactions on Software Engineering andMethodology, 18(4), Article 15 263–273 (2009)
23. Puccinelli, D., Haenggi, M.: Wireless sensor networks: applications and challenges of ubiquitous sensing. IEEE Circuits and Systems Magazine 5 (2005)
24. Roggen, D., Calatroni, A., Rossi, M., Holleczek, T., Förster, K., Tröster, G., Lukowicz, P., Bannach, D., Pirkl, G., Ferscha, A., Doppler, J., Holzmann, C., Kurz, M., Holl, G., Chavarriaga, R., Creatura, M., del Milln, R.: Collecting complex activity data sets in highly rich networked sensor environments. In: Proceedings of the Seventh International Conference on Networked Sensing Systems (INSS), Kassel, Germany (June 2010)
25. Roggen, D., Calatroni, A., Rossi, M., Holleczek, T., Förster, K., Tröster, G., Lukowicz, P., Bannach, D., Pirkl, G., Wagner, F., Ferscha, A., Doppler, J., Holzmann, C., Kurz, M., Holl, G., Chavarriaga, R., Creatura, M., del Milln, R.: Walk-through the opportunity dataset for activity recognition in sensor rich environments (May 2010), http://vimeo.com/8704668
26. Roggen, D., Förster, K., Calatroni, A., Bulling, A., Holleczek, T., Troester, G., Lukowicz, P., Pirkl, G., Bannach, D., Ferscha, A., Riener, A., Holzmann, C., Chavarriaga, R., del Milln, R.: OPPORTUNITY: activity and context awareness in opportunistic open-ended sensor environments. Poster at the 1st European Future Emerging Technologies Conference (FET 2009), Prague, Czech Republic (April 2009)
27. Roggen, D., Förster, K., Calatroni, A., Holleczek, T., Fang, Y., Troester, G., Lukowicz, P., Pirkl, G., Bannach, D., Kunze, K., Ferscha, A., Holzmann, C., Riener, A., Chavarriaga, R., del Milln, R.: OPPORTUNITY: Towards opportunistic activity and context recognition systems. In: Proceedings of the 3rd IEEE WoWMoM Workshop on Autonomic and Opportunistic Communications (AOC 2009), June 2009, IEEE CS Press, Kos (2009)
28. Salber, D., Dey, A.K., Abowd, G.D.: The context toolkit: aiding the development of context-enabled applications. In: Proceedings of the SIGCHI Conference on Human factors in Computing Systems, CHI 1999, pp. 434–441. ACM, New York (1999)
29. Smith, R.B.: Spotworld and the sun spot. In: Proceedings of the 6th International Conference on Information Processing in Sensor Networks, IPSN 2007, pp. 565–566. ACM, New York (2007)
30. Ververidis, C.N., Polyzos, G.C.: Service discovery for mobile ad hoc networks: A survey of issues and tec niques. IEEE Communications Surveys and Tutorials 10(1-4), 30–45 (2008)
31. Yick, J., Mukherjee, B., Ghosal, D.: Wireless sensor network survey. Computer Networks 52(12), 2292–2330 (2008)

Identifying Important Action Primitives for High Level Activity Recognition

Atif Manzoor[1], Claudia Villalonga[2], Alberto Calatroni[2], Hong-Linh Truong[1], Daniel Roggen[2], Schahram Dustdar[1], and Gerhard Tröster[2]

[1] Distributed Systems Group, Vienna University of Technology, Vienna
{manzoor,truong,dustdar}@infosys.tuwien.ac.at
[2] Wearable Computing Laboratory, ETH, Zürich
{villalonga,alberto.calatroni,daniel.roggen,
troester}@ife.ee.ethz.ch

Abstract. Smart homes have a user centered design that makes human activity as the most important type of context to adapt the environment according to people's needs. Sensor systems that include a variety of ambient, vision based, and wearable sensors are used to collect and transmit data to reasoning algorithms to recognize human activities at different levels of abstraction. Despite various types of action primitives are extracted from sensor data and used with state of the art classification algorithms there is little understanding of how these action primitives affect high level activity recognition. In this paper we utilize action primitives that can be extracted from data collected by sensors worn on human body and embedded in different objects and environments to identify how various types of action primitives influence the performance of high level activity recognition systems. Our experiments showed that wearable sensors in combination with object sensors clearly play a crucial role in recognizing high level activities and it is indispensable to use wearable sensors in smart homes to improve the performance of activity recognition systems.

Keywords: Activity recognition, smart homes, action primitives.

1 Introduction

Smart homes aim to support people by adapting to their requirements to accomplish their goals and objectives in dynamically changing and continuously emerging situations. Human activity is the fundamental type of context to build many applications in such homes. Consequently, activity recognition has become an active research field to design dependable systems to recognize human activity. Different sensory modalities, including ambient sensors [13,6], vision based sensors [5,7], and wearable sensors [14], are used to observe the environment and reasoning algorithms work out sensor data to detect activities at different levels of abstraction, ranging from the basic short time low level human action primitives, such as sit, stand, and walk, to high level human activities of daily living (ADL) that span over comparatively longer periods of time, such as drinking coffee, eating sandwich, and cleaning room. Still one of the key challenges in building effective and reliable high level activity recognition systems is to identify an

P. Lukowitz, G. Kortuem, and K. Kunze (Eds.): EuroSSC 2010, LNCS 6446, pp. 149–162, 2010.

optimal set of primitives that express enough information to accurately recognize such human activities. Categorizing significant primitives will also decrease the overhead in terms of sensor cost, human effort, and computing resources. But there is yet a lack of research effort that needs to be undertaken to identify the impact of various types of action primitives in recognizing ADL. Most of the works have been limited to use action primitives from a single sensing modality to recognize ADL.

In this paper we identify the importance of the different types of action primitives, such as human locomotion and object usage primitives, to recognize ADL. Distinguishing the role of such primitives will be crucial for two reasons. First, it will help in designing an ambient intelligent environment to indicate where to place sensors, such as on body, in objects, or in the environment. Second, it will also indicate which action primitive are worthwhile to invest additional effort in designing action primitive spotting algorithms to recognize those action primitives. For example, “hand cutting the bread movement” is an important action primitive as it is giving a clear indication that subject is preparing a sandwich. We used the annotations of the EU project OPPORTUNITY data set [11] that are based on the recordings of the proceedings of data collection activity as action primitives. These annotations include body movement primitives, like walk, sit, and stand, arm movement primitives, like reach, release, and cut, and object or environment usage primitives, like use glass, move chair, and open fridge door. Our experiments determined that although human body locomotion primitives are rare to use in recognizing ADL, they showed better performance than object or environmental usage primitives in recognizing some of the ADL. Human body locomotion primitives used in combination with object and environmental usage primitives showed the best performance in recognizing ADLs that make using wearable sensors an indispensable choice to recognize ADL.

The reminder of the paper is organized as follows. Section 2 gives an overview of the existing work in literature. Section 3 briefly describes the process of activity recognition and classification algorithms that have been used in this work. Section 4 presents the detail of the data set that has been used in the experiments. Section 5 exhibits and discusses the result. Finally we present the conclusion of this research effort in Section 6.

2 Related Work

Research efforts that had been undertaken to recognize ADL are mainly dominated by environmental change or the object motion / usage primitives. Kasteren et al. [13] recognized ADL by collecting and using environmental sensor data in a smart home. Sensors were embedded in doors, cupboards, and refrigerator. Mckeever et al. [8] also used the Van Kasteren data set [13]. Tapia et al. [12] used environmental change sensors that had been installed on doors, windows, cabinets, drawers etc. Lepri et al. [5] recognized the ongoing activities by using the visual sensors. They equipped the living room and kitchen of a flat with web cameras where different subjects performed activities of daily living. They processed the video streams to get the primitives about the location and posture of the subject to recognize high level activities. But these primitives had not proved to be enough to recognize activities like eating, drinking, and cleaning. Mostly aforementioned works used sensor systems that can provide primitives about the environmental state change. Lepri et al. [5] used human body posture primitives with their location to detect the ADL. Neither of these works used action primitives related to human body movements like walk, run, and reach.

Logan et al. [6] used the place lab, an instrumented home environment [4], to collect their data set. They used environment built-in sensors, object motion sensors, and RFID tags. Two 3-axis accelerometer sensors were also worn by subject on his limbs to show his motion. In their experiments they provided a comparison of accuracy that was achieved by environmental and object motion sensors. Their experiments showed that a combination of environmental and object motion sensors provided better results. But they only showed their results for all the activities collectively. They did not provide any information if a certain type of primitive provides better performance for a particular activity. As compared to their work we included action primitives extracted from wearable sensor data in our experiments. We also showed our results for each activity separately so we can distinguish which category of action primitives show better performance for a specific activity.

Maekawa et al. [7] used a customized sensor embedded with a camera, microphone, accelerometer, and a digital compass. That sensor set was worn as a wrist band and collect data while the subject was busy in performing different activities. Although they compared the performance of these sensors by using the data sets by including and excluding each type of sensor, they did not provide any comparison on the basis of different type of action primitives. As compared to these works we not only used the object motion / usage and environmental change action primitives but also included human locomotion primitives to classify ADL. Our experiments also showed a comparison of the accuracies of these action primitives, to recognize ADL, individually and in combination with each other.

3 Activity Recognition from Action Primitives

Common systems that recognize high level activities from different sensing modalities collect data from the sensors embedded in the environment and objects and worn by the human as shown in Figure 1. State of the art machine learning algorithms are used to classify action primitives from sensor data. These algorithms are first trained with sensor data to extract primitives of interest. Primitives give information about different current events in the environment, e.g., “subject is walking” describes human body motion. Sets of such primitives are provided to train machine learning algorithms to recognize activities that spread over a longer period of time. Many machine learning algorithms are available in WEKA [3] that provides researchers an easy access to state-of-the-art techniques in machine learning and had been used for activity recognition in different works such as [6,10,2].

We have used J48, Hidden naive Bayes (HNB), and IBK for the purpose of classification of high level activities. J48 is the WEKA implementation of C4.5 decision tree [9]. C4.5 decision tree is also used in [6,10,2] for activity recognition. C4.5 decision tree chooses one attribute of the data that most effectively splits its set of samples based on the criterion of normalized information gain (difference in entropy). The attribute with the highest normalized information determines the decision at each node. A Bayes classifier is a simple probabilistic classifier based on applying Bayes theorem

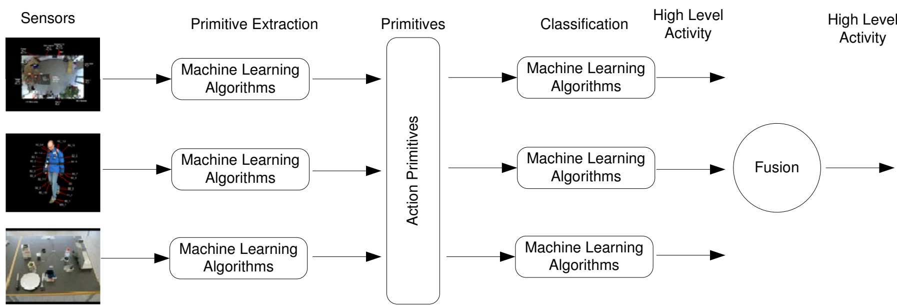


Fig. 1. Activity recognition process using different sensing modalities

with strong feature independence assumptions and is used in [6,10,2] for activity recognition. In this work we have used Hidden naive Bayes [15] that is an extended form of naive Bayes and accommodates the attribute dependencies. HNB creates a hidden parent for each attribute using the average of weighted one-dependence estimators. IBK [1] implements *k*-nearest neighbor, an instance-based learning algorithm that generates classification prediction using only specific instances. *k*-nearest neighbor is also used in [10,2] for activity recognition. We have chosen these commonly used classification algorithms considering different learning strategies used in these algorithms, such as decision tree, probability based learning, and instance based learning. We did not build the histograms of sensor data over a window of time and classified all the samples collected at an instance of time. Reason for these choices was to put more emphasis on the study of the influence of primitive actions on activity recognition rather than on the powers of more advanced classification algorithms or strategies.

A person performs different activities in different parts of a house. She may be busy in kitchen while preparing breakfast or she may be relaxing in the lounge. The different nature of these activities implies that these activities are composed of different primitives. The composition of these activities require to look at each activity individually that will not only give us the opportunity to observe which type of sensors should be used to recognize which activity but it will also indicate which type of sensors should be used in which part of house. Considering these requirements first we look at the influence of each activity individually. Then we study the influence of different action primitives collectively.

4 Data Description

We used the data sets that have been collected in the EU project OPPORTUNITY [11]. The data set about the naturalistic human activities was collected in a sensor rich environment: a room simulating a studio flat with kitchen, deckchair, and outdoor access where subjects performed daily morning activities. 15 networked sensor systems with 72 sensors of 10 modalities were deployed integrated in the environment, objects, on the body. The deployment of the large number of networked sensor systems of different modality make this data set ideal to study the impact of different sensing modalities in activity recognition. Table 1 shows a short description of those activities and their duration for a single run. Twelve subjects executed activities of daily living in this environment, yielding an average of 2 hours of effective data per subject, for a total twenty five hours of sensor data. According to our estimations over 11000 primitives of interactions with objects and over 17000 primitives of interactions with environment have been recorded. This makes the data set highly rich in gesture primitives and the largest for the purpose of multimodal activity recognition.

Table 1. Different activities and duration for single run (in seconds)

Activities	*Description*	*Duration (s)*
Idle	Not performing any activity	583
Relaxing	Go outside and have a walk	157
Early morning	Move around in the room and casually check the objects	276
Coffee time	Prepare coffee with milk and sugar using coffee machine and drink it	129
Sandwich time	Prepare sandwich with bread, cheese, and salami using bread cutter, various knives, and plates and eat it	375
Clean up	Put objects used to original place or dish washer and cleanup the table	183

Table 2 shows the different action primitives that are used in our experiments. These action primitives are extracted from the annotations of the data set performed by experts using the videos of all the proceedings during data collection process. The experts identified all the actions performed by the subject during his activities. Table 3 shows all the sensors that have been deployed in the environment and the type of action primitives that can be extracted from these sensors. Locomotion primitives can be extracted from data collected by the sensors worn on the subject body and include action primitives such as walking, sitting, and lying. Arm locomotion primitives can be extracted from data collected by the sensors worn on the arms of the subject and include the action primitives such as cut, spread, and release. The object data is collected from the interaction of sensors embedded in the arms and objets. Primitives extracted from this data present whether a particular object is used at a specific instance of time. Multiple sensors of different modalities made this data set ideal to perform activity recognition in an opportunistic environment and observe the effectiveness of different sensing modalities. We have used the data of five runs of a single subject.

Table 2. Brief description and values of action primitive categories

<table>
<tr><th>Action primitive category</th><th>Description</th><th>Primitive values</th></tr>
<tr><td>Locomotion</td><td>basic human movements</td><td>walk, run, stand, lie, sit, stairs up, stairs down</td></tr>
<tr><td>Left arm locomotion</td><td>left arm movements</td><td rowspan="2">reach, move, release, lock, unlock, open, close, stir, sip, bite, clean, cut, spread</td></tr>
<tr><td>Right arm locomotion</td><td>right arm movements</td></tr>
<tr><td>Left arm object</td><td>left hand interaction with objects</td><td rowspan="2">fridge, dishwasher, drawer1 (top), drawer2 (middle), drawer3 (lower), door1, door2, switch, table, cup, chair, glass, spoon, sugar, knife salami, knife cheese,salami, bottle, plate, cheese, bread, milk, lazy chair</td></tr>
<tr><td>Right arm object</td><td>right hand interaction with objects</td></tr>
</table>

Table 3. Sensor systems locations and observations

Sensor system	*Location and observation*
Commercial wireless microphone	Chest and dominant wrist. Senses user activity
Custom wireless Bluetooth acceleration sensors	12 locations on the body. Senses limb movement
Custom motion jacket	Jacket including 5 commercial RS485-networked XSense inertial measurement units
Custom magnetic relative positioning sensor	Emitter on shoulder, receiver on dominant wrist. Senses distance of hand to body
Commercial InertiaCube3 inertial sensor system	One per foot, on the shoe toe box. Senses modes of locomotion
Commercial Sun SPOT acceleration sensors	One per foot, right below the outer ankle. Senses modes of locomotion
Custom wireless Bluetooth acceleration and rate of turn sensors	On 12 objects used in the scenario. Senses object use
Commercial wired microphone array	4 at one room side. Senses ambient sound
Commercial Ubisense localization system	Corners of the room. Senses user location
Axis network cameras	3 locations, for localization, documentation and visual annotation
XSense inertial sensor	On the table and chair. Senses vibration and use
USB networked acceleration sensors	USB networked acceleration sensors
Reed switches	13, on doors, drawers, shelves. Sense usage, provides ground truth
Custom power sensors	Connected to coffee machine and bread cutter. Senses usage
Custom pressure sensors	3 on the table, user placed plates and cups on them. Senses usage

5 Experiments

In our experiments we analyzed the impact of the different combinations of action primitives on high level activity prediction. For this purpose we divided the different types of action primitives in seven different combinations. First, we conducted experiments for every individual activity with all primitive sets. Later, we also observed the impact of primitive sets considering all activities collectively. In this section we will discuss the different primitive sets, the impact of those sets on each activity individually and all activities collectively, and finally we will discuss the results and present our recommendations.

5.1 Primitive Sets

Table 4 shows the action primitive sets and the categories of action primitives that have been used in those sets. In S1 we used all the action primitive categories described in Table 2. In S2 we excluded the left arm object movement action primitives and the right arm object movement action primitives. In this set we depended upon the action primitives extracted from wearable sensors like arm motion, e.g., moving, reaching, releasing an object. In S3 we also excluded all the wearable sensors that give us information about the locomotion of human limbs. In this set we used only with action primitives that have values about human actions like walk, sit, and stand. In S4 we used action primitives with both arms locomotion. In S5 and S6 we used left and right arm locomotion respectively. In S7 we used only object sensors, i.e., we had only information about the use of a specific object and we did not have any information that whether concerned person was sitting, standing, lying, or walking. Similarly we did not have any information which hand is used to handle an object. We used these different combination of sensors with the classification algorithms discussed in Section 3. Table 1 presents the detail of the values of these primitives.

Table 4. Different sets of action primitives

Action primitive set	*Categories of action primitives*
S_1	locomotion, left arm movements, right arm movements, left arm object, right arm object
S_2	locomotion, left arm movement, right arm movement
S_3	locomotion
S_4	left arm movement, right arm movement
S_5	right arm movement
S_6	left arm movement
S_7	object movement

5.2 Action Primitives Impact in Recognizing Each Activity Individually

Here we analyze the impact of these primitive sets in recognizing each activity individually. Figure 2 shows the true positive rate of the classification algorithms J48, HNB, and IBK using all primitive sets to recognize the activity *Idle*. The classifiers showed particularly better performance using the action primitive sets S4, S5, S6, and S7. The action primitive sets S4, S5, and S6 present the action primitives extracted from the wearable sensors worn on the limbs. These primitives include information whether the subject has used an object or not. The action primitive set S7 consists of the primitives extracted from object sensors. This action primitive set also provides information about the usage of objects available in the environment. Although primitive sets extracted from object sensors proved relatively better, primitive sets that have been extracted from other sensory modalities are also close. This result was not surprising as when the subject is idle, she is neither interacting with any of the objects nor making much movements. So action primitive extracted from wearable sensor or the objet sensors are not particularly crucial for recognizing this activity and any sensory modality can easily detect whether the subject is idle or not.

Figure 3 shows the true positive rate of the classification algorithms J48, HNB, and IBK using all action primitive sets to recognize the activity *Relaxing*. This activity proved to be most difficult one to recognize. During this activity the subject was either

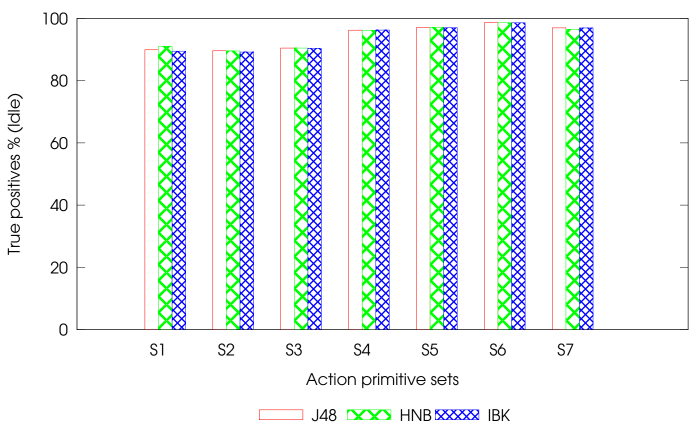


Fig. 2. True positive percentage of activity Idle

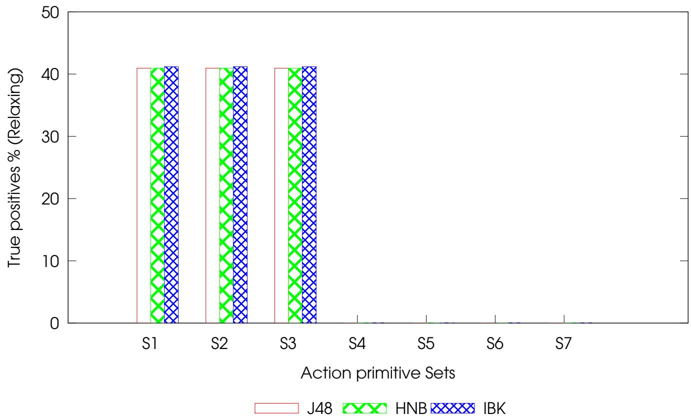


Fig. 3. True positive percentage of activity Relaxing

taking rest or was casually moving around the building. She was neither particularly involved in any activity nor interacting with any object in the environment. The true positive rate to recognize this activity indicates that the classifiers completely failed to recognize this activity using the action primitive sets S4, S5, S6, and S7. The Main reason was that during this activity the subject was neither performing any physical movements nor interacting with the environment and the objects. Classifiers get almost the same feature for this activity as the *Idle* activity. Comparatively the high number of idle activity overwhelmed the decision of classifiers and they got completely confused to distinguish between *Idle* and *Relaxing* activities. The classifiers detected almost all of the *Relaxing* activities as the *Idle* activity. This is also evident from the high false positive rate of J48 using primitive sets S4, S5, S6, and S7 as shown in Figure 8. Primitive sets S1, S2, and S3 proved better in recognizing this activity as the subject was comparatively more dynamic than being completely *Idle*. Recognizing *Relaxing* activity becomes impossible when primitive sets exclude primitives extracted from wearable locomotion sensors. Consequently human body locomotion action primitives extracted from wearable sensors proved vital in recognizing this activity.

Figure 4 shows the true positive rate of classification algorithms J48, HNB, and IBK using all action primitive sets to recognize the activity *Early_morning*. During this activity the subject moved in the room, and randomly checked some objects in the drawers and on the shelf. Although the classifiers using primitive sets S4, S5, and S7 showed better performance in recognizing this activity as compared to recognizing *Relaxing* activity, wearable sensors providing locomotion primitives proved better in this case too. The main reason was that during this activity the subject spends a lot of time to perform physical activities. Again in this case she did not interact much with objects available in the environment. Resultantly, action primitives extracted from object sensors were not able to recognize this activity. Left hand locomotion action primitives proved useless in recognizing this activity. The main reason for their failure is that the subject was casually interacting with different objects and only used dominant right hand for this purpose. Wearable sensor providing human locomotion primitives again proved vital in this case.

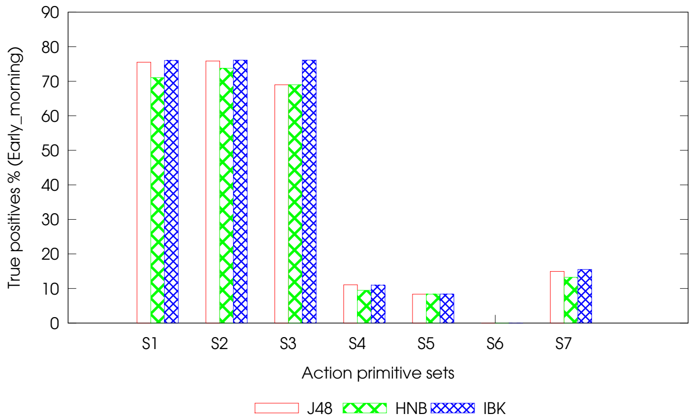


Fig. 4. True positive percentage of activity Early_morning

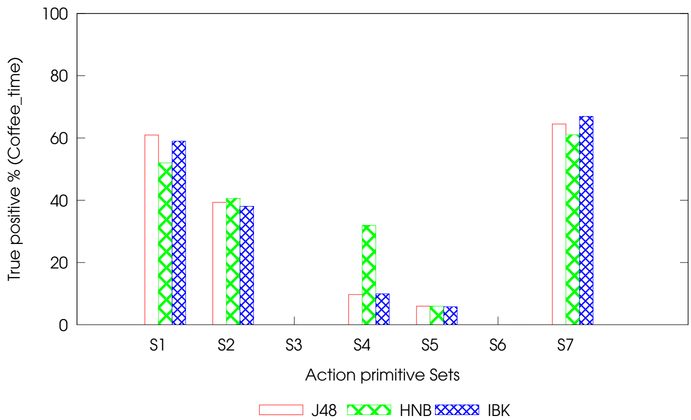


Fig. 5. True positive percentage of activity Coffee_time

Figure 5 shows the true positive rate of the classifiers using all primitive sets to recognize the activity *Coffee_time*. During this activity the subject prepared coffee with milk and sugar by using a machine, took sips of coffee and also interacted with different objects in the environment. As evident from the activity description, this activity is more distinctive on the basis of objects that were used during this activity than human locomotion action primitives. Subsequently object usage primitives also performed comparatively better than human body motion primitives in recognizing this activity.

Figure 6 shows the true positive rate of the classification algorithms J48, HNB, and IBK using all primitive sets to recognize the activity *Sandwich_time*. During this activity the subject interacted with different objects in the environment like bread, cheese, and salami, and bread cutters to prepare the sandwiches. Later the subject ate the sandwiches. Contrasting to *Idle* activity when the subject was motionless most of the time and interacted with few objects, in this activity the subject not only performed many

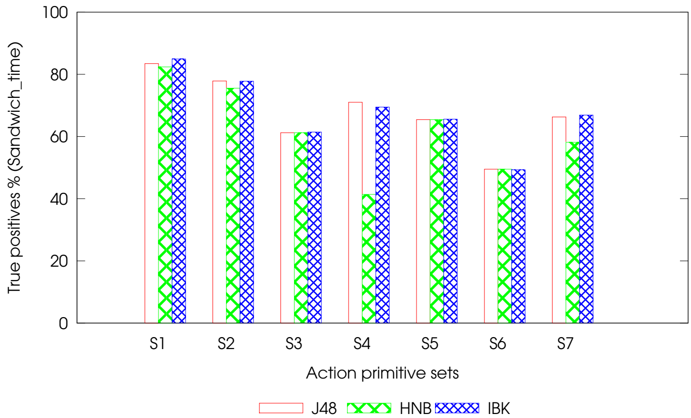


Fig. 6. True positive percentage of activity Sandwich_time

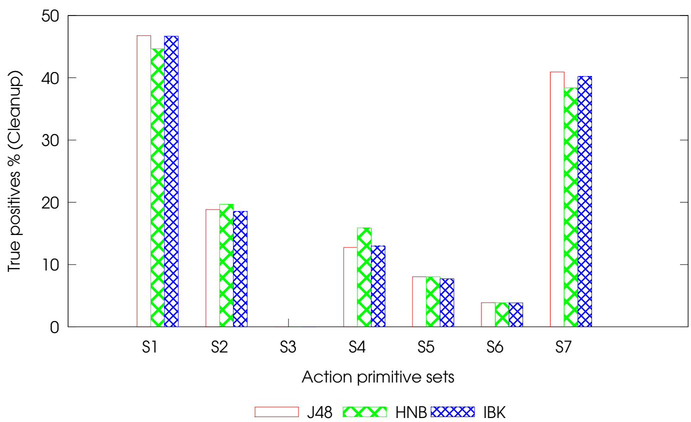


Fig. 7. True positive percentage of activity Cleanup

low level physical activities like cutting the bread but has also interacted with various objects in the environment. As a result all primitive sets proved comparatively better in recognizing this activity. Human body action primitives that can be extracted from wearable sensors data provide better rate of true positives as compared to object usage primitives. But in case of this activity the classifiers clearly performed better when the reason with primitive set that used combination of all action primitives extracted from wearable sensors and object sensors. Combining the action primitives from wearable sensors with primitives about the usage of objects available in the environment provided a clear evidence about *Sandwich_time* activity as indicated by the high true positive rate of algorithms using sensor set S1 in Figure 6.

Figure 7 shows the true positive rate of the classifiers using all primitive sets to recognize the activity *Cleanup*. This activity was the final activity in the drill run for data collection. During this activity the subject put all objects used to original places or dish washer and cleanup the table. Classification algorithms could not show good accuracy for this activity. Body locomotion primitives, such as walk, sit, and stand, proved inadequate when the classifiers used these primitives alone to detect *Cleanup* activity. However, limbs locomotion primitives, such as reach, move, and release, proved better in recognizing this activity. If we compare the performance of all the action primitive sets when used individually, the object usage primitive set showed the best performance as shown in Figure 7. Overall classifiers showed best performance while reasoning with the primitive sets that used combination of human locomotion primitives, limbs locomotion primitives, and object usage primitives to detect *Cleanup* activity.

5.3 Action Primitives Impact in Recognizing All Activities

Figure 8 shows the weighted averages of true positive rates, false positive rates, precision, recall, f-measure, and roc area of J48 classification algorithms while recognizing all activities. The classifier shows highest value of true positive rate with lowest value of false positive rate while using the action primitive set S1 that includes all available types of action. Subsequently we also have good values of other metrics for primitive set S1. Although the use of wearable sensors to recognize high level activities is very rare the classifier also showed comparatively good performance while using primitive set S2 that only contains primitives extracted from wearable sensor data. The reason for the good performance of the classifier using primitive set S2 is the comprehensive nature of the action primitives that were extracted from wearable sensor data. S2 not only provide information about the action primitives like walk, sit, and stand but also indicate that one of the objects available in the environment is used. These primitives also proved very helpful in recognizing activities like *Idle*, when subject is not

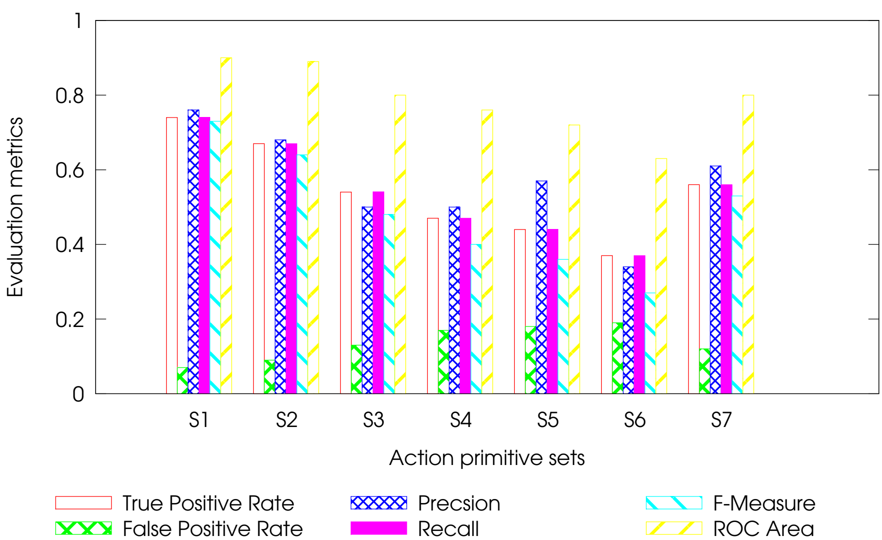


Fig. 8. Evaluation metrics for different primitive sets using decision tree

performing any activity, *Early_morning*, when subject is walking around and handling different objects, and *Relaxing* when subject is siting or lying. However, classifiers get confused when they have to detect a single activity among ones that used same objects as there have not been any information about which object is used as shown by the high value of false positive rate and low value of precision as compared with sensor set S1.

The Classifier also showed comparable performance while using primitive set S3 and primitive set S7. While reasoning with S3 and S7, weighted averages of different evaluation metrics of J48 classifier to recognize all activities are almost equal as shown in Figure 8. The main reason is that human locomotion primitives are better in recognizing activities like *Relaxing* and *Early_morning* while object sensors were proved better in detecting activities in which subject have higher number of interaction with different objects such as *Sanwich_time*. The classifier, using these sets, got confused in recognizing other activities as evident by their high false positive rate and low value of precision. Classifier, using primitive set S4, S4 has not performed as good as while using primitive set S3 and S7. Although the classifiers reasoning with both of primitive sets S5 and S6 showed good performance for some of the activities, they could not show overall good accuracy. Primitive set S7 proved comparatively better. Clearly primitive set S1 that used combination of object sensors with wearable sensors proved the best to recognize human activities in smart home environment.

5.4 Discussion and Recommendations

Figure 8 shows that the classifiers have nearly equal values of the weighted average of true positive rates of recognizing all activities while using human locomotion action primitives and object usage action primitives individually. But if we consider each activity separately that is detected by the classifiers while using different action primitive sets as shown in Figures 2 - 7, we find that those activities are completely different for each type of action primitives. While using human locomotion action primitives individually, the classifiers performed better in recognizing those activities where the subject has less interaction with objects available in the environment. These activities include *Idle*, *Relaxing* and *Early_morning*. The classifiers completely failed in recognizing these activities while using objects and environment usage action primitives alone. But while using object and environmental usage action primitives classifiers performed better in recognizing activities that involve a lot of interaction with objects and environment. These activities include *Coffee_time*, *Sandwich_time*, and *Cleanup*. Collectively the classifiers performed best while using object and environmental usage action primitives and human body locomotion action primitives together.

Limbs locomotion primitives, like reach, cut, and touch, also proved significant in recognizing those activities that include not only using but also performing actions on different objects. Classifiers performed better while working with limbs locomotion primitives extracted from wearable sensors attached with dominant limb than the other limb. Wearable sensors that can be used to extract action primitives like sip or bite are also important in correctly distinguishing activities like drinking coffee or eating a sandwich. As action primitives extracted from wearable sensors used in combination with action primitives extracted from ambient sensors clearly proved better in recognizing high level human activities than action primitives extracted from ambient sensors alone, it is indispensable to use wearable sensors in smart environments to improve the performance of the classifiers to recognize high level human activities. We highly recommend to use wearable sensors with dominant limbs to extract locomotion action primitives. We found that object usage primitives are quite fundamental in recognizing the activities that were performed in areas where the subject have more interaction with the environment and objects, such as home kitchen. It is very important to install ambient sensors in these areas. Limbs locomotion primitives are also proved helpful in recognizing kitchen area activities.

6 Summary

In this paper we compare the performance of the classifiers reasoning with different sets of action primitives to recognize high level human activities. The main purpose of this study is to analyze the impact of the diverse type of action primitives on high level activity recognition and identify important action primitives. We include human body and limbs locomotion action primitives along with objects and environment usage action primitives in our experiments. While reasoning with human body locomotion action primitives individually, the classifiers show good performance in recognizing activities that do not involve interaction with environment, such as *Idle*, *Relaxing*, and *Early_morning* activities. The classifiers failed to recognize these activities while working with objects and environment usage action primitives. This fact makes wearable sensors that can be used to extract human body locomotion primitives an indispensable choice in the development of human activity recognition system. Overall the classifiers show best performance while using human body locomotion action primitives with objects and environmental usage action primitives. The classifiers also show better performance while using limbs locomotion primitives related to dominant arm than using limbs locomotion primitives related to other arm. We recommend to use wearable sensors with dominant limbs and ambient sensors at places where a person have a lot of interaction with environment and objects, like home kitchen, to effectively recognize high level human activities.

Acknowledgements. This work is financially supported by EU FP7 STREP Project SM4ALL (Smart hoMes for ALL) under Grant No. 224332 and EU FP7 FET Project OPPORTUNITY under Grant No. 225938.

References

1. Aha, D., Kibler, D.: Instance-based learning algorithms. Machine Learning 6, 37–66 (1991)
2. Bao, L., Intille, S.S.: Activity recognition from user-annotated acceleration data. In: Proceedings of Internation Conference of Pervasive Computing 2004, pp. 1–17 (2004)
3. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The weka data mining software: an update. SIGKDD Explor. Newsl. 11(1), 10–18 (2009)
4. Intille, S., Larson, K., Tapia, E., Beaudin, J., Kaushik, P., Nawyn, J., Rockinson, R.: Using a live-in laboratory for ubiquitous computing research. In: Fishkin, K.P., Schiele, B., Nixon, P., Quigley, A. (eds.) PERVASIVE 2006. LNCS, vol. 3968, pp. 349–365. Springer, Heidelberg (2006)

5. Lepri, B., Mana, N., Cappelletti, A., Pianesi, F., Zancanaro, M.: What is happening now? detection of activities of daily living from simple visual features. In: Personal and Ubiquitous Computing, pp. 1–18 (2010)
6. Logan, B., Healey, J., Philipose, M., Tapia, E.M., Intille, S.: A long-term evaluation of sensing modalities for activity recognition. In: Krumm, J., Abowd, G.D., Seneviratne, A., Strang, T. (eds.) UbiComp 2007. LNCS, vol. 4717, pp. 483–500. Springer, Heidelberg (2007)
7. Maekawa, T., Yanagisawa, Y., Kishino, Y., Ishiguro, K., Kamei, K., Sakurai, Y., Okadome, T.: Object-based activity recognition with heterogeneous sensors on wrist. In: Floréen, P., Krüger, A., Spasojevic, M. (eds.) Pervasive Computing. LNCS, vol. 6030, pp. 246–264. Springer, Heidelberg (2010)
8. McKeever, S., Ye, J., Coyle, L., Dobson, S.: Using dempster-shafer theory of evidence for situation inference. In: Barnaghi, P., Moessner, K., Presser, M., Meissner, S. (eds.) EuroSSC 2009. LNCS, vol. 5741, pp. 149–162. Springer, Heidelberg (2009)
9. Quinlan, R.: C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo (1993)
10. Ravi, N., Dandekar, N., Mysore, P., Littman, M.L.: Activity recognition from accelerometer data. American Association for Artificial Intelligence (2005)
11. Roggen, D., Calatroni, A., Rossi, M., Holleczek, T., Förster, K., Tröster, G., Lukowicz, P., Bannach, D., Pirkl, G., Ferscha, A., Doppler, J., Holzmann, C., Kurz, M., Holl, G., Chavarriaga, R., Creatura, M., del Millàn, R.: Collecting complex activity data sets in highly rich networked sensor environments. In: Proceedings of the 7th International Conference on Networked Sensing Systems, INSS 2010. IEEE, Los Alamitos (2010)
12. Tapia, E.M., Intille, S.S., Larson, K.: Activity recognition in the home using simple and ubiquitous sensors. In: Ferscha, A., Mattern, F. (eds.) PERVASIVE 2004. LNCS, vol. 3001, pp. 158–175. Springer, Heidelberg (2004)
13. van Kasteren, T., Noulas, A., Englebienne, G., Kröse, B.: Accurate activity recognition in a home setting. In: Proceedings of the 10th International Conference on Ubiquitous Computing, UbiComp 2008, pp. 1–9. ACM, New York (2008)
14. Zappi, P., Lombriser, C., Stiefmeier, T., Farella, E., Roggen, D., Benini, L., Tröster, G.: Activity recognition from on-body sensors: accuracy-power trade-off by dynamic sensor selection. In: Verdone, R. (ed.) EWSN 2008. LNCS, vol. 4913, pp. 17–33. Springer, Heidelberg (2008)
15. Zhang, H., Jiang, L., Su, J.: Hidden naive bayes. In: Twentieth National Conference on Artificial Intelligence, pp. 919–924. AAAI Press, Menlo Park (2005)

A Resource Model for the Real World Internet

Claudia Villalonga[1,2], Martin Bauer[3], Fernando López Aguilar[4], Vincent A. Huang[5], and Martin Strohbach[3]

[1] SAP Research, CEC Zürich, Switzerland
claudia.villalonga@sap.com
[2] Wearable Computing Lab., ETH Zürich, Switzerland
[3] NEC Laboratories Europe, NEC Europe Ltd., Germany
{Martin.Bauer,Martin.Strohbach}@neclab.eu
[4] Telefónica Investigación y Desarrollo, Spain
fla@tid.es
[5] Ericsson Research, Ericsson AB, Sweden
vincent.a.huang@ericsson.com

Abstract. Integrating Wireless Sensor & Actuator Networks (WS&AN) on a large scale allows horizontal applications to access real-world information in real-time and changing the state of the real world, thus providing the basis for the Real World Internet. In this paper we present a resource model that semantically describes sensor, actuator and processing resources for the Real World Internet, this model is designed to efficiently support information queries and execute actuation requests in a large scale system.

Keywords: Real World Internet, Sensor and Actuator Networks, Resource, Resource Model, Resource Description.

1 Introduction

As the deployment of Wireless Sensor & Actuator Networks (WS&AN) become more and more common, the idea is to reuse this infrastructure for multiple purposes, i.e., move away from vertical application silos to a horizontal integration. The result will be a framework that enables a large variety of applications to access real-world information in real-time, thus providing the basis for the Real World Internet. To make the real-world information useful, it is not sufficient to have the plain observations and measurements provided by the sensor nodes themselves. These need to be augmented with semantic information, e.g., what information is provided, when it was measured or observed and where. This makes them useful for a variety of different applications, some of which may not even exist at the WSAN setup time.

Often, applications will not be interested in the information provided by a single sensor, but rather in aggregated or higher-level information, which may be used as context information. Therefore processing element based on different technologies from signal processing to reasoning have to be integrated in the

P. Lukowitz, G. Kortuem, and K. Kunze (Eds.): EuroSSC 2010, LNCS 6446, pp. 163–176, 2010.

framework. For true interaction with the real world, the use of actuator components is needed which should also be exposed to applications on a suitable abstraction level.

To make it easy for applications and services to get access to a heterogeneous mix of sensors, processors and actuators, and to be able to combine these into processing chains, a common abstraction is needed. For this purpose, we have introduced the resource concept. All sensors, processors and actuators are exposed to the overall framework in the form of resources. Since resources will differ in the functionality they can provide, it is not reasonable to enforce one single common interface. Instead, our approach is to define a common resource model based on which sensor resources, actuator resources and processing resources can be described in such a way, that they can be efficiently found and accessed, as well as integrated into processing chains. For this purpose, a syntactic description of the interfaces is not sufficient. In addition, a semantic description is needed to describe not only the result a resource produces, e.g., what information it provides, but also what dependencies it has to other resources to be able to provide these results.

The remainder of this paper is structured as follows. Section 2 provides an overview of the framework architecture, focusing on the resource concept. The core of the paper is Section 3 that describes the requirements on the resource model and the resource model itself. Section 4 shows how the resource model is used in resource discovery as well as setting up of processing chains. Section 5 puts our work into the context of the state of the art. Finally, and Section 6 concludes the paper.

2 Architecture - Resource Concept

Resources are the core concept in our architecture, which is shown in the center of Figure 1. All sensors, actuators, and processors are visible in our framework in the form of resources. WS&ANs can be modeled as resources in different ways, i.e., one sensor may correspond to one resource, or the whole WS&AN may be modelled as one or more resources. Resources are hosted by *Resource Hosts*. A Resource Host could be a sensor node as well as a gateway. Since, for example, not all sensor nodes are directly accessible from our framework, they are made accessible through *Resource Endpoints (REP)*. These are contained in a *REP Host*, which, for example, could be a gateway to a sensor network.

A REP implements one or more *Resource Access Interfaces (RAI)*. The RAI interface is used to interact with a Resource. The RAI interface is not homogeneous, as the functionality provided by a simple sensor significantly differs from the functionality offered by a complex high-level processor. A RAI interface can also support different interaction patterns (e.g. synchronous and asynchronous interactions). For the scalability of our architecture it is important to note that resources only provide information if this was previously requested via the RAI interface.

Our framework provides access to information on two different levels, as observation & measurments augmented with semantic information that can be

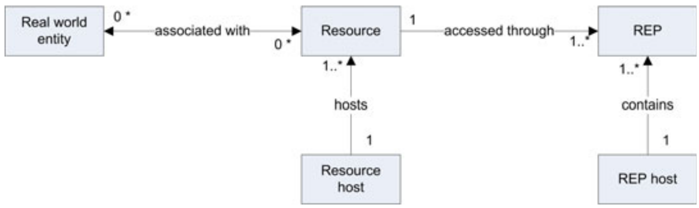


Fig. 1. Resource Concept

directly provided by sensor resources or as context information provided by advanced architecture components as well as by context-level resources. Our context model is based on real world entities, e.g., persons, objects or places. A detailed description of our two level information model can be found in [1]. To be able to provide this context level information, associations between Resources and Real World Entities (see Figure 1) are needed, e.g., the information that a temperature sensor provides the indoor temperature of a room.

Figure 2 shows the architecture of our framework. The Resource Hosts publish Resource Descriptions, which are based on our Resource Model, to the *Resource Directory*, which implements a simple rendezvous-functionality. The Resource concept and the uniform Resource Descriptions enable *Resource Users* to homogeneously discover and access the heterogeneous substrate of WS&ANs. Together Resource Users, Resources and the Resource Directory form the core of the SENSEI architecture.

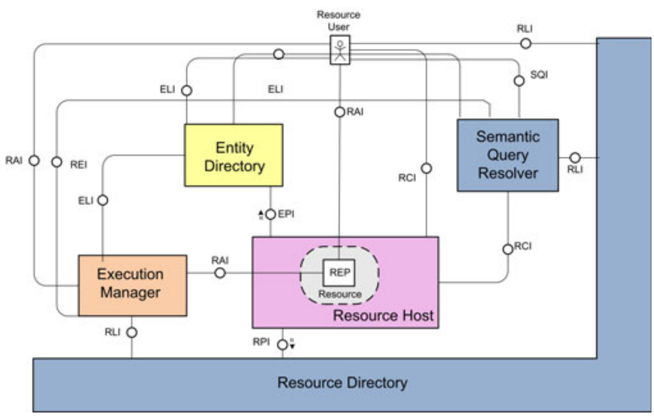


Fig. 2. Framework Architecture

The advanced architecture components consisting of Entity Directory, Semantic Query Resolver and Execution Manager. The *Entity Directory* stores the associations between Resources and Real-World Entities. The *Semantic Query Resolver* is responsible for analyzing requests from the Resource Users, looking up required Resources using the Entity and Resource Directories, planning and, as a result, creating an execution plan, possibly consisting of multiple Resources forming a processing chain. The execution plan is passed on to the Execution Manager that is responsible for setting up the execution on behalf of the Resource User. The architecture also supports cases, where Processing Resources are only created at runtime. Resource Hosts that allow the creation of resources store respective templates in the Resource and Entity Directories. When these are returned as a result of a lookup request, the Semantic Query Resolve (or possibly the Resource User) can trigger the creation of the Resource by calling the RCI interface of the Resource Host with the required parameters.

3 Resource Model

The Resource concept requires that we define an underlying *Resource Model* that clearly models the important aspects of Resources, especially their functionalities, and where and how they can be accessed.

Whereas the Resource Model provides a conceptual view, relating the important aspects, the concrete instantiation of this information can be found in the *Resource Description*, which is published to the Resource Directory.

The Modeling of Resources has to enable finding the appropriate resources and combining the required resources to form processing chains.

To allow human users to find Resources, it may be sufficient in certain cases to describe the Resources with a number of keywords or *tags*, as it is done for pictures or videos on the Web. However, simple free-style tags are clearly insufficient for any machine-based interaction, where the planning and execution of a request is to be executed by a machine. Here the syntax and semantics of the interfaces need to be clearly defined.

The most important aspect for a Resource is the result it can produce. For Sensor Resources and Processing Resources this is the information they can produce; for Actuator Resources it is the post-condition that is achieved. Processing Resources and more complex Actuation Resources also require inputs or pre-conditions to be fulfilled. This means that we clearly have to model the dependencies between inputs and outputs, as well as between pre-conditions and post-conditions. In addition, there may be further non-functional requirements on inputs and pre-conditions, especially quality related aspects. This information is required for building processing chains, where the outputs of Resources provide the input for other Resources and where the pre-conditions of one Resource are produced by other resources.

3.1 SENSEI Resource Model

The Resource Model envisions that each *Resource* has associated a *Resource Description* stating the functionalities provided by resource and how to access them, how it is shown in the Figure 3.

In addition to it, each *Resource* could have associated a list of *Operations* uniquely identified to deal with. These operations have a *Syntactic Operation*

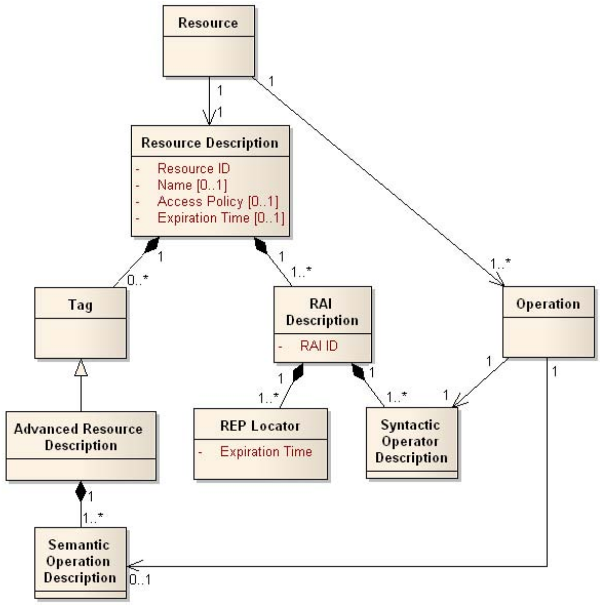


Fig. 3. Resource Model

Description, which describes syntactically the operations, the inputs they take and the returned outputs. The Operations might also have associated a *Semantic Operation Description*, which adds semantic information to provide automatically interpretable information about the operations, its inputs, outputs pre-conditions and post-conditions. For example, a light sensor which offers an operation for retrieving the current level of luminosity in a specific room has associated a Syntactic Operation Description which states that the operation's output is a decimal and a Semantic Operation Description which describes that this decimal represents an instance of the class luminosity. We further explain the operations in the next paragraphs when we introduce the resource access interface (RAI) and the advanced resource description (ARD).

Following with the *Resource Description*, it is necessary to identify uniquely a Resource for administrative domain. It is made through the *Resource ID*, which would take into consideration for example the used technology and the MAC address. It is also possible to have an optional human readable name of the Resource Description through the field *Name*. We have also two optional fields within this class, *Access Policy* and *Expiration Time*. The first one is used to manage access restriction to the *Resource Description*. The second one indicates when the provided information is no more longer valid in time.

Once we have described the relevant information of a Resource, we need to know the resource access interface (RAI). From the conceptual point of view, there are several possibilities to access Resources depending on the technology that we are using like Web Services, Rest and others, but it is also necessary to consider that Resources can be accessed through different URLs. For example, if a Resource change the position, it has other URL to access it, maintaining the resource access interface. Other example that we need to consider is when we have several gateways to access at the same resource with different access policies, all of which share the resource access interface but with different URLs.

This leads us to define two classes, *RAI Description* and *REP Locator*. The first one is used to specify the interface to the operations associated to a Resource and so allows the resource users to know how to access the operations offered by the resource. This information can be a WADL[1] or WSDL[2] file or any other specification file, which describes the method used to access to the resource. This led us to define one or more *RAI Description* for each *Resource Description*. The following component, *REP Locator*, is the link or locator to REPs, where the resource can be accessed and, like the *RAI Description*, one or more *REP Locator* are associated to a *RAI Description*. Besides this information, this link might have a time in which it is no more valid in time like *Resource Description* but, in contrast to *Resource Description*'s field, here is related to the physical link or locator expiration time to access the Resource. So, it might be necessary contain an optional *REP Expiration Time*. Finally, *RAI Description* contains also the *Syntactic Operation Description* of each one of the *Operations* offered by the *Resource*.

Additionally, the *Resource Description* may contain *Tags* that provide additional description of the resource. Also, associated to the *Tag*, we can find the *Advanced Resource Description*, which allow us to characterize the resources and describe their properties more accurately adding semantic information to them, both operations, input parameters, outputs parameters, pre-conditions and post-conditions. This allows us to provide us high level functionalities and automatically interpretable information, which allows to a Resource User to efficiently find resources that match specific parameters on their descriptions, to compose new resources based on the description of the available ones and to process efficient information requests or actuation tasks.

3.2 Advanced Resource Description

The Advanced Resource Description is the tag associated to the Resource Description which describes semantically the resource, basically its operations.

The Advanced Resource Description is defined as an ontology and the Advanced Resource Description tag is an instance of the main ontology class, the *ard:AdvancedResourceDescription* (see Figure 4).

The *ard:AdvancedResourceDescription* contains several parts: the Description, the Resource Type, the Location, the Temporal Availability and the Semantic Operation Description.

Description

The **Description** is some human readable information about the resource and it is represented by the class *rdfs:Literal*.

[1] `https://wadl.dev.java.net`

[2] `http://www.w3.org/TR/wsdl`

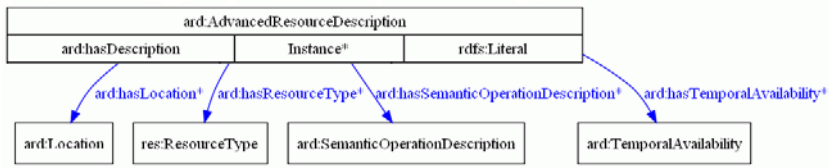


Fig. 4. Advanced Resource Description

The property *ard:hasDescription* which has as domain the *ard:Advanced ResourceDescription* class and as range the *rdfs:Literal* is used to link the human readable text to the Advanced Resource Description.

Resource type

The **Resource Type** defines the type of resource, e.g. sensor, actuator, processor. The types of resources are defined hierarchically in the ontology, the *res:ResourceType* is the root class for the resource types and has as subclasses *res:Sensor*, *res:Actuator*, *res:Processor* and their subclasses.

The *ard:hasResourceType* property links the Advanced Resource Description to the Resource Type and has as domain the *ard:AdvancedResourceDescription* class and as range the *res:ResourceType* class.

Location

The **Location** describes the position where the resource is located and it is represented by the class *ard:Location*.

The *ard:hasLocation*property has as domain the *ard:AdvancedResource Description* class, as range the *ard:Location* class and a multiplicity zero or more since a resource can have different symbolic locations.

The *ard:Location* class has to be further specified for each domain, for example defining two new properties for the latitude and the longitude of the GPS coordinates, or mapped to existing ontologies in order to allow reusability.

Temporal Availability

The **Temporal Availability** is the interval of time during which the resource is available or an operation of the resource is available. The *ard:TemporalAvailability* class is used to define this temporal concept and existing time ontologies can be used.

The *ard:hasTemporalAvailability* property has as domain the classes *ard: AdvancedResourceDescription* and *ard:SemanticOperationDescription* and has as range the class *ard:TemporalAvailability*.

Semantic Operation Description

The **Semantic Operation Description** defines semantically the operations offered by the resource via the *ard:SemanticOperationDescription* class (see Figure 5). For each operation it specifies the inputs that a resource takes in order to provide each output, the pre-conditions and the post-conditions derived from invoking an operation and the temporal availability of the operation.

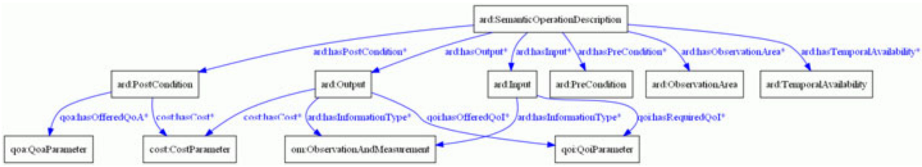


Fig. 5. Semantic Operation Description

The *ard:hasSemanticOperationDescription* property has as domain the class *ard:AdvancedResourceDescription* and as range the *ard:SemanticOperation Description* class. The *ard:SemanticOperationDescription* class is part of the domain of several properties the *ard:hasInput* which has as range the *ard:Input*, the *ard:hasOutput* which has as range the *ard:Output*, the *ard:hasPreCondition* which has as range the *ard:PreCondition*, the *ard:hasPostCondition* which has as range the *ard:PostCondition*, and the *ard:hasObservationArea* which has as range the *ard:ObservationArea*.

An **Input Parameter** taken by an operation of the resource is specified by the *ard:Input*. The inputs of an operation offered by a resource contain the Information Type which describes the type of information that the resource takes to perform a task and has to be compliant with the Information Model. Resources can take as input sensor, context information or other information given as part of the request. The Input can also have associated some QoI Parameters that state the minimum quality required by that input.

The **Output Parameter** offered by an operation of the resource is specified by the *ard:Output* class. The output of an operation offered by a resource contains the Information Type which describes the type of information provided by the resource and has to be compliant with the Information Model. In the case of Sensor Resources, that provide a simple observation and measurement, this output parameter is of type "sensor information". In the case of Context Resources, which provide context information about an Entity of Interest, the output parameter is of type "context".

The **Pre-conditions**, i.e. conditions that have to be fulfilled before invoking an operation offered by the resource, are defined by the *ard:PreCondition* class.

The **Post-conditions**, i.e. the effects resulting from the invocation of an operation offered by the resource, are defined by the *ard:ProstCondition* class. The Post-Conditions can also have associated some QoA Parameters that state the quality of the effect resulting from the invocation of the operation.

The *ard:Input* class and the *ard:Output* class are part of the domain of the *ard:hasInformationType* which has as range the *om:ObservationAndMesurement*, which are classes of the ontology used to represent the Observation and Measurement model (e.g. smoke, temperature) or context if the resource provides information that follows the Context Model. I.e., the *ard:hasInformationType* property links the Resource Model with the Information Model.

The ard:PreCondition class and the ard:PostCondition class are used for actuation.

Observation Area

The **Observation Area** describes physical area for which an operation offered by a resource can provide information or perform an actuation task. As example for a satellite resource, the Observation Area would be the area of the Earth which is observed by the satellite, whereas the Location of the satellite is its position in its orbit around the Earth.

The **Observation Area** is represented by the class *ard:ObservationArea* which can be defined as two pairs of GPS coordinates that define a rectangular area, mapped to an symbolic location, or to any existing ontology for reusability purposes.

Due to symbolic location of the Observation Area, the *ard:hasObservationArea* property, which has as domain the *ard:AdvancedResourceDescription* class and as range the *ard:ObservationArea* class, has cardinality zero or more.

Quality and Cost

An important element of the resource description are quality and cost parameters associated to the input and output of a resource. We distinguish between *Quality of Information* (QoI) and *Quality of Actuation* (QoA):

> **Quality of Information (QoI)** is any metadata that characterizes sensor or context information in such a way that it can be used to infer the reliability of the received information. QoI refers to the information itself and not to the process or the hardware component that possibly provides the information.

> **Quality of Actuation (QoA)** is any information that describes constraints on how well specific actuation request should be carried out.

The advanced resource description captures static and slow changing quality metadata based on a rich set of often used quality parameters. We distinguish between temporal (e.g. measurement time, temporal validity), reliability (e.g. measurement and classification errors) and traceability (e.g. observer, owner, network) QoI paramters.

Likewise QoA parameters describe temporal, spatial and reliability aspects of an actuation operation. Actuation operations can be scheduled at a future point in time and scoped to a spatial area that defines where the actuation should happen. QoA parameters allow a resource to describe the accuracy of the execution time, scopes and the spatial scope.

In addition, cost parameters define how expensive access to a resource is. For planning we use a single cost parameter that may depend on real energy and communication costs or be completely virtual. A resource can specify this cost parameter directly or provide a function for calculating the costs based on technical parameters such as energy consumption or communication costs.

Cost and quality parameters provide the SQR with sufficient information for selecting resources so that an operation can be executed meeting the minimum quality and maximum costs specified by a resource user. Effectively, this allows a business model in which resource providers can charge per access and compete with other providers by offering higher quality at lower costs.

4 Resource Model Evaluation

The Resource Model serves as basis for the correct and efficient work of the Resource Directory and the Semantic Query Resolver (SQR).

Components of the SENSEI architecture make use of the existing web technologies, e.g. they are based on a RESTful design [2] and they are implemented using the RESTlet framework[3]. For the semantic reasoning, the Resource Directory relies on the Jena framework[4], it stores the Advanced Resource Descriptions in the RDF format[5] and supports queries based on the SPARQL language[6].

Resource Hosts publish the resource descriptions in an XML format including the RDF representation for the Advanced Resource Description tag. The Resource User or the SQR can lookup for the resource descriptions matching the tags or the SPARQL query.

Let us imagine a temperature sensor which is connected to the SENSEI framework. It is modeled following our Resource Model and it has attached a resource description which contains semantic information in the ARD tag. In order to retrieve the temperature measured by the sensor, this resource offers an operation which returns as output the temperature as an integer. In the resource description, the RAI Description contains a WADL with syntactic information about operation, e.g. information that the resource returns an integer. The ARD tag (see Figure 6) contains the semantic information about the resource and models the operation as having an output of type *om:Temperature*.

```
<rdf:RDF xmlns:rdf="&rdf;" xmlns:om="&om;" xmlns:res="&res;" xmlns:ard="&ard;" xmlns:rdfs="&rdfs;" xmlns:id="&id;">
        <ard:AdvancedResourceDescription rdf:about="&id;ARD">
                <ard:hasResourceType rdf:resource="&res;Sensor"/>
                <ard:hasSemanticOperationDescription rdf:resource="&id;Op:req_value"/>
        </ard:AdvancedResourceDescription>
        <ard:SemanticOperationDescription rdf:about="&id;Op:req_value">
                <ard:hasOutput rdf:resource="&id;Op:req_value-Out:1"/>
        </ard:SemanticOperationDescription>
        <ard:Output rdf:about="&id;Op:req_value-Out:1">
                <ard:hasInformationType rdf:resource="&om;Temperature"/>
        </ard:Output>
</rdf:RDF>
```

Fig. 6. ARD for a temperature sensor

In the lookup for temperature sensors, the Resource User or the SQR sends an SPARQL query for resources which have an operation with an output of

[3] http://www.restlet.org
[4] http://jena.sourceforge.net
[5] http://www.w3.org/RDF
[6] http://www.w3.org/TR/rdf-sparql-query

type *om:Temperature* (see Figure 7) and the resource directory would find the temperature sensor described above.

```
SELECT ?description
WHERE {
?description ard:hasSemanticOperationDescription ?operation .
?operation ard:hasOutput ?output .
?output ard:hasInformationType ?infoType .
?infoType rdfs:subClassOf om:Temperature .
}
```

Fig. 7. SPARQL query for output temperature

There are more complex cases in which the SQR finds a resource in the resource directory that matches the output but it requests as input some information to calculate that output; this would be the case of a processor resource. Then, the SQR should post another query to the resource directory for resources that provide as output the required input to the processor. This recursive search would continue until all the resources do not need input information.

In the case of actuators, we use the post-conditions and pre-conditions of the operation to model the actuator as a resource. Figure 8 shows the SPARQL query for actuators able to switch on the light in the area [49.406115, 8.685196 - 49.406120, 8.685200], i.e. their postcondition should be light on. It could also be possible for the SQR to do a backward chaining by matching pre-conditions to post-conditions.

```
SELECT ?description
WHERE {
 ?description ard:hasSemanticOperationDescription ?operation .
 ?operation ard:hasPostCondition ?postcond .
 ?postcond rdf:type rdf:Statement .
 ?postcond rdf:subject om:LightStatus .
 ?postcond rdf:predicate om:hasValue .
 ?postcond rdf:object 'On' .
 ?description ard:hasLocation ?location .
 ?location loc:latitude ?lat .
 FILTER(?lat > '49.406115' && ?lat < '49.406120') .
 ?location loc:longitude ?long .
 FILTER (?long > '8.685196' && ?long < '8.685200') .
}
```

Fig. 8. SPARQL query for post-condition light on

5 State of the Art

For the modelling of resources, we did not find any existing approach that completely fit our purpose, as those approaches either did not allow to model what was needed or they were too general and therefore too complex for what we needed. However there are several industrial standards that aim at unifying sensor interfaces and data formats and are presented below.

IEEE 1451[7] is a suite of Smart Transducer Interface Standards, which key feature is the definition of Transducer Electronic Data Sheets (TEDS) that stores transducer identification, calibration, correction data, measurement range, and manufacture-related information. TEDS does not capture any derived semantics from sensor data. The ANSI provides a standard data format for Radiation

[7] http://ieee1451.nist.gov

Detectors used for homeland security specified in the ANSI N42.42 [3]. This standard does not address instrument control, data transmission protocols, or the physical media used for communications. CBRN-CCSI[8] (Common Chemical, Biological, Radiological, Nuclear (CBRN) Sensor Interface (CCSI)) is a standard for sensor physical and electronic interfaces. The standard characterizes sensors as packages of capabilities rather than what they detect. The main drawback of the CCSI is that the documentation is not intended to be publically available and most of the descriptions values have a very restrictive number of alternatives. The OGC Sensor Web Enablement (SWE) working group has created three different standards for defining encodings for describing sensors and sensor observations: Sensor Model Language (SensorML [4]), Transducer Markup Language (TransducerML [5]) and Observations & Measurements (O&M [6] [7]). Transducer Markup Language (TransducerML), is a XML based standard for specifying a standardized way to exchange raw or pre-processed sensor data. SensorML is defined in XML schemas and can, but generally does not provide a detailed description of the hardware design of a sensor. Rather it is a general schema for describing functional models of the sensor. Complementary to the SensorML is the Observations & Measurements (O&M) standard, since it provides a homogeneous way of representing the measures or observations taken by the sensors. In summary, we can conclude that SensorML provides a good framework to represent not just physical sensors, but also virtual, processors, sensor networks, mechanisms to derive high level information from raw sensor data; however, it lacks semantics about the sensors and the information they provide.

Several research efforts have been done to create a universal ontology to represent sensor information. In [8] Eid et al. presented a sensor ontology where the initial vocabulary used in the sensor domain was obtained from IEEE 1451 standard. In a more recent research [9], the ontology has been extended in order to adopt a standard upper ontology (SUMO) [10] to facilitate automatic data fusion and inference in distributed and heterogeneous sensing environments. However, the final ontology has not been published. Other researchers have created an OWL ontology called OntoSensor [11]. The defined ontology extends the IEEE SUMO concepts, and is based on the SensorML and ISO 19115 schemas. The main issue to construct the OntoSensor is to provide the semantics missing in SensorML.

Some work has been done to create ontologies to represent sensors as services. In [12], each sensor is associated with a semantic web service that contains the DAML-S description of the service, partitioned in three components: profile, process model and grounding. The main advantages of ontologies are related to the fact that they provide semantics and possibilities to derive information from the existing one. Most work that has been done in this area refers to application specific ontologies or little information about the ontology is actually published. The recently formed W3C Semantic Sensor Networks Incubator Group

[8] http://www.jpeocbd.osd.mil

(SSN-XG)[9] aims to build a general and expressive ontology for sensors to address the coverage, structural and expressivity issues. Several SENSEI members are also providing inputs to this work.

In SENSEI framework, we have developed a unified resource view. A resource can be a sensor, an actuator or a processor. It is any unit which can provide information or perform actuation tasks. With this unified resource view, it is not enough to have a model which describes only sensor or actuator. Our model can be used to describe a wide range different resources. It provides both basic description which can be used for simple identification, indexing and searching of a resource and advanced description which gives more detailed information about a resource. The advanced description can also be customized for different types of resources. Another advantage of our model is the support of interface description. None of existing model describes the access interface. Since the vision of the SENSEI framework is to support resources of different properties and capabilities, it is not feasible to specify one fixed interface for all resources. Instead, each resource can specify the access interface in the resource model. Thus, our model is versatile and future proof.

6 Conclusion

In this paper we have presented a resource model that semantically describes sensor, actuator and processing resources. The model is designed to efficiently support information queries and execute application requests. Here the main challenge is to efficiently select and compose relevant heterogeneous resources from a potentially large set. Heterogeneity is addressed by providing a semantic description of resource operations. Scalability is facilitated by providing a description that is sufficiently rich for selecting resources. For instance, we semantically describe the information a processing resources requires and provide information about its location and observation or actuation area of a resource. To our knowledge there is no existing work that describes sensor, actuators and resources in a comparable way, i.e. by describing syntactically and semantically the interface or operations offered by the resource in a uniform manner.

We have evaluated our model by showing the benefits in several use cases, showing that a semantic description of sensors, actuators and processing resources allows its lookup on a large scale system. In a later step we will evaluate the Resource Model for the automatic composition of resources and the automatic recursive resource lookup.

Acknowledgment

This paper describes work undertaken in the context of the SENSEI project, "Integrating the Physical with the Digital World of the Network of the Future" (www.sensei-project.eu). SENSEI is a Large Scale Collaborative Project supported by the European 7th Framework Programme, contract number: 215923.

[9] http://www.w3.org/2005/Incubator/ssn/charter

References

1. Villalonga, C., Bauer, M., Huang, V., Bernat, J., Barnaghi, P.: Modeling of sensor data and context for the real world internet. In: Proceedings PerCom 2010 Workshops. IEEE Computer Society Press, Los Alamitos (2010)
2. Fielding, R.T., Software, D., Taylor, R.N.: Principled design of the modern web architecture. ACM Transactions on Internet Technology 2, 115–150 (2002)
3. ANSI N42.42: American national standard data format standard for radiation detectors used for homeland security (2006)
4. OGC®: OpenGIS®Sensor model language (SensorML) implementation specification. version 1.0.0. OGC®reference number 07-000, http://www.opengeospatial.org/projects/groups/sensorweb
5. OGC®: OpenGIS®Transducer markup language (TML) implementation specification. version 1.0.0. OGC®reference number 06-010r6, http://www.opengeospatial.org/projects/groups/sensorweb
6. OGC®: OpenGIS®Observations and measurements -part 1 - observaton schema. version 1.0. OGC®reference number 07-022r1, http://www.opengeospatial.org/projects/groups/sensorweb
7. OGC®: OpenGIS®Observations and measurements -part 2 - sampling features. version 1.0. OGC®reference number 07-022r3, http://www.opengeospatial.org/projects/groups/sensorweb
8. Eid, M., Liscano, R., Saddik, A.E.: A novel ontology for sensor networks data. In: Proceedings of 2006 IEEE International Conference on Computational Intelligence for Measurement Systems and Applications, La Coruna, Spain (2006)
9. Eid, M., Liscano, R., Saddik, A.E.: A universal ontology for sensor networks data. In: Proceedings of 2007 IEEE International Conference on Computational Intelligence for Measurement Systems and Applications, Ostuni, Italy (2007)
10. Niles, I., Pease, A.: Origins of the standard upper merged ontology: A proposal for the IEEE standard upper ontology. In: Working Notes of the IJCAI 2001 Workshop on the IEEE Standard Upper Ontology, Seattle, WA (2001)
11. Russomanno, D., Kothari, C., Thomas, O.: Building a sensor ontology: A practical approach leveraging iso and ogc models. In: The 2005 International Conference on Artificial Intelligence, Las Vegas, NV, pp. 637–643 (2005)
12. Cybenko, G.V., Jiang, G., Chung, W.: Semantic agent technologies for tactical sensor networks. In: Proceedings of the SPIE Conference on Unattended Ground Sensor Technologies and Applications, Orlando, FL, pp. 311–320 (2003)

Platforms for AAL Applications

Gunnar Fagerberg[1], Antonio Kung[2], Reiner Wichert[3], Mohammad-Reza Tazari[3], Bruno Jean-Bart[2], Gerald Bauer[4], Gottfried Zimmermann[5], Francesco Furfari[6], F. Potortì[6], S. Chessa[6], M. Hellenschmidt[3], J. Gorman[7], Jan Alexandersson[8], Jürgen Bund[9], Eduardo Carrasco[9], Gorka Epelde[10], Martin Klima[11], Elena Urdaneta[12], Gregg Vanderheiden[13], and Ingo Zinnikus[8]

[1] Swedish Institute of Assistive Technology, PO Box 510, SE-162 15 Vällingby, Sweden
[2] Trialog, 25 rue du Général Foy, 75008, Paris, France
[3] Fraunhofer Ambient Assisted Living Alliance, c/o Fraunhofer-Institut für Graphische Datenverarbeitung, Fraunhoferstr. 5, 64283 Darmstadt, Germany
[4] Embedded Systems Lab, University of Passau, Innstraße 43, 94032 Passau, Germany
[5] Eberhard Karls Universität Tübingen, Medieninformatik, Sand 14, 72076 Tübingen, Germany
[6] ISTI-CNR, Pisa, Italy
[7] ItalySintef, Trondheim, Norway
[8] DFKI GmbH, Saarbrücken, Germany
[9] Meticube Lda, Portugal
[10] VICOMTech, Spain
[11] Czech Technical University, Czech Republic
[12] INGEMA – Instituto Gerontologico Matia, Spain
[13] TRACE R&D Center, University of Wisconsin-Madison, USA

gunnar.fagerberg@hi.se, antonio.kung@trialog.com, {Reiner.Wichert,Saied.Tazari, michael.hellenschmidt}@igd.fraunhofer.de, bruno.jean-bart@trialog.com, gerald.bauer@uni-passau.de, gottfried.zimmermann@uni-tuebingen.de, {francesco.furfari, potorti,stefano.chessa}@isti.cnr.it, joe.gorman@sintef.no

Keywords: AmI, AAL, Platform, OSGi, URC, Smart Homes, Healthcare, Software Platform, Open Source Policies, Standards, reference platform, standardization, Platform, Application, Policies, Universal Remote Console (URC), Web Service Description Language (WSDL), personal user interface, pluggable user interface, OpenURC, Alliance, Platform for Accessible User Interfaces, ISO, ICT, e-Inclusion.

1 Introduction

AAL – Ambient Assisted Living. When the topic "AAL" comes up, everybody thinks first of fascinating technology that is unobtrusively integrated in "smart home environments" and supports people in their daily lives. This thinking is surely be driven by the ongoing development of impressive services that was and is still provided by the AAL research community as well as the growing number of companies that bring these technology to market.

P. Lukowitz, G. Kortuem, and K. Kunze (Eds.): EuroSSC 2010, LNCS 6446, pp. 177–201, 2010.

In many projects - like MonAMI[14] - that deal with the integration of such mainstreaming services and sensor systems into a common architecture, it quickly became clear that a very important aspect was not sufficient considered: The need of a common platform for AAL systems.

This paper is a summary of articles addressing this issue. Twenty highly qualified experts dealing with the standardization of AAL platforms for many years present their experiences to the topics like “Why platforms are important to AAL”, “What features should be in a AAL platform” and “Why mainstream and open source solutions are important in AAL”.

2 Europe Support for AAL Systems

(Gunnar Fagerberg and Antonio Kung)

This chapter points out the importance of AAL platform in the action plan carried out by the European commission on ICT and Ageing.

2.1 ICT and Demographic Ageing

Ambient Assisted Living (AAL) systems refer to “intelligent systems that will assist elderly individuals for a better, healthier and safer life in the preferred living environment and covers concepts, products and services that interlink and improve new technologies and the social environment” [1]. AAL systems address socio-political issues created by the demographic development in Europe. Care and assistance needs to elderly persons will drastically increase in the future with a situation in 2050 where there will be two retired persons for one working person [2].

AAL systems can help support older people in various application domains [1]: at work, as persons will remain active and productive for a longer time, with an improved quality of work; in the community, by staying socially active and creative through ICT solutions; at home, byenjoying a healthier and higher quality of daily life for a longer time, assisted by technology, while maintaining a high degree of independence, autonomy and dignity.

AAL systems are not only a social necessity but also an economic opportunity. The wealth and revenues of persons over 65 in Europe is over 3,000 billion euro.

2.2 eInclusion in the Digital Agenda for Europe

eInclusion and therefore AAL systems are an integral part of the digital agenda for Europe [5]. Further to the 2006 Riga ministerial declaration on e-Inclusion policy [3], the European Commission has defined an ageing well action plan [4] as well as a series of measures that involve more than one billion euro in research and development between 2006 and 2013: the framework 7 program [6] help fund longer-term R&D; the AAL joint program [7] is dedicated to market oriented R&D; and finally the Competitiveness and Innovation framework program (CIP) within the ICT Policy Support Programme [8] supports initiatives with deployment priorities.

2.3 AAL Platforms

Initiatives funded with the help of the European commission in the framework 7 cover a number of area: mobility/falls, cognitive support, activities of daily light support, service and social robotics, and open platform and tools. In the latter area, projects such as Oasis, Persona, I2Home, MPower, OASIS, MonAMI have taken place, with various specific objectives. UniversAAL [9] is a recent undertaking which has the objective to integrate the various features developed in the previous projects and to make available to the R&D community an unified platform.

It is expected that this will be the starting point to an R&D ecosystem which will promote the development of the needed ICT applications for ageing, at work, in the community and at home.

3 Research Vision for AAL Platforms in Germany

(Reiner Wichert and Mohammad-Reza Tazari)

Despite the huge market potential for AAL products and services, the AAL branch is still on the cusp of a mainstream breakthrough. A lack of viable business models is considered almost consentaneously to be the main market obstacle to a broad implementation of innovative AAL systems. This short paper describes the pressing necessity to agree on a common AAL platform and the way how this could be achieved.

3.1 Establish a Joint AmI and AAL Community

The AAL-vision incarnated that one day sensors and systems give seniors a helping hand in their own home, e.g. by measuring, monitoring, and raising alarms if necessary. A lot of scientific research have been done in the past years to achieve this objective. Numerous projects has resulted in a considerable amount of applications and product concepts which ended up with a variety of prototypical isolated solutions [10]. Unfortunately the industry is still waiting to step into the huge market potential. Thus, the main question is now about the reasons for that and how we can motivate the industry to open their range of products to the AAL market.

A lack of business models as the foundation for a cooperation between developers, service providers, medical device manufacturers and the housing industry is almost unanimously named as the greatest marketing hindrance to a broad implementation of innovative assisted living systems. The high costs of the isolated solutions are the cause of this deficit which was highlighted in a very unfavourable cost-effect ratio in a BMBF-sponsored study of AAL market potential and development opportunities [11]. Here e.g. for cardiac insufficiency costs of 17,300 Euros per extended year of life expectancy were calculated.

Taking health assistance as an example, unfortunately older individuals usually have not only one medical illness, but rather are confronted with numerous symptoms and medical requirements. Only when covering the aging person's entire clinical picture the costs can be minimised as then the people can remain longer at home and the unallocated funds gained from postponing nursing home stays can compensate the investments in AAL solutions. For this purpose, isolated solutions must become part of an overall solution and share resources. However, as the data exchange formats and protocols of these singular solutions are incompatible, the components of one application cannot be used by another without modification. Alterations by an expert system integrator are required to combine new and existing partial solutions, thereby making the total end solution prohibitively expensive [12]. Additionally, isolated solutions disregard possible overlaps so that share-able resources must be replicated and paid for repeatedly as the respective systems are only offered as complete packages. This results in an increase of expenses for private buyers, in long term even for care providers and health insurance companies.

As a consequence, research in Germany is coming to the conclusion that future AAL solutions must be based on few flexible platforms with ascertainable market shares that allow modular expansion, self-integration into the environment, and customisation to the individual's needs and health curve. Various AAL platforms have already been developed and largely validated within several projects, such as Amigo, PERSONA, and SOPRANO. They mostly focus on dynamic distributed infrastructures for the self-organisation of devices, sensors and services. Unfortunately, these solutions have not lead to the desired breakthrough, although each possesses substantial advantages as an AAL infrastructure. For this reason, the Eurpean Commission decided to fund a last research project within the current framework program, with the vision of a common Open Source platform for AAL. The research project selected was universal [13] which began in February this year with the goal to consolidate the state-of-the-art towards such a platform based on the most attractive middleware platforms of the recent years. Also German researchers are involved in this initiative. We believe that the consequent openness chosen by universAAL is the main means that can help to go beyond all boundaries, be it organizational, national or even EU. This openness isintended to include all aspects, from work processes to work results based on a permissive open source license model, from opening early results for discussion to incorporating external contributions until handing over the maintenance of the platform to an established AAL community after the project end. However, there is also a clear need to convince the technological industry of this vision. Only when the industry is ready to develop products and services based on a common platform, AmI and AAL will get a real chance for having success.

4 Building an Open Platform: The MonAMI Experience
(Bruno Jean-Bart and Gerald Bauer)

This chapter presents experiences and challenges encountered during the design of a new and open AAL platform based the OSGi and URC standards for the MonAMI FP6 project. The open platform is used to provide various services that help elderly and disabled people to manage daily tasks more independently. We describe the creation of an AAL system that includes more than twenty services developed by different service providers in a common system. The benefit of the MonAMI architecture is that the system can be easily enriched with new "smart" home services and "accessible" user interfaces without a deep knowledge of the underlying sensors, network, and services.

4.1 Introduction: The MonAMI Experience

Mainstreaming on Ambient Intelligence (MonAMI) [14] is an FP6 EU project that started in September 2006. Its overall objective is to investigate, specify, validate and promote an approach based on mainstream technologies whereby Ambient Assisted Living (AAL) applications can be deployed in a cost effective manner. One of the main objectives is to build an architecture that matches the needs of end users by providing useful and easy configurable services, a reliable and secure architecture that can be easily accessed or extended, and affordable hardware. The resulting architecture is depicted in the Figure 1 below.

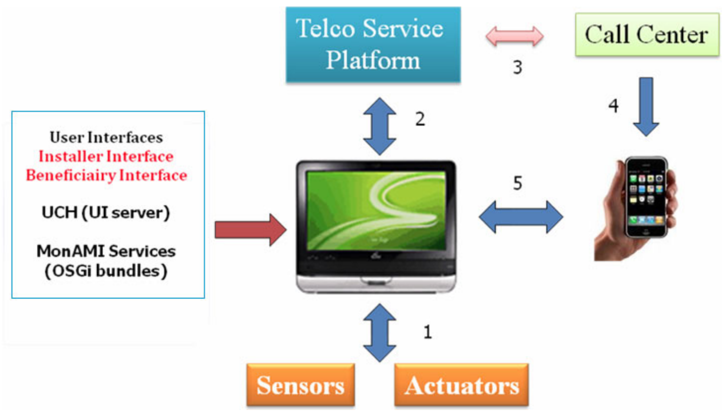


Fig. 1. The MonAMI Architecture

The equipment installed at home consists of a standard touch screen PC which is equipped with Wi-Fi and connected to a broadband Internet network. The PC displays simple and accessible web pages which enable the user to interact with MonAMI applications or services. These services are grouped into five packages:

- *AMICASA*: services for the remote control of lights, shutters, doors, etc. It also includes a set of automatic services, e.g. lights are switched on when a room becomes dark. The service *I go to bed* may include actions to switch off the lights in the living room and switch on the lights in the bedroom.
- *AMISURE*: services that improve personal safety, e.g. when an appliance remains on longer than expected, an alarm informs the user and the care center.
- *AMIVUE*: services which provide the means to observe the ambient environment of the flat. It can be used by the care giver to remotely check the current situation.
- *AMIPAL*: reminder services to inform the person about the meetings or actions to be taken on a given day (e.g. taking pills).
- *AMIPLAY*: a set of accessible and simple games to enjoy when the user is alone.

MonAMI is currently preparing the deployment of the implemented applications in a field trial involving 55 homes in Slovakia, Spain and Sweden in order to assess the usability and scalability aspects of the implemented services.

4.2 The MonAMI Architecture: An Open AAL Platform

The major goal for the MonAMI architecture is to identify the features which simplify the business environment for developing AAL services, which will in turn foster the creation of a business ecosystem. The business environment includes the stakeholders and decision makers involved in the development and operations of AAL services. Among the features identified, the most decisive one is the use of a set of common interfaces, called OSGI4AMI, in the home platform in order to create a separation between services and the rest of the environment (e.g. web server, home devices, user interface). As a result, the business environment for developing services is simplified. Figure 2 shows how the MonAMI implementation supports those features by using Java interfaces to connect bundles.

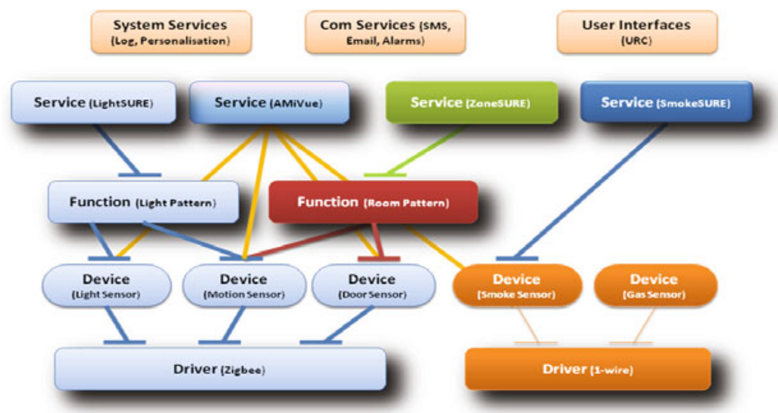


Fig. 2. MonAMI bundles connected via OSGi4AMI interfaces

The home platform is based on the OSGi framework [16]. OSGi enables a Java-based system to be remotely managed and is currently used in most smart home environment applications. OSGI4AMI [18] is based on a comprehensive ontology of devices and services which is mapped into a set of Java interfaces. The resulting architecture has the following advantages:

- it creates a clear separation between applications and the set of devices (e.g. sensors, actuators) used in MonAMI. Programmers of services do not have to adapt their code to specific devices and technologies.
- it includes facilities to configure and personalize the services,
- it enables the development of a rich set of AAL services based on standard interfaces which includes sensors, monitoring (e.g. temperature patterns), communication (e.g. SMS, email), and a generic connection to user interfaces.

Besides service applications, the user interface is another important and required feature. MonAMI services must be locally accessible to elderly and disabled people as well as remotely available to caregivers and relatives. In order to implement these requirements, MonAMI uses a second framework based on URC (Universal Remote Console) [17] that was developed by the I2Home project and fully meets the above requirements. Again, URC enables a complete separation between the service contents and the way of presenting information to the beneficiary of the service. The user interface can be changed, adapted, or personalized to the user without changing the contents of the applications.

4.3 An Ontology Mapped to OSGi : OSGi4AMI

An ontology is a representation of system elements and their relationships. In MonAMI, an ontology was developed to describe the devices and the basic AAL services. The devices and services definition form a set of interfaces used both for the bundle interactions and for user interfaces.

The result of this approach is called OSGI4AMI. All Java interfaces are included in one OSGi bundle that is imported by all other devices, drivers, and service bundles. At the bottom of the architecture in Figure 2, the driver plays a major role as it manages the links with devices. When a device is connected to the local network, the driver creates a new OSGi service (a virtual device), with the class name of the corresponding OSGi4AMI interface (e.g., *TemperatureSensor*, *LightActuator*).

This virtual device may be used by any higher level bundle, such as *LightPattern* or *AMIVue*. The reference object of this virtual device, a light sensor for instance, can implement the *LightSensor* interface as well as other interfaces of the ontology. For example, the *Configurable* interface makes the virtual device configurable, and the *Generic Communication Service* interface makes the device accessible from a remote client such as the user interface.

As a result of the architecture described above, the MonAMI platform enables service providers to develop innovative AAL services to help disabled and elderly persons live at home more easily. This straightforward approach makes it easy to design a component that is compatible with the MonAMI infrastructure. A service developer simply needs to know the OSGi4AMI ontology to use existing devices, the AAL basic service specification, and the user interfaces.

4.4 Accessible User Interfaces

The MonAMI service platform makes a clear separation between the two main functional blocks shown in Figure 3. The Human Machine Interfaces (HMI) block runs on a Universal Control Hub (UCH). The other block consists of AAL services which run on an OSGi framework. In the terminology of UCH, the AAL services are seen as a *multipletarget.* The UCH is an HMI server which implements the ANSI standard Universal Remote Console (URC). The UCH enables the development and execution of multiple, accessible, and flexible HMI clients on different targets such as a touch screen computer or an iPhone which are tailored to the user's needs (e.g., an elderly person or caregiver) as seen in Figure 3.

Fig. 3. OSGI4AMI is used both for user interface and for the service definition

4.5 Conclusion

This paper introduces an open AAL platform that was designed by the MonAMI consortium. It shows how a full, functional smart home environment based on different mainstream technology was integrated in a common system. The approach based on the ontology of OSGi4AMI interfaces has proved to be efficient during service development, in which more than five team members worked on the development of over twenty MonAMI services. OSGi4AMI helped to develop and to integrate efficient new AAL services and to adapt or extend already existing services. OSGi4AMI could be the foundation of a standard for AAL, as it can be easily extended whenever new services need to be created. It could lead to the creation of a central AAL web store where new services or drivers could be downloaded on home gateways, in the same way as new applications are easily downloaded from Apple's App Store.

5 URC and WSDL – Towards Personal User Interfaces for Web Services *(Gottfried Zimmermann)*

This chapter introduces a concept of applying the Universal Remote Console (URC) framework to Web services, in particular to those described by the Web Service Description Language (WSDL). A new standardization effort is currently initiated under ISO/IEC JTC 1, Subcommittee SC 35, to extend the URC series of standards by a part on Web service integration. Both developers and users of Ambient Intelligence environments (including Ambient Assisted Living environments) will benefit: Developers will harness the power of Web services and pertaining development tools for the development of intelligent services, and the users can use the services through their preferred personal user interfaces.

5.1 Introduction

Web services are becoming increasingly ubiquitous in the form of public Internet-wide services and private services in protected environments, including Ambient Assisted Living applications. Even devices and appliances in the digital home are being made network-accessible by exposing them as Web services.

The Universal Remote Console (URC) framework, as defined in ISO/IEC 24752 [19], defines a mechanism for personal user interfaces that “plug” into networked applications of any kind. Thus a user can use any controller and user interface of their choice to access and control the applications. The applicability and usefulness of the URC technology has been demonstrated in various projects, including the Trace Center’s MyURC open-source projects [20], i2home [21], VITAL [22] and MonAMI [23].

For a Web service to adopt the URC concepts, it needs to expose its Web service interface as a composition of user interface socket elements. Both the User Interface Socket Description language and the Web Service Description Language (WSDL) [24] define suitable extension mechanisms for such integration.

It is expected that this approach will help adoption of the URC technology for Web services, and thus make personalized and pluggable user interfaces widely available for Web services.

5.2 Technical Work

A new work item proposal is currently being voted on in the Subcommittee SC 35 of ISO/IEC JTC 1. It proposes to add a part 6 to the multi-part standard ISO/IEC 24752, Universal Remote Console. Part 6 will define "syntax and semantics of mapping information that describes how the elements of the socket are to be mapped to the elements of a Web service interface. Such mapping information can be included either within a user interface socket description, as a separate mapping document, or within a Web service description language (WSDL) document."

This alignment of the URC technology to Web services gives rise to a variety of possible implementation scenarios, depending on whether the mapping happens on the Web service, on a middleware component, or on the URC.

Scenario 1. The Web service offers a networking interface for access to the socket, in addition to the Web service operations it provides (fig. 4). Such a socket interface could implement the URC-HTTP protocol [25] or any other protocol based on ISO/IEC 24752-2. Thus a URC can connect to the user interface socket on the Web service, and retrieve a user interface from the Web service or from any resource server. It should be noted that the socket interface could be expressed in terms of a WSDL interface with generic operations for accessing the socket elements, but this would still require an additional description of the socket elements, their dependencies and other properties.

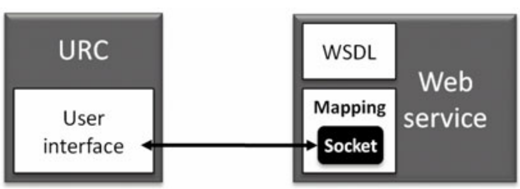


Fig. 4. Mapping resides on Web service

Scenario 2. There is a middleware component that translates the WSDL operations of the Web service into a socket interface (fig. 5). Also, the middleware can provide alternate user interface based on the socket interface for any controller to use. The user interfaces come from a local repository, or are retrieved from any resource server. Inthis case neither the Web service nor the controller need any knowledge of the URC framework. An implementation of such a middleware exists in the Universal Control Hub (UCH) [26].

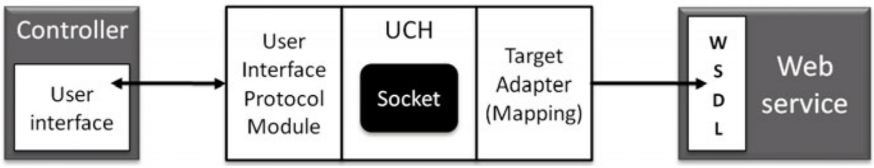


Fig. 5. Mapping resides on middleware (UCH)

Scenario 3. A URC connects to the Web service via the service's native operations, as specified in the WSDL document (fig. 6). It then maps these operations into a socket, thus exposing socket elements to any user interface running on the URC platform. User interfaces are either stored on the URC, or come from any resource server.

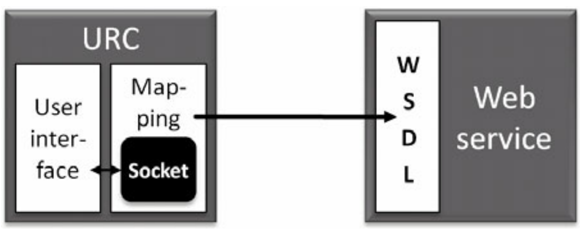


Fig. 6. Mapping resides on URC

5.3 Expected Benefits and Risks

A language for mapping between a Web service interface and a socket will close a gap and make the URC and Web service worlds more interoperable. Developers of Web services, including services in the context of AAL, will be able to harness the power of the URC framework to provide for personal user interfaces. The development of a standard will also spur the development of appropriate development tools.

Users of Web services will benefit as well, since they will be able to operate Web services with the device and user interface of their choice. Third parties will be able to provide supplemental user interfaces for existing Web services, thus making them more accessible and usable for particular user groups that have not been catered for by the Web service provider.

It is expected that any Web service will be able to present URC-based user interfaces in the manner described in this paper, if the developers think about its abstract user interface (socket) early in the development. There are, however, some

Web services already in use that will need a more complex mapping than can be expressed by a declarative language such as being developed within ISO/IEC 24752-6. In these cases the mapping will have to be implemented as manually written code, in order to make URC-based user interfaces available for these Web services. The mapping code will preferably run either on the Web service itself or on a middleware component (e.g. as target adapter of the UCH).

Acknowledgments. This work has been funded by the US Department of Education, NIDRR, under Grant H133E030012 (RERC on IT Access); and by the EU 6th Framework Program under grant FP6-033502 (i2home). The opinions herein are those of the authors and not necessarily those of the funding agencies.

6 Challenges for the Building of AAL Platforms

(Antonio Kung and Francesco Furfari)

AAL systems are ambient intelligent systems which need dedicated platforms because they have to support domain specific features and interfaces. This paper presents the service/platform interface approach, lists challenges to be solved, and suggests measures to facilitate the development of AAL platforms.

6.1 Need for an AAL Platform

A platform can be defined as a computing architecture including software and hardware that serves as a foundation to application programmers. Since it separates specific features implemented by a small community of system experts from application features implemented by a larger community of developers, it allows much faster development of applications.

Several initiatives specific to Ambient Assisted Living (AAL) have been launched to address the issue of creating service platforms, such as Soprano [27], Persona [28], MPower [29], MonAMI [14], OASIS [31], I2Home [32] or UniversAAL [33] projects. This paper briefly presents how the contributions on platforms from these initiatives can be harmonized.

6.2 The Service-Platform Interface

All initiatives mentioned above have converged towards home server with one clear interface: the service/platform interface. This single interface, if properly defined, could facilitate the transformation of the currently fragmented markets into a single unified market. It would also facilitate user centered design since application development iterations could be made possible by allowing faster implementation and modification of applications. Finally, it would ensure that applications would be technology independent. Note that the term interface is used generically in singular. In practice there might be many interfaces.

Harmonizing these initiatives does not mean imposing a single agreed platform. It means reaching an agreement concerning the interface between services and platforms so that various platform solutions can compete. This agreement is AAL specific, but to our knowledge there is not yet any discussion on the definition of common overarching interfaces which would enforce a clear separation between stakeholders developing applications and stakeholders developing platform elements. We believe this is a prerequisite for an AAL ecosystem.

6.3 Challenges

We cover now a number of key challenges that have to be addressed: the integration of innovation in platforms, the integration of transversal features, interoperability of interfaces, technology independence, and finally the support of multiple business models.

6.3.1 Integrating Innovation

While AAL platforms should be based on widely used technologies, they do include features that are specific and novel. The AAliance research agenda [34] provides an overview of such challenges in different contexts (at home, within a community, at work). It identifies technology challenges related to sensing, reasoning, acting, interacting, and communicating. Research is on-going, but results must be integrated in platforms. Thus the challenge is to maintain and coordinate two threads of development activities: on one hand the integration of research features into a platform, and on the other hand the integration of components in a platform that meets industry requirements.

6.3.2 Integrating Transversal Features

Future AAL platforms should take into account features that are transversal. Scalability issues will arise when millions of platforms are deployed and personalised. Quality of service issues will arise for specific applications with different dependability requirements, e.g., related to health monitoring or alarms. Liability issues will arise when the chain of applications and technology has not worked properly. Confidentiality and data protection issues could have negative social consequences. Solving these issues may have far reaching effects on application and platform design.

6.3.3 Interoperability

Interoperability is not easy to reach because it relates to many domain specific aspects. The Interoperability working group of the German BMBF/VDE Innovation Partnership on AAL [35] included experts from the following domains: Home/building automation, medical IT, household appliances, consumer electronics, network technology, AAL middleware platforms, sensor networks, robotics, cognitive systems, standardisation, and certification.

Furthermore, any attempt for standardisation will involve dedicated task forces involving the various competing stakeholders in a given domain. These working groups will focus on competitive issues, i.e., what is mandatory, what is optional, and what is proprietary, as well as on extensions issues, i.e., the impact of new features on interoperability. The definition of the CHAIN standard [36] is a good example. This group of more than 10 white goods manufacturers had to reach a consensus, taking into account competitive and extensions issues.

The AALiance project has also identified areas where there is no standard: “Remote Maintenance of AAL Systems, Connecting to Medical Sensors, AAL

Terminology, vocabularies, ontologies, AAL Middleware / Service Execution Environment, Emergency Calls, and Connection to Call Centres" [37, 38].

There are two approaches for successful standard adoption. Either we have industry-led initiatives with well defined but focused and narrower objectives, such as the Continua Health Alliance [39], or we have an overall consensus between all stakeholders within an application domain and technology stakeholders responsible for a given platform. We believe that this second approach is what is missing for AAL platforms.

6.3.4 Technology Independence

Technology independence serves indirect but important purposes. First of all, if there are several competing platforms it allows application stakeholders to switch from one platform to another without modifying applications. Secondly, it allows platform stakeholders to switch from one technology to another, again without modifying applications (for instance to another RF network solution).

The challenge is to specify interfaces in a technology independent manner. The CHAIN specification [36] made sure that interoperability specifications would consist of three parts: 1) a network technology-independent specification with semantic interoperability (e.g. the "start appliance" message is technology independent), 2) a network technology-independent specification of syntactic interoperability (e.g. the "start appliance" message is represented by value 6), and finally 3) a network-dependent mapping (e.g. the message is mapped on a Konnex network). A working group in CENELEC is working on IFRS, an Interoperability Framework Requirements Specification which provides the "language" for expressing semantic interoperability which is independent from the underlying network standard [35]. Such a framework could be used in the future for technology independent specifications.

6.3.5 Multiple Business Models

There could be several business models depending on national/regional specificities. In some countries, AAL applications could be subsidised at the public level (e.g., social security), and in other countries this could be at the private level (e.g., an insurance company). Local policies have an impact on the way AAL applications and platforms are procured and operated. The resulting different business models could have an impact on the service/platform interface requirements. For instance different technology features would be needed depending on whether sharing data between different domains (e.g. care versus health) is possible or not. The challenge therefore is to ensure that the specified standards and interfaces be compatible with the various operating models. This could require configuration capability to select various AAL platform profiles.

6.4 Conclusion

To make such platforms a reality, we suggest two measures: first, the launching of a research and industry initiative for an open source platform [42, 43], and secondly, the launching of an EC policy supported long-term process for the definition of AAL solutions and standards.

We would like to acknowledge the work and contribution of the MonAMI and UniversAAL partners, as well as the positive feedback obtained through the various interactions with the AAL community and the European Commission. MonAMI and UniversAAL are partially funded by the European Commission under the 6th and 7th framework programs.

7 The AAL Open Association Manifesto

(F. Furfari, F. Potortì, S. Chessa, M.-R. Tazari, M. Hellenschmidt, R. Wichert, J. Gorman and Antonio Kung)

AAL (Ambient Assisted Living) has great potential for positively influencing the lives of many people. But impact so far has been less than hoped, partly due to fragmentation of research efforts and the lack of a standardised approach for developers. To address this, we are forming the AALOA (AAL Open Association), and invite you to join in our efforts.

7.1 AAL - Promising But Problematic

The abbreviation "AAL" stands for Ambient Assisted Living [44] and is about making smart use of technology to support well-being in the preferred living environment for people who might otherwise find this difficult (e.g. infirm or very elderly people who want to continue living in their own homes). Research in the area is motivated by socio-political issues of the ageing population, and offers a promising approach with potentially wide-reaching benefits. It involves many ICT-related R&D disciplines in an application field that has attracted much attention. Several initiatives have emerged to tackle the challenges involved [45][46][47], and significant incremental progress has been achieved on many fronts. But a major AAL breakthrough, leading to a standardized approach and thereby to widespread adoption, is still not in sight. A way of doing things that has general acceptance and can almost be assumed, like the Apache Server is in the web world, is missing in the world of AAL.

Why have there been no AAL breakthroughs? From an R&D perspective, part of the answer is to be found in fragmentation of research efforts in the area of AmI (*Ambient Intelligence* [48] - also referred to as Ubiquitous and Pervasive Computing [49][50]). AmI is the key research discipline that underpins the domain of AAL, and many innovative ideas and approaches have emerged from research projects, conferences etc. in recent years. The field has matured over time – but so far with no converging conclusions.

From a market perspective, there are two obstacles. The first arises from the lack of technical convergence: this leads to development of very different technical solutions that are difficult to compare, so there is no baseline against which to assess user experiences in the types of scenarios envisaged by AmI. It's hard to market something whose benefits you can't clearly quantify. The second obstacle is market fragmentation. The whole concept of "ambience" is all about making use of everything around you as part of a single overall solution. But today's commercial reality is that the growing number and types of devices around us (mobile phones, home theatres, games consoles, media servers, home gateways etc.) are treated as separate market segments – even though the devices themselves have the potential to interact. A paradigm shift is needed, but who will risk the investments and changes in business models needed in the absence of a precise model adopted by a large ecosystem of artifacts?

The concept of *co-opetiton*[51] - collaboration among competitors - has been put forward as a way to achieve commoditized infrastructures and been successfully deployed in some cases. But for there to be any chance of a real paradigm shift, a transversal cooperation over diverse market segments with the involvement of many stakeholders is needed. That is one of the key things that the AALOA aims to achieve.

7.2 AALOA – An Open Association Promoting AAL Research, Development Uptake and Impact

The signatories of this manifesto consider that the time has come to do something about the problems hindering progress in the area of AAL. We believe that this is something that transcends individual projects or organizations, and needs a long-term approach, with broad involvement from all types of stakeholders. This manifesto is intended as an invitation to join us in our mission, which is to:

- Bring together the resources, tools and people involved in AAL in a single forum that makes it much easier to reach conclusions on provisions needed to achieve AAL progress;
- Make sure that all technology providers, service providers and research institutions involved in AAL are either directly involved in AALOA or (as a minimum) aware of decisions it promotes;
- Involve end-user representatives in all work of AALOA;
- Identify key research topics in AAL, and reach agreement on prioritization of these;
- Design, develop, evaluate, standardize and maintain a common service platform for AAL.

Our mission is founded on a long-term technical vision. This will evolve over time, but gives an indication at the initiation stage of the direction in which we want to go. In our vision, ordinary hardware resources such as displays, keyboards and storage devices that nowadays need drivers integrated into Operating Systems (OS) will evolve into pluggable networked resources. We foresee the emergence of new programming languages, based on resource and service discovery paradigms, facilitating the development of AmI applications.

There will be a shift away from the idea of developing applications that run on different PCs and OSs towards the concept of developing applications for “AAL spaces”. Middleware [52] will be widely used, and help developers to identify the features available in the environment (sensors, other devices, services) and write programs which can exploit large classes of them effectively, without needing to know their actual whereabouts or be concerned with low-level configuration details.

This will involve more than just developing pluggable components: it will mean that developers will effectively be able to contribute to several distributed applications - without even knowing all of them beforehand. “AAL Spaces” will become the equivalent of today’s PCs (in terms of widespread availability, standardization and acceptance) and new markets will emerge for software and hardware products, involving houses, cars, airports, hospitals and public spaces..

7.2.1 Getting Started: Defining a Reference Architecture

The hardware specification of the original IBM PC of the eighties, when several independent manufacturers started to produce peripherals and compatible hardware thanks to the standardization of connector interfaces and the availability of specifications, was one of the key enablers that led to the ubiquity of PCs we know today.

One of the first tasks of the AALOA will be to do something similar for the AAL domain: define a *reference architecture* to standardize the resources available in AAL environments, and how to integrate them. This will encourage the creation of new brands and the coalition of firms around new business opportunities.

7.3 Your AALOA Needs YOU

To achieve our mission, and contribute to bringing about this long-term vision, the signatories of this manifesto started to incubate the AALOA – the Ambient Assisted Living Open Association. As its name suggests, anyone can join the AALOA, and this manifesto should be considered as a direct invitation to do so.

The AALOA can only achieve its mission if its membership represents a significant proportion of the people and organisations involved in AAL/AmI, in one way or another. We invite you to join the association, and to participate in its activities: to bring fresh ideas, to propose workshops and projects and to contribute actively to the growth of the association. For details of how to join, please visit the web site: <http://www.aaloa.org>

The detailed organisational structure of the AALOA is in the process of being formalised in a set of statutes. These are still under development, and people responding to the invitation to join will have the opportunity to influence their development.

We envision a not-for-profit organization, with two boards that nominate common elective offices: a Governing Board following common best practices of open source communities and an Advisory Board composed of industry and user communities. The latter will be organized into working groups whose role is to advise AALOA’s open source community about emerging technical and market challenges.The following challenges have to be addressed: the integration of innovation in platforms, the integration of transversal features, interoperability of interfaces, technology independence and finally the support of multiple business models.

7.3.1 The Open Source Policy

The importance of open source software in the industry has risen to prominence in recent years, especially in the development of software infrastructures. Closed, proprietary approaches become less attractive as standardised infrastructure software becomes a commodity: high development costs due to the complexity of such software, uncertainty due to the "winner-takes-all” effect and diminishing marginal returns make the market for infrastructure software a risky business. The open source approach, on the other hand, promises easier software maintenance, allows cooperation between competitors and helps spread production costs over a multiplicity of stakeholders [53] [54].

7.4 Call for Project Proposals

The association will be organized as a federation of projects, one representative of each project being a member of the Governing Board.

Proposals for new projects can be submitted to the Governing Board, whose main role will be their evaluation with respect to the association’s mission, while still encouraging the emergence of diversity, and avoiding monoculture. Projects will autonomously organize their governance rules. Over time common rules suggested by practice may be formally adopted.

As one of the association’s objectives involves building an open source community working on service platforms for AAL, projects related to software development are to be expected. But we emphasise that other types of projects are also welcome. The next section describes an example of one such.

We are setting up resources for building and managing projects. You can access these resources by submitting a project proposal with a list of individuals or organizations that support your project idea. Visit the web page at http://www.aaloa.org/projects for details.

7.4.1 The EvAAL International Competition

EvAAL has been the first project proposed to AALOA promoters and it is a paramount for the AALOA purposes. In fact, an important action for the assessment of the research results in this area is based on the analysis and comparison of the existing solutions provided by the research community [55]. To this end, we intend to promote an international competition called EvAAL (“Evaluating AAL Systems through Competitive Benchmarking”). The competition is intended to raise awareness of and interest in AAL, and to spread knowledge about the state-of-the art to a large audience. To do this, we will issue an annual “Call for Competition Ideas”, in which we will invite practitioners and experts to propose the topics and rules for that year’s competition. The idea received will be assessed and possibly merged, before the competition itself is announced. The competition itself will invite people to compete by developing hardware/software artefacts supporting the selected topic.

Generally, the competition will be organized around one or several of the functions enabling AAL spaces, such as:

- sensing
- reasoning
- acting
- interacting
- communicating

In order to stimulate the participation of PhD students, a cash prize will be awarded to the competition winner(s) each year. We would like this to be something significant, such as an amount equivalent to a research grant for one year at an international university. All participants in the contest will have the opportunity of publishing a peer-reviewed paper describing their system. For details about the contest please visit the EvAAL web site at http://evaal.aaloa.org

7.5 Acknowledgments

The idea of forming the association arose from discussions between some of the institutions involved in the projects PERSONA and universAAL, funded respectively in FP6 and FP7 (the Sixth and Seventh Framework Programme of the EU), but similar ideas were also discussed by partners of other projects who had recognized the need for a common effort in the field of AAL/AmI, as well. Today, the Manifesto is a dissemination effort of the EU projects MonAmI, OASIS, OsAmI-commons, PERSONA, SOPRANO, universAAL and WASP. The subscribers listed below are people who support the ideas promoted by the Manifesto and are willing to participate in the life of the association.

In addition to the subscribers, there are few promoting organizations (details to be found on the Web site) that have allocated resources for carrying out the tasks in the incubation phase of AALOA, until its bylaws are finalized and the association itself is established as a legal entity. Nevertheless, more effort and voluntary contribution is still needed. Hence, we encourage you, as the reader of this manifesto, to get involved in this open process!

7.6 Subscribers

Francesco Furfari, CNR-ISTI, Italy; **Francesco Potortì**, CNR-ISTI, Italy; **Stefano Chessa**, University of Pisa, Italy; **M. Tazari,**Fraunhofer-IGD, Germany; **Michael Hellenschmidt**, Fraunhofer-IGD, Germany; **Reiner Wichert**, Fraunhofer-IGD, Germany; **Joe Gorman**, SINTEF, Norway; **Sergio Gustavo GuillenBarrionuevo**, ITACA Univerity Polytechnic Valencia, Spain; **Juan Pablo Lázaro Ramos**, TSB, Spain; **Marius Mikalsen,** SINTEF, Norway; **Antonio Kung**, Trialog, France; **Bruno Jean-Bart**, Trialog France; **Gunnar.Fagerberg**, SIAT, Sweden; **SilvioBonfiglio**, FIMI, Italy; **Jesus Bermejo**, Telvent, Spain; **StenHanke**, AIT-HBS, Austria; **Andreas Hochgatterer**, AIT-HBS, Austria; **Michele Amoretti**, R&S Info, Italy; **Sergio Copelli**, R&S Info, Italy; **Richard Dapoigny**, University of Savoie, France; **Milan Petkvic,** Philips, The Netherlands; **Susan Schwarze**, Prosyst, Germany; **Kai Hackbarth**, Prosyst, Germany; **César Iglesias Díaz-Bastien**, dbtlex, Spain; **Peter Wolf**, FZI, Germany; **Andreas Schmidt**, FZI, Germany; **Armando Roy Delgado**, University of Zaragoza, Spain; **Anastasia Garbi**, Exodussa, Greece; **Elena Avatangelou**, Exodussa, Greece; **MartijnBennebroek**, Philips, The Netherlands; **François Letellier**, OW2, France; **Thomas Karopka**, IT Science Center RügengGmbH, Germany; **VicençSoler**, University of Barcellona, Spain; **Ricardo Serafin**, TSB, Spain; **Paolo Bellavista**, University of Bologna, Italy; **Antonio Corradi**, University of Bologna, Italy; **Juan Carlos Naranjo Martinez**, Univerity Polytechnic Valencia, Spain; **Laura BelenguerQuerol**, Univerity Polytechnic Valencia, Spain; **Jorge Falco**,University of Zaragoza, Spain; **Roberto Casas**,University of Zaragoza, Spain; **Jose Ignacio Artigas**,University of Zaragoza, Spain;**FabioPaternò**, CNR-ISTI, Italy; **Marco Eichelberg**, Offis, Germany; **Antonio Maña**, University of Malaga, Spain.

8 Injecting the Universal Remote Console Ecosystem: The Open URC Alliance[1] *(Jan Alexandersson, Jürgen Bund, Eduardo Carrasco, Gorka Epelde, Martin Klima, Elena Urdaneta, Gregg Vanderheiden, Gottfried Zimmermann and Ingo Zinnikus)*

As a consequence of the large interest in standardized open platforms that can implement scenarios and requirements posed by the AAL and eInclusion communities, the ISO/IEC 24752 standard Universal Remote Console has emerged from a single project infrastructure within the FP6 project i2home into a steadily growing ecosystem with parties ranging from more technology oriented over business but also stakeholders that represent users. In order to coordinate and make trans- parent activities as well as provide a meeting place for different stakeholders, we have started building up the OpenURC Alliance. In this talk, we present the time line and construction of the Alliance, point at some current and future activities.

8.1 Introduction

Mostly scoped within the areas of Ambient Assisted Living (AAL) or eInclusion, we witness in the last decade many initiatives both on domestic as well as on European eleven aiming at developing ICT platforms that are used to implement scenarios for persons with special needs. Prominent motivations for these are the desire to make it possible for elderly to stay longer in their homes as in the EU funded project MonAMI [60], live an independent life as elderly citizens as in the German project SmartSenior [62] or at an extreme make possible for paralyzed persons to interact with their environment via Brain Computer Interfaces (BCI) as in the EU funded FP7 project Brainable, see [63].

In the EU-funded project i2home[59], we answered the question how to make interaction with ICT in general accessible for everyone by implementing the Universal Remote Console (URC) standard. In the course of the project an improved version of URC was standardized under ISO/IEC 24752 [58].

Following a user-centered design methodology [56, 57], the i2home project developed, implemented and evaluated different user interfaces for four user groups: persons suffering from mild Alzheimers disease, young persons with mild cognitive impairments, elderly and partially-sighted and blind persons.

8.1.1 The Two URC USPs

One intuitive view point of the URC technology is the middleware approach called Universal Remote Console (UCH) as depicted in figure 7. In line with many other middleware solutions, the UCH includes the heterogeneous appliances and services (targets) and display them as abstract user interfaces (Socket Layer in the figure). A socket is an abstract description of the input/output behavior of the target.

[1] This work has been partially funded by the European Commission under the grant FP6-033502 (i2home) and FP6-030600 (VITAL), the Saarland funded SensHome project, the BMBF funded project SmartSenior and by the National Institute on Disability and Rehabilitation Research (NIDRR) US Dept of Education grant # H133E090001/ H133E080022 to the Trace R&D Center, University of Wisconsin- Madison. The opinions herein are those of the author and not necessarily those of the funding agencies.

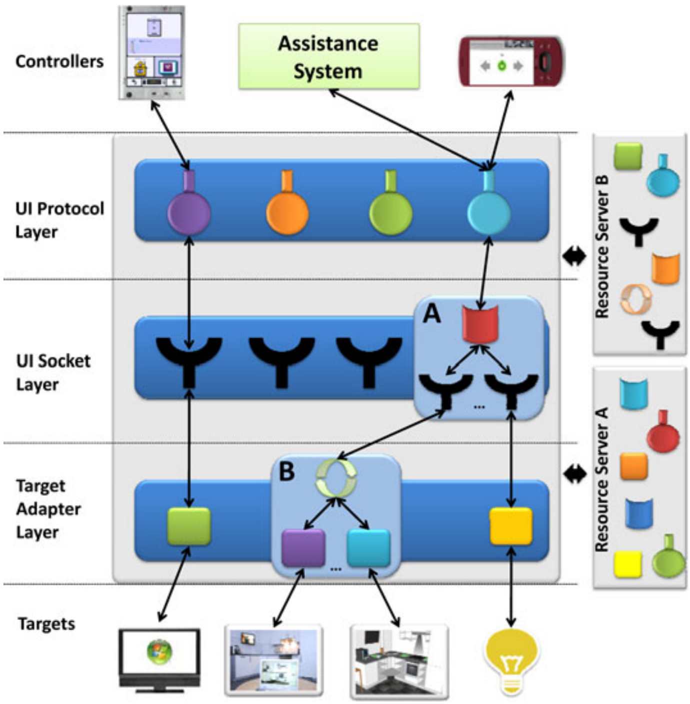


Fig. 7. The URC framework with its different layers (from the bottom): Target Adaptor Layer, Socket Layer, UI Protocol Layer. In the figure, two additional plugin extensions are depicted: A) A CEA-2018-based Task-Model Engine; B) A Synchronization Module used for implementing synchronized scenarios.

URC does not pose any requirements on the targets other than being controllable over a network. Currently, UCHs can interact with UPnP devices, ENOCEAN, WSDL services, ZigBEE, bluetooth just name a few technologies. User interfaces ranges from simple automatically generated HTML pages, accessible HTML pages, flash-based UIs, multimodal speech- and gestures-based UIs (on iPhones, Android phones, ...), TV-based, etc. have been used in diverse scenarios, like comfort, security, information services, peer-to-peer gaming within the Smart Homes, i2home implementations of the URC approach additionally allow for plugins, e.g., a CEA 2018 task model engine, e.g., [65] which can be used to combine services forming compositions thereof.

Apart from the separation of concerns which allow user interface developers to focus on the development of user interfaces rather than technical issues, there are two unique selling propositions of the URC technology not present in most other frameworks:

– *Pluggable user interfaces* meaning that there is a clear separation between the appliances and services on the one hand and the user interfaces on the other. With this approach it is possible to dynamically replace UIs with the same functionality.We strongly believe that this approach should be incorporated into ICT in general since it provides the necessary ingredients for personal and thus potentially accessible user interfaces.

– *Resource Servers* A resource necessary for connecting a target with a UI is available via resource servers. In the last years, special forms of this concept has been instantiated by, for instance, Apple with its app store. The URC approach, however, allows sharing arbitrary resources necessary for connecting a target with a user. Also, a UCH can interact with several resource servers thus allowing providers to maintain their own resources on own resource servers

8.2 Injecting the URC Ecosystem

During the course of the project, i2home provided parallel running EU-funded projects, such as, the VITAL [61] and MonAMI [60] projects with access to the URC implementation. In fact, MonAMI's field trials (50 flats) include the i2home URC implementation. But also industry has started showing interest in the URC technology and so at this year's CEBIT, German Telekom demonstrated a scenario where the URC technology serves as a middleware (in the cloud) for three different UIs (in car, PC-based and mobile-phone based) within a traveling scenario. In addition to India and the US, we also observe an adoption within academic research in Australia. Accumulated, we count projects using URC technology with about 100 partners and EURO 80 Mi, see figure 8.

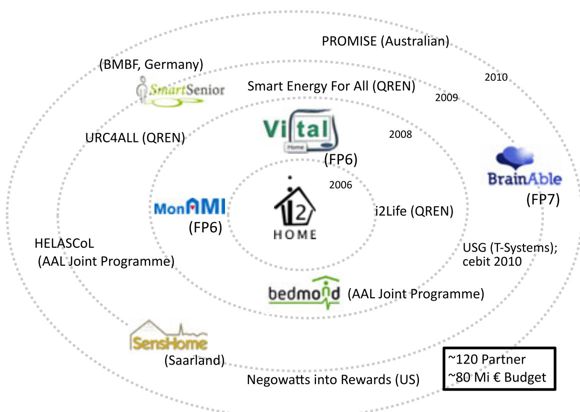


Fig. 8. International projects using URC technology

This has encouraged us to form the international OpenURC Alliance, see http:// www.openurc.org as an international platform for fostering the ecosystem that will be necessary for further the knowledge about and usage of the URC technology. The OpenURC Alliance provides a meeting place for different stakeholders interested in this technology.

Currently we are investigating what role the URC approach can play in the AALOA[2] initiative and the UniversAAL project, see [64].

8.3 Structure and Timeline of the OpenURC Alliance

At the time of writing, the OpenURC Alliance consists of three Committees:

- **Governance.** Here, the work consists of structuring the alliance as a whole. Also, work include the writing of membership rules and By-Laws.
- **Marketing.** This committee develops material for presentation and marketing, organizes presentations at conferences and fares.
- **Technical.** The technical committee further the URC technology by producing technical documents and perform standardization work. This committee maintains open-source reference implementations and technical documentation.

We are currently developing a fourth committee called the User Committee whose purpose it is to add to the alliance the perspective of the users, such as ethical considerations and how to execute user-centered design for persons with special needs, e.g., within gerontological scenarios.

8.3.1 Time line

The OpenURC is already up and functioning as an ad hoc group. We are targeting a launch of the OpenURC Alliance as a legal body in 2010-11 timeframe. The launch of a new web site is planned for early 2011. In the meantime we continue to use the current http://myurc.org web site of the current Universal Remote Console Consortium, which will fold into the new OpenURC Alliance.

8.4 Conclusion and Outlook

We have argued and showed that the Universal Remote Console approach should play a role in any framework that takes accessibility seriously in the area of ICT. The URC approach provides an open standards-based fundamental functionality for any framework or platform where the diversity of users are to be supported and where accessibility is addressed. In order to provide a platform for different stakeholders within the AAL community to communicate and collaborate in this area, we are starting the OpenURC Alliance. The Alliance will provide a meeting place for providing information about the URC, exchanging ideas, meeting other stakeholders interested in the URC technology and, finally, manage open source resources.

References

1. Ambient Assisted Living Roadmap. AAliance, `http://www.aaliance.eu`
2. `http://www.ICT-ageing.eu`

---

[2] http://www.aaloa.org

3. Ministerial Declaration: Riga, Latvia (June 11, 2006), http://ec.europa.eu/information_society/events/ict_riga_2006/index_en.htm
4. Action Plan on Information and Communication Technologies and Ageing (June 14, 2005), http://ec.europa.eu/information_society/activities/einclusion/policy/ageing/action_plan/index_en.htm
5. http://ec.europa.eu/einclusion
6. http://cordis.europa.eu/fp7/home_en.html
7. http://www.aal-europe.eu/
8. http://ec.europa.eu/information_society/activities/ict_psp/index_en.htm
9. http://www.universaal.org/
10. Wichert, R.: Configuration and Dynamic Adaptation of AAL Environments to Personal Requirements and Medical Conditions. In: HCI International 2009. LNCS, vol. 5615, pp. 267–276. Springer, Heidelberg (2009)
11. Berndt, E., Wichert, R., et al.: Marktpotenziale, Entwicklungschancen, Gesellschaftliche, gesundheitliche und ökonomische Effekte der zukünftigenNutzung von Ambient Assisted Living (AAL)-Technologien. Schlussbericht des vom BMBF geförderten ITA-Projekts FKZ 16|1575 (2009), http://publica.fraunhofer.de/
12. Wichert, R., Norgall, T., Eichelberg, M.: AAL auf demWeg in die Praxis – kritische Faktoren und Handlungsempfehlungen. In: Verband der Elektrotechnik Elektronik Informationstechnik: 3. Deutscher AAL-Kongress. VDE-Verlag, Frankfurt (2010)
13. universAAL: UNIVERsal open platform and reference Specification for AAL, http://www.universaal.org/
14. MonAMI IST project, http://www.monami.info
15. TAHI, The Application Home Initiative, http://www.theapplicationhome.com
16. OSGi: http://www.osgi.org
17. Universal Remote Console, http://myurc.org/
18. Marco, A., Casas, R., Bauer, G., Blasco, R., Asensio, A., Jean-Bart, B.: Common OSGI Interface for Ambient Assisted Living Scenarios. Ambient Intelligence and Smart Environments (2009)
19. International Standard ISO/IEC 24752. Information Technology - User Interfaces - Universal Remote Console (5 parts). ISO/IEC (2008)
20. Tools and Prototype Implementations for the URC Framework, http://myurc.org/tools/
21. Intuitive Interaction for Everyone with Home Appliances based on Industry Standards (i2home), http://www.i2home.org/
22. Value-based IT ALignment (VITAL), http://www.vital-project.org/
23. Mainstreaming on Ambient Intelligence (MonAMI), http://www.monami.info/
24. W3C Recommendation: Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language. W3C Recommendation (June 26, 2007), http://www.w3.org/TR/2007/REC-wsdl20-20070626/
25. URC Consortium. URC-HTTP Protocol 2.0 specification, http://myurc.org/TR/urc-http-protocol2.0
26. URC Consortium. Universal Control Hub 1.0 specification, http://myurc.org/TR/uch1.0
27. Soprano ISS project, http://www.soprano-ip.org
28. Persona IST project, http://www.aal-persona.org
29. MPower IST project, http://www.sintef.no/Projectweb/MPOWER

30. MonAMI IST project, http://www.monami.info
31. Oasis IST project, http://www.oasis-project.eu
32. I2Home IST project, http://www.i2home.org
33. UniversAAL IST project, http://www.universaal.org/
34. Ambient Assisted Living Strategic Research Agenda. AAliance, http://www.aaliance.eu
35. BMBF/VDE Interoperability working group, http://www.vde.com/de/Technik/AAL/Publikationen/Kongress-undFachbeitraege/Seiten/BuchInteroperabilit%C3%A4tvonAAL-Systemkomponenten.aspx
36. CECED Chain specification, http://www.ceced.org/
37. Ambient Assisted Living Strategic Research Agenda. AAliance, http://www.aaliance.eu
38. Policy Paper on Standardisation Requirements for AAL. AAliance, http://www.aaliance.eu
39. http://www.continua.org
40. Kung, A., Fagerberg, G.: Alliance for an AAL Open Service Platform. In: AALIANCE conference - Malaga, Spain (March 11 and 12, 2010), http://www.aaliance.eu
41. Interoperability framework requirements specification for service to the home (IFRS). In: Cenelec Workshop Agreement CWA 50560 (June 2010)
42. AALOA: http://www.aaloa.org
43. OpenURC: http://www.openurc.org
44. AAL as introduced to the European research, http://www.aal169.org/Published/aal2103.pdf
45. AAL Joint Programme, http://www.aal-europe.eu/aal-association
46. Ambient Assisted Living Roadmap, http://www.aaliance.eu
47. Continua Health Alliance, http://www.continuaalliance.org/
48. Aarts, E.H.L., Encarnação, J.L.: True Visions: The Emergence of Ambient Intelligence. Springer, Heidelberg (2006)
49. Weiser, M.: The Computer for the 21st Century. Scientific American 265(3), 66–75 (1991)
50. Satyanarayanan, M.: Pervasive Computing: Vision and Challenges. IEEE Personal Communications 8(4), 10–17 (2001)
51. Brandenburger, A.M., Barry, J., Nalebuff, B.J.: Co-opetition. Doubleday, NewYork (1996)
52. Bellavista, P., Corradi, A.: The Handbook of Mobile Middleware. Auerbach Publications (2006)
53. Behlendorf, B.: Open Source as a Business Strategy Open Souces. O'Reilly (1999)
54. Letellier, F.: Open Source Software: the Role of Nonprofits in Federating Business and Innovation Ecosystems, http://en.wikipedia.org/wiki/Fran%C3%A7ois_Letellier
55. Connelly, K., Siek, K.A., Mulder, I., Neely, S., Stevenson, G., Kray, C.: Evaluating Pervasive and Ubiquitous Systems. IEEE Pervasive Computing 7(3), 85–88 (2008)
56. Being human: Human-computer interaction in the year 2020 (2007), http://research.microsoft.com/hci2020/
57. Cooper, A.: The Inmates are running the Asylum: why high-tech products driveus crazy and how to restore the sanity, 1st print edn. SAMS, Indianapolis (1999)
58. ISO.: ISO/IEC 24752: Information Technology — User Interfaces — Universal remote console — 5 parts. International Organization for Standardization (2008)
59. i2home: Interaction for everyone with home appliances based industry standards (2006–2009), http://www.i2home.org (funded by the EU/FP6)

60. Monami: Mainstreaming on ambient intelligence (2006–2010), http://www.monami.info (funded by the EU/FP6)
61. Vital: Vital assistance for the elderly (2006–2010), http://www.vital.org (fundedby the EU/FP6)
62. Smartsenior (2009–2012), http://www.smart-senior.de (funded by BMBF, Germany)
63. Brainable—autonomy and social inclusion through mixed reality brain-computerinterfaces: Connecting the disabled to their physical and social world (2010–2012), http://www.brianable.org (funded by the EU/FP7)
64. Universaal—universal open platform and reference specification for ambient assisted living (2010–2013), http://www.universaal.org/ (funded by the EU/FP7)
65. Rich, C.: Building task-based user interfaces with ansi/cea-2018. Computer 42(8), 20–27 (2009)

Author Index