Compensating for On-Body Placement Effects in Activity Recognition

Kai Kunze

This dissertation has been submitted to the
Universität Passau
Fakultät für Informatik und Mathematik
Thesis Advisor: Prof. Dr. Paul Lukowicz
Co-Advisor: Prof. Dr. Hans Gellersen

June 2011

Acknowledgments

My sincere gratitude goes to Paul Lukowicz, my thesis advisor, who is often on a tight schedule. Yet, he always makes time to provide valuable research input and inspiration. I can always rely on his support and vast experience.

Additionally, I would like to give a warm thank you to Hans Gellersen. I'm proud to have him as second advisor, he's an extraordinary researcher and I valued his input and help throughout my Ph.D. studies.

Thanks to David Bannach, Gernot Bahle, Josef Neuburger, Gerald Pirkl, Jens Weppner and Kamil Kloch, for being great peers, for critic and proofreading, Georg Ogris, Tobias Franke, Jinyuan Cheng, Agnes Grünerbl, Michael Barry, Karl Stockinger and Peter Barth for being outstanding colleagues and for sparking interesting discussions.

Special thanks go to Ian Cloete, Christoph Schuba, Helmut Stadmüller and Ernst Heinz, all exceptional and inspiring teachers. Without their guidance, I would definitely not work in research today.

Thanks to Thad Starner, Albrecht Schmidt and Michael Beigl for giving valuable advice and sharing their expertise. Other friends and inspiring colleagues, I want to thank: Oliver Amft, Ulf Blanke, Todd Farrell, Martin Kurserow, Till Riedel, Michael Springmann, Kristof Van Laerhoven and Jamie Ward.

Cyrille Thouvenin, Roland Westrelin, Eric Lemoine and Nicolas Fugier thanks for an amazing time in Grenoble.

Kurt Partridge, Bo Begole, Rowan Nairn and Max Van Kleek thank you for a great summer at PARC.

Holger, Melanie, Sanjit and Julia deserve also a thank you for their close friendship and for putting up with me over the last couple of years.

Most importantly, I sincerely thank Renate Kunze, my mother, and Anita and Helmut Woll, my grandparents, for their love and support. No matter what, I can always count on them. Without them this work would not have been possible.

Kai Kunze
Passau, 2011

Abstract

This thesis investigates, how placement variations of electronic devices influence the possibility of using sensors integrated in those devices for context recognition. The vast majority of context recognition research assumes well defined, fixed sensor locations. Although this might be acceptable for some application domains (e.g. in an industrial setting), users, in general, will have a hard time coping with these limitations. If one needs to remember to carry dedicated sensors and to adjust their orientation from time to time, the activity recognition system is more distracting than helpful. How can we deal with device location and orientation changes to make context sensing mainstream? This thesis presents a systematic evaluation of device placement effects in context recognition. We first deal with detecting if a device is carried on the body or placed somewhere in the environment. If the device is placed on the body, it is useful to know on which body part. We also address how to deal with sensors changing their position and their orientation during use. For each of these topics some highlights are given in the following.

Regarding environmental placement, we introduce an active sampling approach to infer symbolic object location. This approach requires only simple sensors (acceleration, sound) and no infrastructure setup. The method works for specific placements such as "on the couch", "in the desk drawer" as well as for general location classes, such as "closed wood compartment" or "open iron surface". In the experimental evaluation we reach a recognition accuracy of 90% and above over a total of over 1200 measurements from 35 specific locations (taken from 3 different rooms) and 12 abstract location classes.

To derive the coarse device placement on the body, we present a method solely based on rotation and acceleration signals from the device. It works independent of the device orientation. The on-body placement recognition rate is around 80% over 4 min. of unconstrained motion data for the worst scenario and up to 90% over a 2 min. interval for the best scenario. We use over 30 hours of motion data for the analysis. Two special issues of device placement are orientation and displacement. This thesis proposes a set of heuristics that significantly increase the robustness of motion sensor-based activity recognition with respect to sensor displacement. We show how, within certain limits and with modest quality degradation, motion sensor-based activity recognition can be implemented in a displacement tolerant way. We evaluate our heuristics first on a set of synthetic lower arm motions which are well suited to illustrate the strengths and limits of our approach, then on an extended modes of locomotion problem (sensors on the upper leg) and finally on a set of exercises performed on various gym machines (sensors placed on the lower arm). In this example our heuristic raises the displaced recognition rate from 24% for a displaced accelerometer, which had 96% recognition when not displaced, to 82%.

Contents

List of Figures

List of Tables

1

Motivation

"An approximate answer to the right problem is worth a good deal more than an exact answer to an approximate problem."
- John Tukey

Our everyday lives get more and more saturated with computing devices and embedded in them, a wide variety of sensors. With access to powerful computing, sensors and a ubiquitous Internet, why is context recognition, and with it pervasive computing, not further along and more broadly adapted? With few exceptions, most context recognition research done to date assumes dedicated sensing devices with fixed locations. These locations are often carefully chosen to suit a particular application. This is a major limitation hindering broad adoption of pervasive computing systems. Who wants to bother to put on a second "sensor suit" before leaving to work in the morning or re-adjust shifted sensors every couple of hours? Is there a way to deal with device placement and orientation changes to make context sensing more mainstream?

Pervasive Computing has matured. Today, we rely more and more on computing devices to help us in our daily activities, with the smart phone becoming an important platform [11, 25, 35, 9, 18]. As these devices –and with them computing– become tighter integrated in our everyday lives, the performance bottlenecks, even on mobile platforms, are no longer RAM capacity, CPU speed etc. We use computing more and more in environments in which we cannot focus our attention completely on the device in question (e.g. during our daily commute, while waiting in a queue, during lectures/meetings). In these situations, human attention is sparse. It becomes the crucial bottleneck.

For the user to be able to interact with computer systems even in "busy" environments, pervasive computing introduces context recognition as a core concept [16]. Context recognition infers relevant information about the current situation of the user utilizing sensors carried on the body and distributed in the environment. This information, the user's "context", helps to minimize direct user interaction. Pervasive applications adjust their behavior pro-actively according to the given situation.

Context recognition is not just a theoretical concept introduced by the research community. The tech industry also embraces context-aware computing (computing utilizing context recognition) as a game changer. TechCrunch, a company offering news around high-tech and web start-ups, names "Context-Aware Apps" as one of 7 most important technologies of 2011 and describes them as follows:

> *"Context-Aware Apps: Whether it's search, mobile, or social apps and services, the most useful apps people will keep coming back to are the ones which help people cut through the increasing clutter of the Internet. Apps that are aware of the context in which they are being used will serve up better filtered information... In a world of information overload, context is king."* [1]

Companies not only acknowledges pervasive computing, some products already adopt pervasive computing concepts with a focus on context recognition technologies. Foremost, location-based services gain more and more momentum, from simple location check-ins (Foursquare, Gowalla, Facebook places etc.) over location-based reminders (Apple's Reminder App) to recommending restaurants close by (Yelp). Although Schmidt et. al. already realized in 1999 that "There is more to context than location" [44], it took a while for services and applications to utilize other sensing modalities. The first products with notable commercial success just emerged over the last years, for example the Nintendo Wii, a gaming console using inertial motion sensors, and Fitbit [2], acceleration and air-pressure sensors embedded into a clasp that records steps taken, stairs climbed and simple modes of locomotion. Although these products are already useful and a commercial success, they seem to apply quite simple context recognition algorithms (e.g., the Nintendo wii remote uses thresholding on the acceleration intensity levels to detect specific motions). How much better could we get employing more complex context recognition approaches, already well established in the pervasive computing research?

Common sensing modalities in pervasive computing include motion (acceleration, angular velocity) and sound. Generally, we cannot expect a developer with no background in signal processing or pervasive computing to figure out the user's context from raw accelerometer or sound data. For example, it is more

[1] http://techcrunch.com/2011/01/02/seven-technologies-that-will-rock-2011/
[2] http://fitbit.com/

difficult to infer the modes of locomotion from accelerometer data than the user's location from a given GPS fix. This is foremost due to the GPS "raw data" being closer to the contextual information, e.g. getting the user's location and shops/buildings in his immediate surrounding is easy using Google Maps, Openstreet Map or similar services, once we have a GPS fix. Additionally, the software support for location services is better on most platforms. Therefore, better APIs and software libraries for other sensing modalities are needed and in the process of being built. However, the main issue, why pervasive computing research is not more broadly adopted, lies somewhere else. With few exceptions (e.g. [28]) the bulk of context and activity recognition research assumes known fixed sensor locations often carefully chosen to recognize specific tasks (e.g. [36, 3]). Therefore, for each application, the user has to put specific sensors at certain well-defined positions on his body or in the environment. Yet, it is unrealistic to expect this from the user for a more widespread use of pervasive systems. The burden placed on the user is too high. While the user is on the move, he is sometimes in highly augmented environments with a lot of contextual information from smart object. Sometimes, in places with little or no infrastructure, the user needs to rely on the smart objects carried on his body. The best we can realistically expect in terms of context sensing is that at any given point in time the user carries a more or less random collection of sensor enabled devices (mobile phone, watch, headset etc.) on different body locations, eg. in his pockets, bag, wrist. To reach a more wide spread adoption of context recognition applications, we should utilize these user-owned, sensor-enable devices. However, is it even possible to use these device sensors to infer information about the user's situation? Generally, the quality of these built-in sensors is not much different from sensors typically used in dedicated wearable sensing systems. Thus, for example, the iPhone features the LIS302 acceleration sensor, which has also been widely used in wearable context recognition [11, 25, 35, 9]. Sampling rates and AD conversion accuracies are also comparable.

Mobile phones and other smart devices, however, are carried by users in a variety of locations [20]. In most cases they are not firmly fixed to the body but placed in a pocket or bag where the can shift around and change orientations frequently. Not only is the on-body location unknown, the devices are also moved out of place over time. Of course, shifting sensors during long term deployment is also a – mostly ignored– issue for dedicated wearable sensing systems.

This thesis describes approaches and methods to deal with exactly these issues of changing sensor locations, shifts in orientation and displacements. As such my research narrows the gap between the need for context-aware applications and the practical problems, encountered when trying to implement them in real life.

The next section discusses the current research landscape in pervasive computing with a special focus on orientation and placement independence, followed by a detailed related work discussion centered around the contents of this thesis.

The chapter ends with a detailed description of the contributions and an outline.

1.1 State of the Art in Context Recognition

Context recognition is an important basis for pervasive computing. In the early 1990s, Mark Weiser coined the terms ubiquitous computing and calm technologies. Following his vision, computing is to transparently surround and support us during everyday life [54, 53]. Thus, computer systems need to become *proactive*. To support the user in any given situation, computing needs to be able to "perceive" the real world. The means to achieve this are summarized in the term "context recognition". For a more detailed discussion on "Context" and some more formal definitions see Dey et. al. and Schmidt et. al. [42, 43].

Context recognition, today, developed into an interdisciplinary field building on embedded systems, signal processing, machine learning, statistics and artificial intelligence. Since context sensing is a core concept of pervasive computing, there exists a vast body of research. As such, there are many ways to provide a structural overview about the field. In the following, three possible categorizations of the research and their relations to the thesis topic are discussed in , namely inference type, sensing modality and infrastructure vs. on-body sensing.

Infrastructure vs. on-body sensing – Continuing the initial work of Mark Weiser, researchers first focused on infrastructure sensing, building pervasive rooms using stationary devices (e.g. fixed cameras, microphone arrays) as sensory inputs [12, 40]. Installation costs and the lack of significant applications hindered a wide adoption in everyday life. Most room features were nice to have (e.g. automated access and capture), yet not crucially important. Today, there are some efforts to make infrastructure sensing available to the masses. Patel et. al. shows interesting research using the power line as sensor, to detect the type of electronic devices in use and to utilize it as RFID reader [37, 38].Complementary to the infrastructure sensing approach is research focusing on on-body activity sensing. In on-body sensing, we use devices carried or worn by the user as sensory inputs [6]. A major advantage of this type of sensing is it "follows" the user, as he carries the system with it. On the other hand, to carry around dedicated sensing devices places an additional burden on the user. These devices can be annoying and heavy, especially regarding early wearable research prototypes. Often, multiple accelerometers positioned on the users' body are used to support diverse applications, from a meeting annotation tool to motion analysis in sports. [23, 33, 58]. Sound, in an on-body sensing scenario, can be used to infer a particular room the user is in, distance between devices and even some distinct activities (e.g. the use of a coffee grinder) [46, 57]. There are also more

and more hybrid approaches combining infrastructure and on-body sensing. In this case, on-body sensors interact with devices in the environment. The most developed context type in this aspect is location. For a more detailed discussion of different location sensing approaches, including relative and absolute positioning, please refer to Section 2.1.

Sensing modality – Common sensing modalities in early work include mostly sound and vision, yet also acceleration is very prevalent. The simplest sensors used in activity recognition are binary. They produce only an activation signal, e.g., RFID readers/tags and ball switches. More sophisticated experimental setups integrate motion sensors (accelerometers, gyroscopes and magnetic field sensors combined), force resistive sensors, sound and a location system to detect activities from fine-grained work steps at a car assembly to food intake gestures [36, 1]. Context recognition also includes some more exotic sensing modalities, from eye movement capture using electrooculography to emotional state detection over galvanic skin response [7, 55]. It is often difficult to pick the correct modality for the application task at hand. So far, researchers rely heavily on experience.

Inference type – The kind of context recognition algorithms used ranges from simple thresholding over frame-by-frame recognition approaches to sophisticated time-series algorithms [6, 36]. Inference often follows a chain. Close to the raw sensor data, embedded systems and signal processing methods are applied, followed by one or several machine learning/artificial intelligence approaches. These, in turn, utilize some specific knowledge encoded from the given application domain. A very popular research topic is also the fusion of different sensor modalities using specific inference types, namely feature and classifier fusion (and hybrid approaches). For an introduction to this topic see Ruta et. al. [41].

This thesis centers on on-body sensing, although some of the presented approaches work very well in an infrastructure setting, especially the active sampling method in Chapter 2.

To review the current state of context recognition we will explore recent research along the two other categories: sensing modality and inference type. Table 1.1 gives an overview. We center on some highlights from this summary moving along the sensor modality axis first and the inference type second, starting with "binary" for "raw/thresholding" inference, over motion, sound, vision, capacitive to multiple. For each sensing modality, we first look at relative simple inference types from raw over frame-by-frame to more complex time series and hierarchical approaches.

"Binary" regarding the sensor modality stands for the granularity of the sensor resolution. The sensor can only distinguish activation versus no activation,

Table 1.1: Exploring activity recognition research along the two dimensions: sensing modality and inference type. We present the reference and the name of the first author, as well as the application domain activity recognition is applied to in their respective research.

Sensing Modality/ Inference Type	Binary	Motion	Sound
Thresholding-Raw	Langheinrich[27] Groceries	Siewiorek[45] Activity levels	
Frame-by-frame	Gordon [17] Office work	Bao [6] Locomotion	Stäger[46] Assisted Living
Time series	Taipa [49] Household	Westeyn[56] Autism	Wyatt[57] Social Dynamics
Hierarchical	Patterson[39] Daily routine	Huynh[19] Daily routine	Choudhury[10] Conversation

Sensing Modality/ Inference Type	Vision	Capacitive	Multiple
Thresholding-Raw	-	-	-
Frame-by-frame	Kerhet[21] Movement	Cheng[8] Swallowing	Ward[32] Workshop
Time series	Starner[47] Overview	Amft[1] Food intake	Ogris[36] Assembly
Hierarchical	Andriluka[2] People Tracking	-	Bannach[5] Morning Routine

e.g. a switch sensor integrated in a cupboard door emits either "door opened" or "door closed". Langheinrich et. al. provide an excellent example for the simplest inference type using the raw signal from binary sensors [27]. They use RFID tags embedded in consumables bought in the supermarket. The bought products are match against recipes and the user receives recipe suggestions. Moving to the frame-by-frame inference type, features are usually calculated on the sensor data over a sliding or jumping window. These are then used to do a frame-by-frame classification with standard machine learning algorithms (e.g. KNN, desicion trees). Gordon et. al. show how to utilize simple binary ball switches as an interesting alternative to accelerometers. The new design presented in the paper is very small (2 x 3 mm) and works well for high frequency movements [17]. Although they deliver only binary information, the initial analysis on an office data set indicates that they can also be used as a complementary sensor to accelerometers due to their ability to capture high frequency components. Combining both increases the overall accuracy. Van Laerhoven et. al. present a comparison between traditional ball switches and accelerometers [52]. They use multiple ball switches in several orientations to compensate for the information loss compared to an accelerometer.

Motion in general and the accelerometer in particular is a very prominent sensing modality used in context recognition. Siewiorek et. al. present a mobile phone platform that can log the users activity levels during everyday activities, using simple thresholding on the acceleration norm. Very early work from Bao used the frame-by-frame classification approach to detect modes of locomotion: walking, standing, sitting etc. [6]. Frame-by-frame classifiers work well on context types that are repetitive in nature (e.g., walking). Using time-series approaches is relatively common for more complex activity recognition based on motion. Mostly Hidden Markov Models and Conditional Random Fields are applied. Westeyn et. al. introduce a system that assesses the risk for autism in toddlers [56]. They integrate motion sensors into toys, recognizing specific repetitive gestures indicative of autism. The recognition results are used to store and augment a video for later expert review. The application scenarios for motion based inference are pretty wide and range from daily routine over furniture assembly to car manufacturing, usually a combination of frame-by-frame, time series and hierarchical inference methods is applied to reach satisfactory recognition results in realistic application scenarios [3, 36].

A good reference for sound-based context recognition is research by Stäger et. al. [46]. They present an approach evaluated on low power special purpose hardware, optimizing power consumption and recognition rate. Both of them are obviously competing goals. Given the needed accuracy and the power constraints, their method enables to find the best trade-off. The application scenarios shown are kitchen work and, more general, assisted living. Very interesting work using sound and higher level inference methods comes from Wyatt et. al.[57]. They try

to determine the structure and changes in social networks by detecting face to face communication.

The use of vision in activity recognition research is sparse compared to motion or sound, especially if we focus on on-body sensing. This is largely due to the fact that cameras are a complex sensor type and influenced by many noise sources (lighting, reflections etc.). Vision inference also requires a significant amount of computing power. One of the few usage scenarios for vision in a wearable setting is the recognition of sign language [47]. An overview about the role of computer vision in activity sensing is given by Starner et. al. [47].

Using multiple sensors for inference, also called multimodal activity recognition, presents additional problems, as one needs to find the means to successfully fuse them. Most work in this category uses one sensing modality to do the segmentation of the sensor data before the classification is performed. Ward et. al. use body-worn accelerometers and microphones to recognize workshop activities (drilling, sandpapering etc.) using an interesting segmentation technique [32]. The user wears a microphone on the wrist and one on the torso. To recognize if an interesting event happens (i.e. the user works with this hand), one compares the intensities of the two microphones. If the intensity on the wrist is higher than a given threshold, it is very likely the sound originated from an activity the user performed with his hand. Ogris et. al. follow a similar segmentation approach by filtering according to location. They use an indoor location system to pre-select the activity class. As many given activities can only be performed at specific locations, this can be used to limit the set of activity candidate of the classification stage (e.g., you wont brew coffee in your car). In addition they apply a variety of sensor fusion methods (voting etc.) to a car assembly data set. More detailed work related to multimodal activity recognition can be found in Sections 2.1 and 5.2.

These categories are, by no means, meant to present a complete classification of the context recognition field. Yet, they help to categorize this thesis. We will center on on-body sensing using mainly motion sensors with a specific emphasis on novel inference algorithms and sensor fusion methods.

The integration of the more common sensing modalities in smart appliances and the wide-spread adoption and usage of these devices opens up new, fascinating opportunities for activity recognition, moving slowly from recognizing the small-scale activities of a single person towards inferring collective social activities e.g. crowd density, emotional state and enabling citizen science [13, 4]. We will discuss this field in greater detail in the future work section of the conclusion.

1.2 Related Work

While there is a vast variety of context recognition applications and sensor modalities, as seen from the examples above, so far traditional research work follows a specific pattern. Given an application domain, the researchers use dedicated sensor devices hand-picked for the tasks according to intuition and experience. To perform context recognition, a standard procedure is to aggregate the sensor data using some kind of feature calculation e.g. a sliding window approach. There is no standardized approach for picking them yet. Also, the feature selection and recognition methods often rely on specific sensor devices with fixed position and known orientation [31, 33, 23].

Using these methods implies for the user to carry dedicated sensors and fixing them at specific placements. This is impractical for a wide range of application scenarios. Integration in existing devices and device placement independence are two important requirements to apply context recognition in real life settings. Device placement independence depends highly on the sensing modality used. Most inferences based on accelerometers depend on specific on-body placement and orientation, as variations of the accelerometer placement lead to changes in the acceleration signal. The motion distribution on the sensor axes changes significantly even with small variations. Other modalities are a bit more placement independent, e.g. sound, bluetooth/wlan signal strength. Subsequently, we discuss the specific scientific background and state of the art towards more placement independent activity recognition using everyday objects. To better understand how feasible it is to use regular objects, e.g. mobile phones or keys, we take a look at research in activity recognition, focusing on sensor device integration and approaches to deal with orientation and placement independence.

Device Placement Independence

For the remainder of this thesis, we distinguish between three sensor deployment changes: coarse variations, fine grain changes and device orientation changes. Coarse variations imply a change in the "global position" of the device, e.g. putting a sensing device from the table in the pocket. Fine grain variations involve shifts within a "global position". Orientation changes refer to changes of the reference system of the sensor (leaving its global position unchanged), e.g. turning a sensor 180 degrees around an arbitrary axis. A more detailed classification of deployment changes is given in the aims and contribution section later in this chapter. Three basic approaches to deal with device placement changes are found in the literature:

Unawareness: The most trivial method is to not deal with device placement variations at all. As soon as the user recognizes a miss-classification from

the system, it is up to him to fix it. Most of the work cited in the previous section belongs into this category.

Usage of less dependent sensors/features/algorithms: Different modalities exhibit differing degrees of susceptibility to the discussed changes. A microphone is more placement independent than an accelerometer, for example. Exploiting this, however, relies heavily on the application scenario and the recognition tasks to support. For example, Van Laerhoven et. al. introduce simple switch sensors and show that they are less body placement dependent than accelerometers with similar recognition rates for simple activity recognition tasks [51].

Another possibility to become more device placement independent is to use more robust features. One can aggregate the sensor data using features that alter less due to changes in orientation and placement. For example, a feature often used in accelerometer based activity sensing is the norm of the acceleration vector, as it is orientation independent. Of course, introducing these aggregations will generally lead to information loss. As soon as some aggregations are introduced, the recognition rates will decrease. To compensate this, it is common to combine different sensor modalities. Krause et. al. describe such sensor fusion methods [24].

Variations in sensor placement and orientation can be integrated in the training set, relying on the machine learning algorithm to abstract these. Lester et. al. show how to do activity recognition, utilizing these three concepts, multiple modalities, independent features and a test set with large variations [29]. They use a small dedicated sensor board to reliably detect modes of locomotion in a user group (12 participants over 12 hours of data) that carried the device on various body locations. The inference is based on frequency features, also using the accelerometer norm as aggregation. Modes of locomotion, however, count to the very basic activities to be detected. Lu et. al. apply two concepts, using a robust sensing modality -sound- and again a data set with large variability [30]. The system is implemented on an iPhone and able to recognize ambient sounds common in daily living scenarios.

Placement and orientation inference: Most work in current activity recognition research is done towards orientation independence. There are some heuristics for accelerometer sensing to detect the vertical orientation [34, 22]. Thiemjarus presents an approach to perform activity recognition device orientation independently, posing the orientation as a classification problem [50]. She uses a dedicated device on the hip equipped with an accelerometer. The orientation inference algorithm is trained on the different device rotations. Presented with a changed device orientation, it then re-

turns a rotation matrix to be applied to the accelerometer data. Afterwards, an unaltered classifier can be used. Although an interesting approach, the paper only shows it working for 4 device orientations of a device attached to a belt. It still needs to be assessed how generalizable the method is. Steinhoff et. al. show several methods on how to tolerate orientation changes and displacement effects for motion sensors (accelerometer, gyroscope and compass) in the front trouser pockets [48]. They use the two heuristics described before, rest periods and low pass filtering in combination with principle component analysis methods (for comparisons) to infer the user's direction of motion. Yet, this can just be applied to dead-reckoning and similar applications. The closest work related to the displacement contribution comes from Forster et. al. presenting a genetic programming method for feature extraction. Although the method can compensate sensor displacements, it requires training with multiple sensors [14]. Forster et. al. also introduce an online unsupervised classification algorithm for accelerometers that can deal with sensor displacements, yet the algorithm needs to run for the complete usage time of the sensor device [15]. It seems to depend highly on the chosen activity classes and the method cannot compensate larger displacements.

From dedicated device to appliances

Some early work from Schmidt et. al. describes device integration of sensors, to alter screen rotation depending on how the users hold the device. In recent years, mobile phones are increasingly becoming the platform of choice for context aware applications. According to industry estimates in 2010, around 30% of all mobile phones will be equipped with an accelerometer. For smart phones this figure is close to 100%. Many high end devices are also equipped with a variety of additional sensors such as a digital compass, gyroscopes, GPS and WiFi interface which can be used for indoor location. In addition a phone trivially has a microphone which can be used for auditory scene recognition.

There is already some initial work using mobile phone as sensing devices. Lane et. al. give a good introduction and overview about this topic [26]. Chronis et. al. try to tackle social interactions using mobile phones, attempting to detect shifts in habits and correlating them to events in the users life. They show how they track political opinion in a study conducted at a dorm room at MIT using regular off the shelf smart phones [11]) Mobile phones can also be used to localize the situation a user is in [35]. Ofstad et. al. use audio fingerprints collected over the built-in microphone of the iPhone to detect the semantic location of the user. Although Lester et. al. use dedicated hardware, their work discussed in detail in the previous section still contributes towards better device integration, as the sensor board is designed to be easily integrated in a phone [28]. These examples,

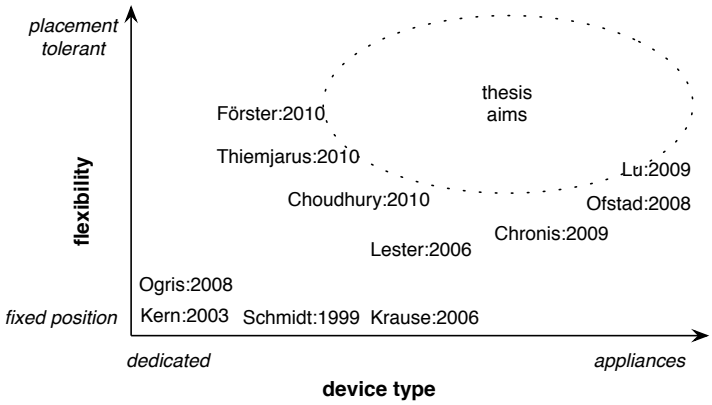


Figure 1.1: Significant related work in relation to the aims of this thesis, depicted in two dimensions: flexibility (tolerant is the methods to displacement) and device type (dedicated sensing device towards integrated in everyday appliances).

however, employ only less dependent sensing modalities, e.g. sound, bluetooth, or large diverse training sets to deal with placement issues.

In Summary

To the best of my knowledge, there has been no detailed study about placement effects on common activity sensing modalities. Nobody, so far, tried to detect the on-body-positioning of devices. Providing heuristics towards better orientation and placement independent activity recognition is also rather unexplored, disregarding the few exceptions given in the related work. Figure 1.1 depicts the current state of the art in research in the two dimensions set by two requirements: Placement and orientation independence (named flexibility) and the device integration. Although only to be taken as indication, it summarizes the lack of research towards flexible context recognition methods that are able to be used on commodity devices. Ogris et. al. use dedicated hardware without dealing with device orientation or placement changes, yet are able to detect very complex, large sets of activities [36]. Foerster et. al. show some work to deal with sensor displacement. They use dedicated hardware and rely on multiple accelerometers on different body parts making it hard to use this approach in everyday appliances [14, 15]. Lu et. al. and Ofstad et. al. take off-the-shelf smart phones for their inference. They leverage only sound and bluetooth as sensing modalities, as those are more resistant to placement issues [35, 11]. Each chapter, in turn, provides some more detailed analysis on related work specifically focused on the theme at hand.

The area in which the aims of this thesis contribute is also depicted in Figure 1.1. Of course, the complete area is too huge to be tackled by one dissertation.

Therefore, we go into a description of the concrete goals.

1.3 Aims

If we utilize everyday devices owned by the user for context recognition, sensors are not firmly fixed to the body but placed in a pocket or bag where they can shift around and have different orientations. In general we can distinguish three types of device placement variations:

1. Coarse variations related to the body part on which the device is carried. Typical examples include front or back trousers pocket, jacket pocket, arm holster, hip holster or a bag [20].

2. Fine grain variations within a given coarse location. This includes a phone shifting around in the pocket or a holster (as often used for running) being pulled up or down on the arm.

3. Variations of orientation of the device with respect to the users' body. Thus, a mobile phone could be put into the pocket with the front facing towards or away from the body. In addition devices may turn around the axis perpendicular to the body, in particular if they are small and loose in pockets.

In this thesis, I discuss the impact of the above device placement variations on the performance of context recognition systems. Specifically I address the following questions:

1. How are common sensing modalities used today influenced by the different placement issues?

2. What techniques can be used to mitigate placement effects and make recognition systems more placement invariant?

3. How can environmental and on-body placement be detected to allow the system to adapt, e.g. select a classifier trained on a specific location?

1.4 Contributions

This thesis presents a systematic evaluation of device placement effects on activity recognition. It analyzes the problems for each of the distinct parts and presents solutions to specific problems detailed below. The aims section 1.3 already classified these parts in terms of variations and the subsequent overview 1.5 places them in the structure of the thesis. In the following, the main contributions are given:

A categorization of the placement factors concerning context recognition systems is proposed. Although there are research efforts regarding placement independence, those conducted so far are focused on single sensing modalities and specific use-cases (e.g. [48]) instead of generalizing towards some classification of placement problems. I propose a categorization of the placement factors independent from specific use cases taking into account common types of sensor modalities.

A systematic evaluation of these device placement factors on the common sensing modalities used in activity recognition today is presented. Actual placement effects of sensors are identified and assessed on a signal level according to their severeness on the activity recognition process.

Solutions and heuristics are outlined to minimize the impact of these factors for more reliable, realistic context recognition. Depending on the placement effects, actual classification processes are introduced (e.g. for coarse grain variations it is sensible to recognize the current placement first and then apply a classifier specifically tuned to it). For other, more fine grained variations, heuristics to compensate them are shown.

1. I present a method to infer the symbolic placement of a device (including locations on the body versus in the environment) using an active sampling approach with sound and vibration/acceleration.
2. To deal with coarse grain variations in placement, I develop and evaluate an approach to detect the device placement on the body for common on-body positions using motion sensors.
3. To deal with displacement issues, I present and assess a heuristic based on a rigid body approximation using motion sensors.
4. To infer the orientation of a device, I evaluate a possible solution based on accelerometers and the assumption that the user is walking straight.

1.5 Overview and Outline

Placement effects can be broken down into the 3 different types of variations. They present the basis for the main questions dealt with in this thesis.

Figure 1.2 gives a detailed description and a categorization of the problems posed for this work. Coarse variations in sensor signals give information on whether a device is worn on the body or placed somewhere in the environment. Some environmental placements come with their unique sensor signature depending on the modality. This question is handled in Chapter 2 "Device Placement in the Environment or On-Body".

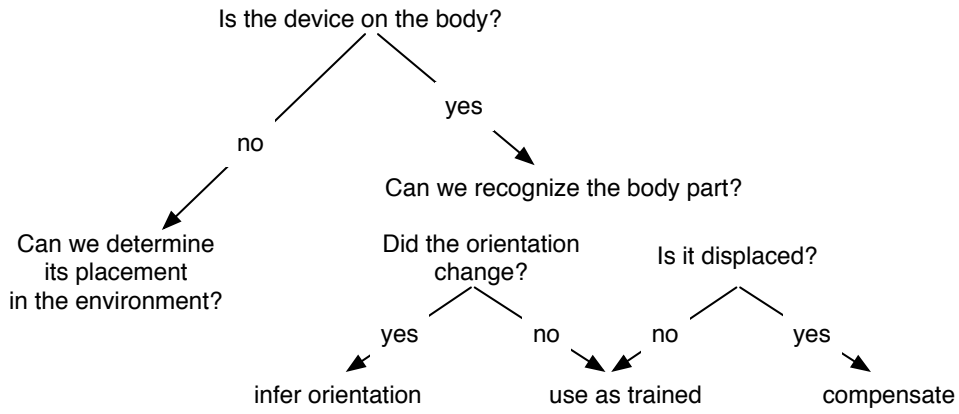


Figure 1.2: Thesis Overview

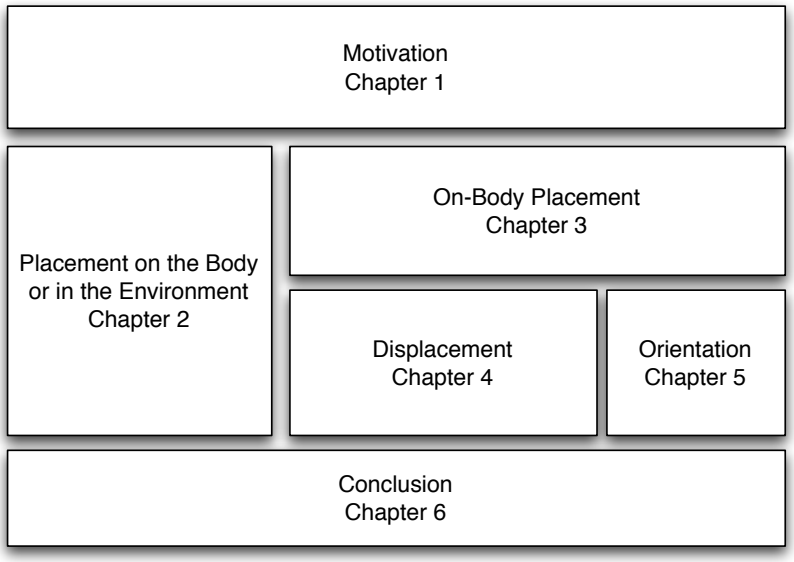


Figure 1.3: Thesis Chapters

If the device is placed on the body, the next logical question is which body part it is located on. Impacts on the sensor signals and possible recognition solutions dealing with this sub-question are handled in Chapter 3 "On-Body Placement".

The next two special issues discussed result from long-term deployment. We deal with translational shifts in Chapter 4 "Displacement" and orientation changes

Table 1.2: Publications included in this thesis with the respective chapter.

Chapter	Publication
1	Kunze,K., Partridge, K. and Lukowicz, P. Compensating placement variations in body worn context recognition systems *Submitted to IEEE Pervasive Computing Magazine, 2012*.
2	Kunze, K. and Lukowicz, P. Symbolic object localization through active sampling of acceleration and sound signatures. *In Proceedings of the 9th international Conference on Ubiquitous Computing*. Innsbruck, Austria, September 16 - 19, 2007.
3	K. Kunze and P. Lukowicz. Using acceleration signatures from everyday activities for on-body device location. *11th IEEE International Symposium on Wearable Computers*, Sep 2007. K. Kunze, P. Lukowicz, H. Junker, and G. Troester. Where am i: Recognizing on-body positions of wearable sensors. *LOCA'04: International Workshop on Location and Context Awareness* , Jan 2005.
4	Kunze, K. and Lukowicz, P. Dealing with sensor displacement in motion-based on-body activity recognition systems. In Proceedings of the 10th international conference on Ubiquitous computing (UbiComp '08). Seoul, Korea, September, 2008.
5	Kai Kunze, Paul Lukowicz, Kurt Partridge, Bo Begole, Which Way Am I Facing: Inferring Horizontal Device Orientation from an Accelerometer Signal, *13th IEEE International Symposium on Wearable Computers*. Linz, Austria, 2009.
6	Kunze, K., Bahle, G., Lukowicz, P., and Partridge, K. Can magnetic field sensors replace gyroscopes in wearable sensing applications *In ISWC '10: Proceedings of the 2010 11th IEEE International Symposium on Wearable Computers*. Seoul, South Korea, 2010.

in Chapter 5 "Orientation".

Figure 1.3 depicts the chapters. The on-body placement chapter is related to the orientation and displacement chapters. Therefore, it is recommended to read them in order. The conclusion chapter provides a summary of the thesis and pointers for future work. Table 1.2 gives an overview about the publications used in this thesis and their corresponding chapters.

1.6 Bibliography

[1] O. Amft. On-body sensing solutions for automatic dietary monitoring. *Pervasive Computing*, 8(2):62–70, 2009.

[2] M. Andriluka, S. Roth, and B. Schiele. People-tracking-by-detection and people-detection-by-tracking. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1 –8, june 2008.

[3] S. Antifakos, F. Michahelles, and B. Schiele. Proactive instructions for furniture assembly. *Lecture Notes in Computer Science*, Jan 2002.

[4] P. Aoki, R. Honicky, and A. Mainwaring. Common sense: Mobile environmental sensing platforms to support community action and citizen science. *Adjunct Proceedings of Ubicomp*, Jan 2008.

[5] D. Bannach, K. Kunze, J. Weppner, and P. Lukowicz. *Integrated tool chain for recording and handling large, multimodal context recognition data sets*. ACM, New York, New York, USA, Sept. 2010.

[6] L. Bao. Physical activity recognition from acceleration data under semi-naturalistic conditions. Master's thesis, MIT, 2003.

[7] A. Bulling, D. Roggen, and G. Troster. It's in your eyes: towards context-awareness and mobile HCI using wearable EOG goggles. In *UbiComp '08: Proceedings of the 10th international conference on Ubiquitous computing*. ACM, sep 2008.

[8] J. Cheng, O. Amft, and P. Lukowicz. Active capacitive sensing: Exploring a new wearable sensing modality for activity recognition. In *Pervasive*, pages 319–336, 2010.

[9] T. Choudhury, G. Borriello, and J. Lester. Assessing wellness by sensing everyday activities and interactions. *seattle.intel-research.net*, 2005.

[10] T. Choudhury, J. M. Rehg, V. Pavlovic, and A. Pentland. Boosting and structure learning in dynamic bayesian networks for audio-visual speaker detection. In *Proceedings of the 16 th International Conference on Pattern Recognition (ICPR'02) Volume 3 - Volume 3*, ICPR '02, pages 30789–, Washington, DC, USA, 2002. IEEE Computer Society.

[11] I. Chronis, A. Madan, and A. Pentland. Socialcircuits: the art of using mobile phones for modeling personal interactions. *Proceedings of the ICMI-MLMI'09 Workshop*, Jan 2009.

[12] N. Dimakis, J. K. Soldatos, L. Polymenakos, P. Fleury, J. Curín, and J. H. Kleindienst. Integrated Development of Context-Aware Applications in Smart Spaces. *IEEE Pervasive Computing*, 7(4):71–79.

[13] N. Eagle and A. S. Pentland. Reality mining: sensing complex social systems. *Personal and Ubiquitous Computing*, Jan 2006.

[14] K. Forster, P. Brem, D. Roggen, and G. Troster. Evolving discriminative features robust to sensor displacement for activity recognition in body area sensor networks. In *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2009 5th International Conference on*, pages 43 –48, 2009.

[15] K. Forster, D. Roggen, and G. Troster. Unsupervised classifier self-calibration through repeated context occurences: Is there robustness against sensor displacement to gain? In *Wearable Computers, 2009. ISWC '09. International Symposium on*, pages 77 –84, 2009.

[16] H. W. Gellersen, A. Schmidt, and M. Beigl. Multi-sensor context-awareness in mobile devices and smart artifacts. *Mob. Netw. Appl.*, 7(5):341–351, 2002.

[17] D. Gordon, H. Schmidtke, M. Beigl, and G. von Zengen. A novel microvibration sensor for activity recognition: Potential and limitations. In *Wearable Computers (ISWC), 2010 International Symposium on*, pages 1 –8, oct. 2010.

[18] K. Hinckley and E. Horvitz. Toward more sensitive mobile phones. In *Proceedings of the 14th annual ACM symposium on User interface software and technology*, pages 191–192. ACM New York, NY, USA, 2001.

[19] T. Huynh, M. Fritz, and B. Schiele. Discovery of activity patterns using topic models. In *Proceedings of the 10th international conference on Ubiquitous computing*, UbiComp '08, pages 10–19, New York, NY, USA, 2008. ACM.

[20] F. Ichikawa, J. Chipchase, and R. Grignani. Where's the phone? a study of mobile phone location in public spaces. *Mobile Technology, Applications and Systems, 2005 2nd International Conference on*, pages 1 – 8, Oct 2005.

[21] A. Kerhet, M. Magno, F. Leonardi, A. Boni, and L. Benini. A low-power wireless video sensor node for distributed object detection. *J. Real-Time Image Processing*, 2(4):331–342, 2007.

[22] N. Kern, B. Schiele, H. Junker, P. Lukowicz, and G. Tr̈oster. Wearable sensing to annotate meeting recordings. In *Proceedings Sixth International Symposium on Wearable Computers ISWC 2002*, 2002.

[23] N. Kern, B. Schiele, and A. Schmidt. Multi-sensor activity context detection for wearable computing. 2003. European Symposium on Ambient Intelligence.

[24] A. Krause, D. Siewiorek, A. Smailagic, and J. Farringdon. Unsupervised, dynamic identification of physiological and activity context in wearable computing. *Wearable Computers*, Jan 2003.

[25] A. Krause, A. Smailagic, and D. Siewiorek. Context-aware mobile computing: learning context- dependent personal preferences from a wearable sensor array. *Mobile Computing, IEEE Transactions on*, 5:113 – 127, Feb 2006.

[26] N. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. Campbell. A survey of mobile phone sensing. *IEEE Communications Magazine*, 48(9), Sep 2010.

[27] M. Langheinrich, F. Mattern, and K. Römer. First steps towards an event-based infrastructure for smart things. *Ubiquitous Computing Workshop, PACT*, 2000.

[28] J. Lester, T. Choudhury, and G. Borriello. A Practical Approach to Recognizing Physical Activities. *Lecture Notes in Computer Science*, 3968:1, 2006.

[29] J. Lester, T. Choudhury, and G. Borriello. A practical approach to recognizing physical activities. *Proceedings of Pervasive*, Jan 2006.

[30] H. Lu, W. Pan, N. Lane, and T. Choudhury. Soundsense: scalable sound sensing for people-centric applications on mobile phones. *7th International Conference on Mobile systems*, Jan 2009.

[31] P. Lukowicz, J. Ward, H. Junker, M. Staeger, G. Troester, A. Atrash, and S. Starner. Recognizing workshop activity using body worn microphones and accelerometers. In *Pervasive Computing*, 2004.

[32] P. Lukowicz, J. Ward, H. Junker, M. Staeger, G. Troester, A. Atrash, and T. Starner. Recognizing workshop activity using body worn microphones and accelerometers. In A. Ferscha and F. Mattern, editors, *Pervasive Computing*, volume 3001 of *Lecture Notes in Computer Science*, pages 18–32. Springer Berlin / Heidelberg, 2004.

[33] J. Mantyjarvi, J. Himberg, and T. Seppanen. Recognizing human motion with multiple acceleration sensors. In *2001 IEEE International Conference on Systems, Man and Cybernetics*, volume 3494, pages 747–752, 2001.

[34] D. Mizell. Using gravity to estimate accelerometer orientation. In *Proceedings Seventh International Symposium on Wearable Computers IS WC 2002*, 2003.

[35] A. Ofstad, E. Nicholas, R. Szcodronski, and R. Choudhury. Aampl: accelerometer augmented mobile phone localization. *ACM international workshop on Mobile entity localization*, Jan 2008.

[36] G. Ogris, T. Stiefmeier, P. Lukowicz, and G. Troster. Using a complex multimodal on-body sensor system for activity spotting. *Proceedings of the 12th IEEE International Symposium on Wearable Computing*, Jan 2008.

[37] S. N. Patel, T. Robertson, J. A. Kientz, M. S. Reynolds, and G. D. Abowd. At the flick of a switch: detecting and classifying unique electrical events on the residential power line. In *UbiComp '07: Proceedings of the 9th international conference on Ubiquitous computing*. Springer-Verlag, sep 2007.

[38] S. N. Patel, E. P. Stuntebeck, and T. Robertson. *PL-Tags: Detecting Battery-less Tags through the Power Lines in a Building*, volume 5538. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.

[39] D. J. Patterson, D. Fox, H. Kautz, and M. Philipose. Fine-grained activity recognition by aggregating abstract object usage. In *Proceedings of the Ninth IEEE International Symposium on Wearable Computers*, pages 44–51, Washington, DC, USA, 2005. IEEE Computer Society.

[40] A. Pentland. Smart rooms. *Scientific American*, 1996.

[41] D. Ruta and B. Gabrys. An overview of classifier fusion methods. *Computing and Information Systems*, 7(3-4):146–153, 2000.

[42] D. Salber and A. Dey. The context toolkit. *SIGCHI conference on Human Computer Interaction*, 1999.

[43] A. Schmidt, K. Aidoo, and A. Takaluoma. Advanced interaction in context. *Handheld and Mobile Computing*, 1999.

[44] A. Schmidt, M. Beigl, and H. Gellersen. There is more to context than location. *Computers & Graphics*, 23(6):893–901, dec 1999.

[45] D. Siewiorek, A. Smailagic, J. Furukawa, A. Krause, N. Moraveji, K. Reiger, J. Shaffer, and F. L. Wong. Sensay: a context-aware mobile phone. In *Wearable Computers, 2003. Proceedings. Seventh IEEE International Symposium on*, pages 248 – 249, oct. 2003.

[46] M. Stager and P. Lukowicz. Power and accuracy trade-offs in sound-based context recognition systems. *Pervasive and Mobile Computing*, 2007.

[47] T. Starner, B. Schiele, and A. Pentland. Visual contextual awareness in wearable computing. In *Proceeding of the Second Int. Symposium on Wearable Computing. Pittsburgh, October*, 1998.

[48] U. Steinhoff and B. Schiele. Dead reckoning from the pocket - an experimental study. In *Eighth Annual IEEE International Conference on Pervasive Computing and Communications (PerCom 2010)*, Mannheim, Germany, 04/2010 2010. IEEE, IEEE.

[49] E. Tapia, S. Intille, and K. Larson. Activity recognition in the home using simple and ubiquitous sensors. In A. Ferscha and F. Mattern, editors, *Pervasive Computing*, volume 3001 of *Lecture Notes in Computer Science*, pages 158–175. Springer Berlin / Heidelberg, 2004.

[50] S. Thiemjarus. A device-orientation independent method for activity recognition. In *Body Sensor Networks (BSN), 2010 International Conference on*, pages 19 –23, 2010.

[51] K. van Laerhoven and H. Gellersen. Spine versus porcupine: a study in distributed wearable activity recognition. *Wearable Computers*, 2004.

[52] K. Van Laerhoven and H.-W. Gellersen. Spine versus porcupine: A study in distributed wearable activity recognition. In *8th International Symposium on Wearable Computers (ISWC 2004)*, pages 142–149. IEEE Computer Society, IEEE Computer Society, 2004.

[53] M. Weiser. Some computer science issues in ubiquitous computing. *Communications of the ACM*, 36(7), July 1993.

[54] M. Weiser. The coming age of calm Technology. *Xerox PARC Retrieved July*, 1996.

[55] T. Westeyn, P. Presti, and T. Starner. ActionGSR: A Combination Galvanic Skin Response-Accelerometer for Physiological Measurements in Active Environments. *Wearable Computers, 2006 10th IEEE International Symposium on*, pages 129–130, 2006.

[56] T. Westeyn, K. Vadas, X. Bian, T. Starner, and G. D. Abowd. Recognizing mimicked autistic self-stimulatory behaviors using hmms. *Wearable Computers, IEEE International Symposium*, 0:164–169, 2005.

[57] D. Wyatt and T. Choudhury. Creating Social Network Models from Sensor Data. *NIPS Network Workshop*, 2007.

[58] K. ZHANG and F. Pi-Sunyer. Improving energy expenditure estimation for physical activity. *Science in Medicine*, 2004.

2

Placement on the body or in the environment

"If you don't know where you are, even the best compass won't help." -Unkown

The first coarse-grain placement variations we tackle is how to distinguish whether a device is carried by a user or placed in a specific location in the environment. We discuss the impact specific environmental placements have on different sensor modalities. Then we detail a particular solution based on simple sensors routinely used in sensor nodes and smart objects (acceleration, sound). By using vibration and short, narrow frequency "beeps" to sample the response of the environment to mechanical stimuli, no infrastructure is required. Our method works for specific placements such as "on the couch", "in the desk drawer" as well as for general location classes such as "closed wood compartment" or "open iron surface". In the latter case, it is capable of generalizing the classification to locations the object has not encountered during training. We present the results of an experimental study with a total of over 1200 measurements from 35 specific locations (taken from 3 different rooms) and 12 abstract location classes.

Kunze, K. and Lukowicz, P. Symbolic object localization through active sampling of acceleration and sound signatures. *In Proceedings of the 9th international Conference on Ubiquitous Computing*. Innsbruck, Austria, September 16 - 19, 2007.
nominated for best paper.(Acceptance rate: 14%)

To detect wether a device is carried on the body or placed in the environment is just a special case of recognizing the symbolic location of the device (see Fig-

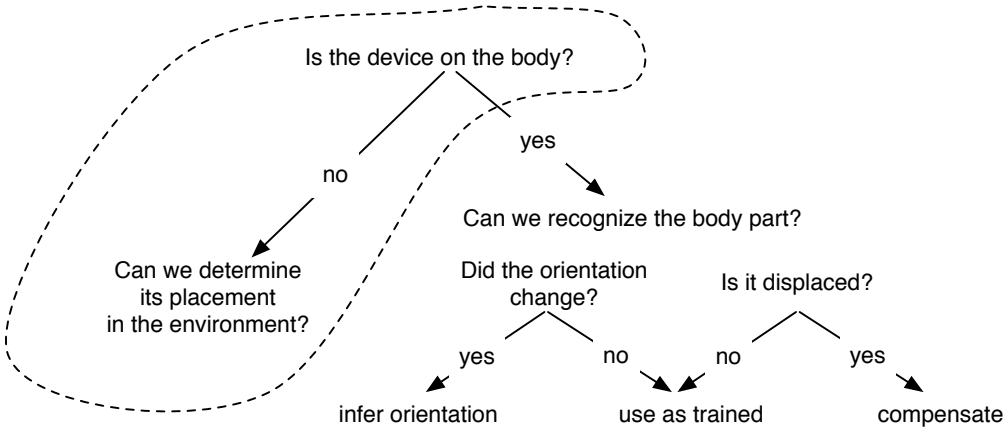


Figure 2.1: Thesis Overview marking the section dealt with in this chapter.

ure 2.1). "Symbolic", in this instance, means assigning a label to a given device placement, e.g. "on the table", "in a draw". The symbolic location of an object can be interesting for a variety of reasons. Most obvious is the "where did I put my x" scenario. This scenario is highly relevant in so called assisted living systems. Such systems use on-body devices for behavioral monitoring and assistance for elderly and cognitively impaired persons. An important concern is to make sure that the user carries the device all the time. This implies checking if the device is on the user's body and, if not, using for example the TV, the radio or the phone to remind him to pick it up. In particular, for cognitively impaired users, it is important to be able to tell the user where the device is located, in case it was lost.

Trivially, whether the user carries a device on the body or not is instrumental to using this device for context recognition. In fact, knowing if the mobile phone is in a pocket, in the hand, or lying on a table has been among the first suggested context sensitive applications [14]. Generally, we can use the location of smart objects as an indication of user's needs. Thus, if a device is put in the drawer where it is usually stored, it is reasonable to assume that it will not be used in the near future and it can go into power saving mode. Going even further the location of a set of objects can be an indication of more general user activity and intentions. Keeping in mind that placement on its own is a valuable information source for context recognition systems, this dissertation sees it more as being a low level part of an inference chain, on which complex systems can be built.

The following assumptions are the basis for the remainder of this chapter:

1. Detecting wether a device is on the users body or not is a specific case of a more general problem inferring the symbolic location of an object.

2. Some context sensitive applications prefer symbolic classifications – "on the shelf", "close to the printer" – to absolute position coordinates.

3. The symbolic location classes introduced later are mainly chosen to show the merits and limitations of the given sensor modalities (acceleration and sound). They are, however, inspired by assisted living scenarios and can be used in such.

To better understand how one can detect the symbolic placement, the next section discusses different, proposed solutions and related work followed by an exploration of environmental impacts on different sensing modalities . Finally, we present our approach of active sampling the environment with a rigorous experimental evaluation.

2.1 Technical Possibilities and Related Work

The simple straight forward way to deal with there environmental placement issues is to integrate sensors directly into the symbolic location. Thus, there is no need to recognize their placement as it is known and fixed after manufacturing. Switches or accelerometers are placed on doors, the stove etc. Prevalent intelligent home scenarios mostly apply this option. Of course, this method entails all limitation of infrastructure-based, fixed sensing.

Determining the symbolic placement of a device can be seen as a specific case of indoor location estimation. Yet, indoor location is known to be a hard problem. Hightower et. al. give a detailed overview about indoor localization techniques [5]. As described above, our work aims at the localization of simple objects in environments with no, or only minimal augmentation. This means that many of the more reliable, standard methods are not applicable. This includes ultrasonic location such as the BAT or the MIT cricket systems which both require extensive instrumentation of the environment with ultrasonic transceivers [18, 12]. In addition ultrasonic system require free line of sight and will fail to locate objects in closed compartments. This means that infrastructure free, relative positioning methods based on ultrasonics are also unsuitable [4]. Cost and effort also make it infeasible using complex time of flight based radio frequency (RF) methods such as the commercial UBISENSE ultra wide band system [1]. This holds, as well, for radio frequency identification (RFID), requiring a reader or tag to be put on every location which needs to be recognized.

Simple Beacon Based Systems Much work has been put into localization based on simple RF beacons, often based on standard communication systems such as

[1] www.ubisense.net

Bluetooth, Zigbee and of course WLAN [9, 1, 8]. This includes a wide body of work on positioning in wireless sensor networks [2]. In particular, work based on low power radio systems is clearly relevant to object localization. This is more a complementary rather than a competing approach. Such systems are virtually all based on signal strength, which is inherently unreliable in complex, indoor environments. As a consequence, they are predominantly used for room level location (determining which room or large room segment a sensor node is in). This is not sufficient for the type of symbolic location targeted by this paper. However knowing approximate physical location can be used to constrain the search space for our symbolic location method.

Indirect Localization with Sensor Signatures Both sound and acceleration have been previously used in location related research. Scott et. al. present a technique for performing accurate 3D location sensing using off-the-shelf audio hardware [15] . Van Kleek et al. show some work in this direction, using sound fingerprints to detect collocation [7].

The general concept of using acceleration signatures to extract location related information can be traced to the 'Smart-Its Friends' paper, [6]. Building on this idea, Lester et. al. have demonstrated how to determine if a set of devices is being carried by the same person by correlating their acceleration signatures [10]. In Chapter 3 we take this concept even further, showing how acceleration signatures can be used to determine where a user is carrying a device.

The most direct relation to the work presented in this chapter is a patent by Griffin [3] titled "User hand detection for wireless devices". It proposes to use vibration detected by an acceleration signal to determine if a mobile phone is in the user hand, in a holster or in a holder.

2.2 Environmental Placement Impacts

It is reasonable to wonder what impacts a specific placement has on a given sensing modality. We discuss these placement impacts on commonly used pervasive sensing modalities, namely, motion, sound and radio waves.

Motion sensors will receive no to little activation when they are placed in the environment compared to being worn on the body. The placement itself however can entail characteristic movements, e.g. the vibrations of a fridge cooling aggregate.

Changes in device placement affect sound to a much higher degree. Different environments have their distinct sounds. As shown in Figure 2.2, when the microphone of a device is obstructed by specific material, frequency dampening is to be expected. This is bad, if we try to classify sounds in the environment. On the other hand, this dampening is distinct for the given placement, therefore it

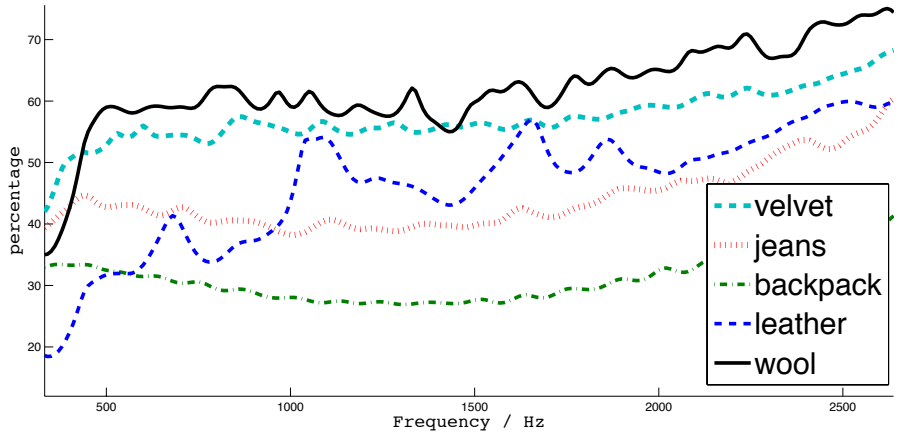


Figure 2.2: Sound dampening depending on different fabric types. "White noise" in the frequency range from 500 to 2500 Hz is played on a regular pc speaker and recorded by an iphone 3gs. The iphone speaker is covered with the given fabric types. The plot shows the percentage difference between the clean recorded signal and the recordings obstructed by fabric. Each fabric has a distinct absorption spectrum.

can be used to determine the location. This fact is explained in closer detail when we look at the technical background of our method in section 2.3.

The environmental impact on radio waves is well known and applied. There are several commercial efforts and an extensive research body using these impacts, for example, for indoor location technologies [5, 4]. Radio waves, however, are not only influenced by the environmental placement, but also by their immediate surroundings; radio waves in the 2 GHz range, for example, are obstructed by large amounts of water (e.g. the human body). As seen in Figure 2.3, the signal strength from several wifi access points is distinct in different locations. However, the signal strength also highly depends on the on-body placement of the device recording it.

As these examples illustrate, environmental placement affects sensor signals in a complex way. There are specific locations that can be distinguished using just passive sensor data, yet this method is very limited regarding general placement inference. Thus, the approach to actively sample the environment (i.e. the device itself emits a given stimulus and analyses the response of the environment) seems to be far more promising.

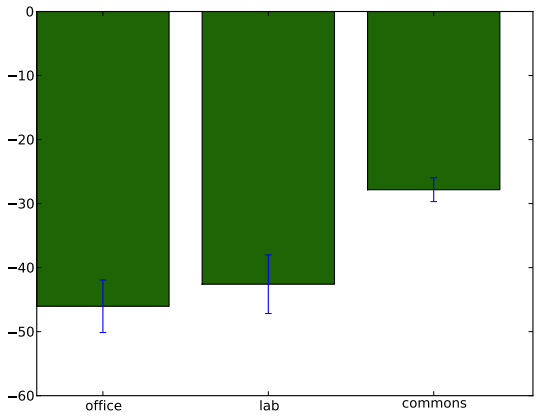


Figure 2.3: Wifi signal strength recorded by a mobile phone in different rooms of the same building. The phone is put on a desk for 5 minutes stationary in several rooms. The experiment is repeated 15 times. We show the mean average signal strength in dBm for three rooms: office, lab and commons. The difference in strength between rooms is statistically significant.

2.3 Theoretical Background

Our method is derived from the observation that a ringing mobile phone sounds differently depending on where it is located. Whereas a phone in a jacket pocket sounds 'dampened', a phone on a metal cabinet can make the entire cabinet resonate. This is true for a ringing phone as well as for a merely vibrating phone. We, thus, propose to use sound from a built-in speaker and vibration from a built-in vibro-motor to create a mechanical "excitation" of the environment and analyze the response with an accelerometer and a microphone.

In abstract terms, the above method is about analyzing the response of the environment to a mechanical "excitation" with different frequencies. By vibrating the device we provide a low frequency (a few Hz) and relatively high intensity (as compared to sound) source of excitation. By emitting fixed frequency "beeps" we generate different, low intensity high frequency stimuli. The accelerometer detects the low frequency response (in our case up to 15Hz due to a device sampling frequency limit of 30Hz), the microphone the high frequency part.

The response to the above stimuli falls into several categories. First, we receive a low frequency response. It is directly mechanically coupled to the vibrating object and can be detected by the accelerometer. This response can range from a more or less complete absorption of the vibration energy (e.g. when the object is lying on pillow) to a resonant response where the surface on which the device

is lying, joins in the vibration. This fact contains information on two placement properties. For one, it can reveal if, and how the device is fixed (in the hand, in a tight pocket, lying freely). In addition, it reveals the hardness and elasticity of the surface on which the device is placed. This information can be expected to reliably distinguish between soft surfaces (e.g. a sofa) and hard ones like a table. Distinction between several similarly hard surfaces (e.g. metal and stone) is difficult.

Second, we get a high frequency response from the vibration, which is essentially sound from the device hitting the surface. Assuming the placement of the device does lead to this kind of response (it will not, if the device is in a soft pocket or say hanging), it is quite location specific. The sound depends not only on the surface material but also on the overall structure. Thus a small, solid cube will produce a different sound than a large thin surface, even if both are made of the same material. Finally, objects light and close enough to the device to be influenced by the vibration (e.g. a key chain) might also contribute to the sound. In general, this is a source of noise rather then usable information.

Third, we get a high frequency response from the beeps which is given by the absorption spectrum of the environment. [2] Clearly this response is only useful if it comes from the immediate vicinity of the device. This can either be the surface on which the device is lying or, if the symbolic location is a closed compartment, the walls of this compartment (see sections 2.4 and 2.7 for a discussion of microphone placement issues).

Table 2.1: Frequencies and their absorption by the selected material, given as a fraction of perfect absorption[11].

frequency	128 Hz	256 Hz	512 Hz	1,024 Hz	2,048 Hz	4,096 Hz
concrete unpainted	0.010	0.012	0.016	0.019	0.023	0.035
brick wall painted	0.012	0.013	0.017	0.020	0.023	0.025
carpet on concrete	0.09	0.08	0.21	0.26	0.27	0.37

It is well known that the acoustic absorption spectrum is a distinct material property. The topic has been extensively studied in the context of musical instruments and sound isolation in construction [11]. Typically, the absorption is given at discrete frequencies as a fraction of the perfect absorption of an open window (lack of any reflecting surface) of equal area. As an example, we consider the coefficients in Table 2.1. This clearly demonstrates that, in principle, even seemingly similar materials can be separated with a small number of discrete frequencies.

[2] Note that the absorption also influences the sound caused by the device vibration.

2.4 Approach

Procedure Description The proposed method consists of two parts. Each part can be used individually or in combination with the other.

The first part is based on vibrating the device using a vibration-motor of the type commonly found in mobile phones. During the vibration, which last a couple of seconds, motion data is recorded with an accelerometer and sound with a microphone. The motion and sound signals are used separately for an initial location classification using standard feature extraction and pattern recognition methods. The final classification is obtained through fusion of the two classification results.

The second part is based on sound sampling. The device emits a series of beeps, each in a different, narrow frequency spectrum. The microphone is usually positioned is such a way that it receives only little energy directly from the speaker. Instead a significant part of the energy comes from reflections from the immediate environment (see bellow for a more detailed discussion). For location recognition the sound received from the different beeps is compared.

When the two parts are used together, the corresponding results are fused using an classifier fusion method.

Applying the Procedure: Specific Locations vs. Location Classes Our method provides information on abstract properties such as surface material as well as information on properties characteristic of a single specific location (e.g. a solid cube vs. large surface with several legs). As a consequence this chapter investigates two different usage modes of our method:

1. "Specific Location Mode". In this mode, we train the system on concrete locations such as a specific table or a specific chair. The advantage of this approach is that the user is provided with exact location information. The main disadvantage is the effort involved in training each individual location. In addition, there is the question of being able to distinguish a large enough number of locations to satisfy relevant applications.

2. "Abstract Location Class". In this mode, we group locations into abstract classes. The two main criteria are the surface material and being open (e.g. tabletop) or closed (e.g. inside a cupboard). In this mode the system is trained on several instances of each class. It is then able to recognize other arbitrary instances of this class. Thus the training problem is avoided, as the system can pre-trained at 'production time' and given to users without the need for further training. The disadvantage lies in the less exact location information, which has to be further interpreted and/or combined with additional information to find out where the object is actually located.

Issues to Consider

Microphone and Speaker Placement As described above, for the analysis of the absorption spectrum we must ensure that the sound emitted by the loudspeaker is reflected from the surface on which the device is lying and/or, in case of the symbolic location being a closed compartment, from the compartment walls. The second part is trivial. The first implies an appropriate placement of the microphone and the speaker. Optimally the speaker and the microphone should be located close to each other on the side of the device, preferably (but not necessarily) facing downwards with a sound proof barrier blocking the direct sound path between them. The main problem in implementing this type of setup is the definition of "on the side" and "downwards". In the worst case, we could be dealing with a cubic or round object with no preferred "down" or "side". For such an object, two speakers located at a 90 degree angle need to be used to ensure that there is always a sidewards facing one. Our experiments (see section 2.6) indicate that the position of the microphone is less critical and we achieved good results despite the microphone facing upwards, so that one microphone might suffice.

Variations within Symbolic Locations Many symbolic locations such as "table" or "desk" have considerable physical dimensions. This means that the response to the mechanical stimuli may be subject to spatial variations. For example, the low frequency response to vibration (acceleration data) may be different over the leg of a table than in its middle. Similarly, on a table adjacent to the wall, the response to the "beep" will vary depending on how close to the wall the device has been placed. As a consequence, for both, training and testing, a sufficient number of random physical locations must be sampled for each symbolic location (as has been done in experiments described in section 2.6).

Number of Relevant Locations Clearly there are limits to how many locations can be reliably recognized. In common environments such as home or office, there are many places where objects can be put. The question is, whether the number of locations that can be distinguished is sufficient to be useful in relevant applications. An authoritative answer to this question can only be found through an analysis of specific applications. Subsequently, we make no claim for such an answer, we focus on the technology merits instead, demonstrating the following:

1. Our system shows reasonable recognition performance even using the combined data set of 35 locations. In our experiments these are collected from 3 rooms. It seems unlikely that this would not be sufficient covering all relevant symbolic locations in a single room. At the same time, room level location of RF enabled sensor nodes is a manageable problem.

2. Provided that an adequate number of sufficiently abstract classes is chosen, the issue of the number of locations is avoided by the "abstract location classes" usage mode. In the experiments, we demonstrate near perfect recognition for 7 and reasonable results for 12 classes. The type of classes used in the experiments ("open wood surface", "closed wood cabinet" etc.) is clearly abstract enough to describe a large number of locations.

Sensor Requirements In the introduction we have stated our aim of developing a method suitable for smart objects. Accelerometers and a microphones are among the most widely used components in small sensor nodes. Small loudspeakers capable of emitting beeps are also commonly integrated in those nodes. As will be described in section 2.5 we work with frequencies between 500 and 4000 Hz, which can be handled by small, cheap speakers and microphones. Finally, although vibration motors have so far not been used in sensor nodes, they are available in sizes around 1cm and smaller (see figure 2.11a) at reasonable costs.

In summary it can be said that the proposed sensor configuration is compatible with the target domain of small, cheap smart objects.

Complexity Any method that is to be deployed on low end sensor nodes and smart objects needs to be resource conscious. However, when considering the method proposed in this paper it is important to remember that it is not meant for continuous tracking of a moving device. Instead we assume that the method runs once after the acceleration sensor has detected that the device has been moved and is left to rest. Therefore, speed and power efficiency of the algorithm are not so essential. We just need to show that with typical resources available in such nodes it is feasible to either perform the required computation or transmit the data to a remote server for processing. For the sake of simplicity, we restrict ourselves to the communication requirements of the raw data. With 16 bit resolution and the sampling rates given in section 2.5 we require a data rate of about 130 Kbps for the sound and a about 5 Kbps for the acceleration. These rates have to be sustained for a total of 13 seconds.

With respect to online execution, we merely point to related work by our group in which we have studied implementations of sound and acceleration based activity recognition[17]. With sampling rates, features and classifiers similar to the ones proposed in this paper we were able to demonstrate power efficient execution on nodes using the TI MSP 430 microcontroller with less then 100K of RAM. Therefore, a sensor node should be able to execute the proposed method –or at least computing most of the features (in particular the FFT)– to avoid transmitting the raw sound data.

2.5 Recognition Method

As described in section 2.4, our approach can be divided into two distinct methods, mechanical vibration and sound sampling.

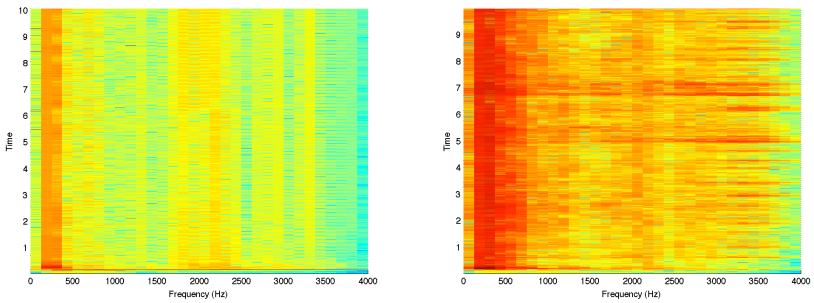
Table 2.2: Features used for frame-by-frame classifications

Feature Name	Description
Standard features	Zero Crossing Rate, median, variance, 75% percentile, inter quartile range
Frequency Range Power	computes the power of the discrete FFT components for a given frequency band.
Sums Power Wavelet Determinant Coefficient	describes the power of the detail signals at given levels that are derived from the discrete wavelet transformation of the windowed time-domain signal. This feature has successfully been used by [16].
Root Mean Square (RMS)	$\sqrt{\frac{1}{N} * \sum_i x_i^2}$, with N the number of samples in a sliding window, and x_i the i'th sample of the window.
Number of Peaks	The number of peaks in the window with different thresholds, low medium and high.
Median Peak Hight	The median of the peak hight.

Vibration

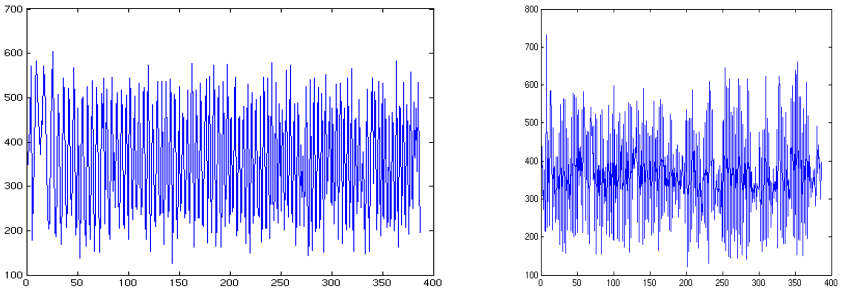
During the vibration phase, the device itself records the sound and the acceleration. Figure 2.4 and Figure 2.5 show some signal examples for sound and acceleration recorded in different symbolic locations. Classification is performed separately on each signal and the information of the two modalities is combined on classifier level (see Section 2.5).

Vibration Sound Processing About 30 individual features are calculated over a 500 msec. sliding window (250 msec. overlap). We pick 5 based on initial tests:



(a) (b)

Figure 2.4: The vibration sound spectrum recorded for a carpet, on the left, and a desk, on the right.



(a) (b)

Figure 2.5: The acceleration norm from bed (a) and the stereo (b). On the x-axis are the samples (30 per second) and on the y-axis the magnitude. The accelerometer on the Nokia 5500 Sport discretizes the values between +/- 600, 250 being 1 g.

the zero crossing rate, the frequency range power, 75%Percentile, sums power wavelet determinant coefficient and the median (see Table 2.2). We trained common machine learning algorithms using these features, e.g. K-NN, Naive Bayes, C 4.5. As all machine learning algorithms provid comparable results and we need a ranking mechanism for the different classes, we use the Naive Bayes classifier in the following. The frame-by-frame output provided by the Naive Bayes classifier is smoothed using a majority decision over the entire length of a single vibration phase. We also perform experiments using Hidden Markov Models either on the features calculated in the 500ms windows or on the classifier output of the frame by frame classifier. Since none of the above produced significant improvement, we use the less computationally intensive majority decision.

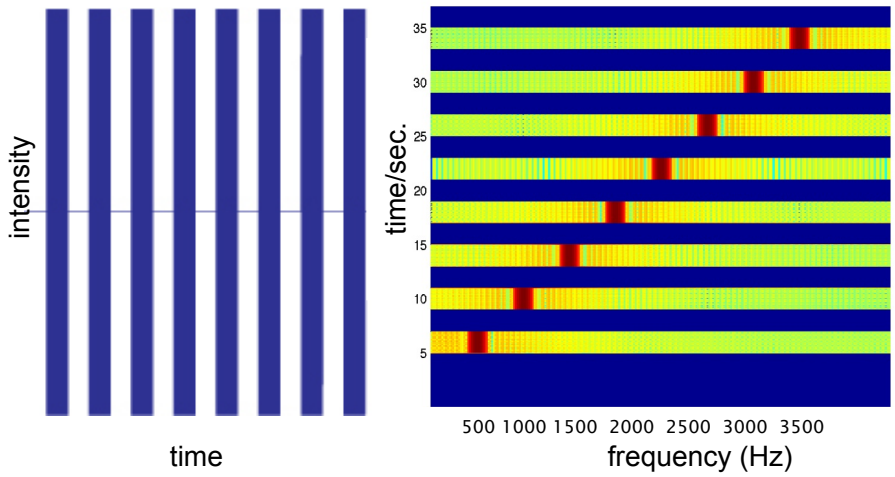
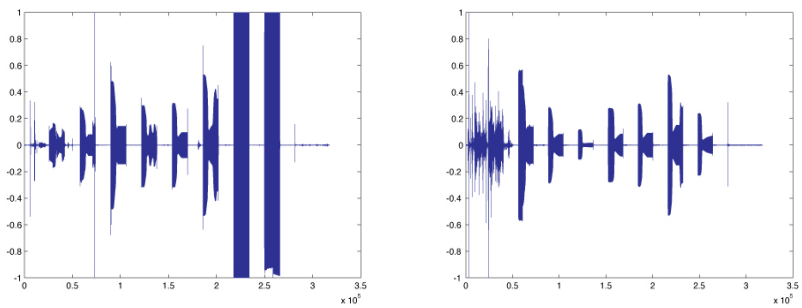


Figure 2.6: The played fingerprint audio, with the distinct frequencies, on the left in the time domain, on the right in the frequency domain.



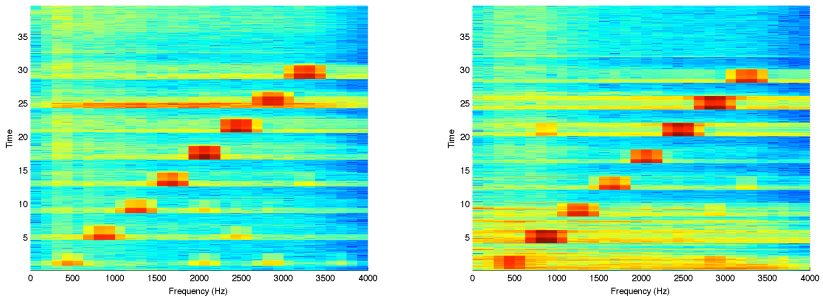
(a) (b)

Figure 2.7: The audio fingerprints for drawer and backpack in the time domain

Vibration Acceleration The process described above for the vibration sound is essentially repeated for the acceleration. The only differences are the length of the window (1 sec with 0.5 sec. overlapping) and the final feature set (variance, the RMS, number of peaks, median peak height, the 75%Percentile, inter quartile range).

Sound Sampling

The active sound sampling procedure differs from the vibration method in several ways. We know from literature (see section 2.4) that few discrete frequencies between a few hundred and a few thousand Hz are enough to separate a large



(a) (b)

Figure 2.8: The audio fingerprints for drawer and backpack in the frequency domain

range of materials in terms of their absorption coefficients. Therefore, we select 8 discrete, equidistant frequencies between 500 and 4000. Figure 2.6 shows the signal emitted by the speaker in time and frequency domains. The frequency range choice is dictated by the specification of small, cheap speakers (not capable of very low frequency tones) and available sampling rates -the used mobile phone is just capable of sampling with 8000 Hz. Some sample recordings for different symbolic placements can be seen in Figure 2.7 (in the time domain) and in Figure 2.8 (in the frequency domain). From the recorded beeps we first isolate 8 frequency fingerprints using a variable intensity threshold. As features we empirically select RMS, frequency range power and the sums power wavelet determinant coefficient using the mutual information metric. These features are determined out of 30 features calculated using a 200 msec. sliding window with 150 msec. overlap.

The features of all 8 frequency prints are combined into one feature set. This means that a feature instance contains the calculated RMS etc. of each frequency band. The rest of the procedure is identical with the vibration recognition (frame by frame classification using C 4.5 and majority decision).

Fusion

The two main approaches to fusion are signal/feature level and classifier level fusion. Feature level fusion works best for features that are computed at the same sampling rate (sliding window size). This is not the case for the three recognition modalities described above. As the different window sizes are determined heuristically to produce best results for each modality, dropping them for the sake of fusion makes little sense. As a consequence, no direct feature level fusion is investigated. We, however, investigate a fusion approach based on the results of the frame by frame classification (see Figure 2.9). This can be viewed as a kind of feature level fusion, since its result is input to the majority decision. Thus,

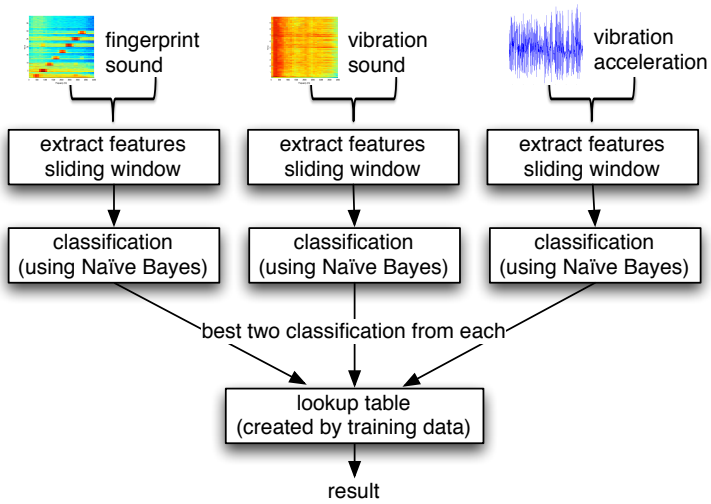


Figure 2.9: The fusion recognition method as overview.

we compute the majority decision for an event over the frame by frame results from all three modalities put together, instead of computing it for each modality separately.

In terms of classifier fusion we opt for a Bayesian Belief Integration method (see [13] for an overview of classifier fusion methods). This method uses the confusion matrix obtained from testing the classifiers on the training data set to determine class probabilities for different combinations of classifier outputs. This allows the system to take into account the peculiarities of each classifier. With just 3 classifiers and a constrained number of classes it is also computationally tractable. If the number of classes is increased the method could be replaced by e.g. logistic regression.

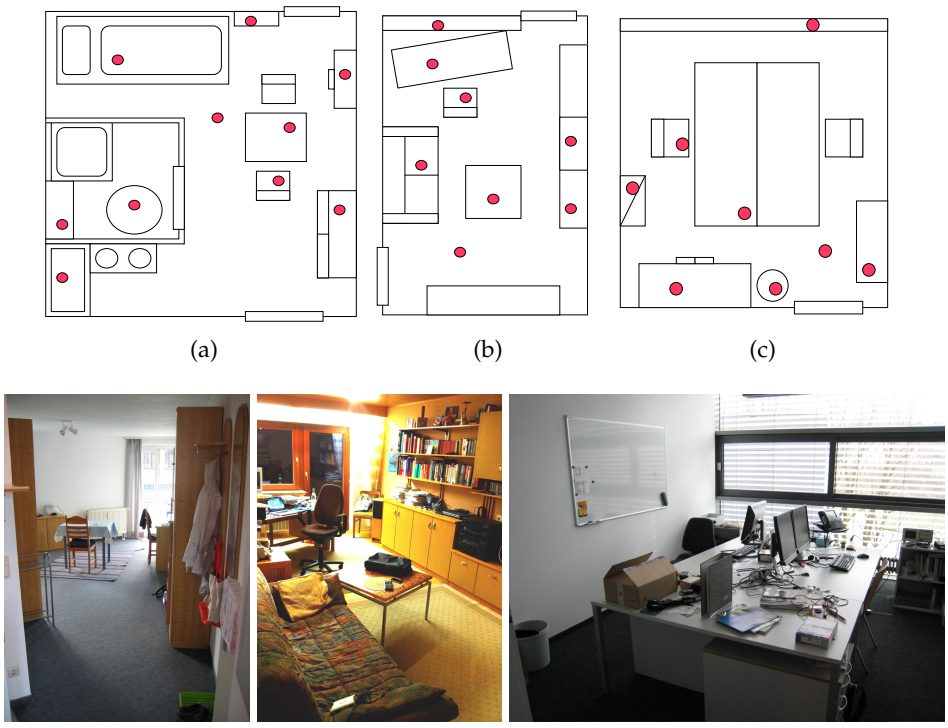
2.6 Experimental Validation

During the evaluation, we design and conduct experiments for both modes of our method, specific and abstract location. We always introduce the details to the specific location mode first, going into details about the scenario and procedure.

Validation Scenarios

Specific Location Mode

As basis for our study, we pick three scenarios: an office, a living room, and a one room student apartment. In each scenario a set of obvious locations for placing objects are selected. These include the furniture present in the room (both open



(a) (b) (c)

(d) (e) (f)

Figure 2.10: The top figures show schematics of the rooms used for the experiments. The symbolic locations we try to detect are marked in red for the apartment living room and office. Below the schematics there are actual photos from the locations.

such as table or sofa and closed such as cupboards), the floor, the window ledges and additional objects such as the stereo. In the office scenario we also include three pockets (two different pockets of a jacket and a jeans pocket), the inside of a backpack and a suitcase as well as a trashcan. A full listing of the investigated location is given in Table 2.3 and illustrated in Figure 2.10. There are 16 locations in the office, 9 in the living room and 10 in the apartment (total of 35).

We record 30 experimental runs on each specific location (a total of over 1000 events), each time randomly varying the exact position of the recording. The object is placed according to positions drawn randomly from a uniform distribution. Of the 30 runs, 10 are randomly picked to train the classifiers, the remaining 20 are used as test set. Evaluation is performed first on each individual scenario (assuming that room level location could be obtained by other means). We also perform an evaluation on a data set containing all locations from the three scenarios, to see how our method behaves when the number of locations increases.

Table 2.3: Chosen symbolic locations and abstract location classes. The letter in front is the identification for the individual confusion matrix plots presented later in the paper. The letter in brackets behind the class description, is the identifier for the confusion matrix plot over all 35 locations. In j. , o. j. and tr. pocket stand for inside jacket, outside jacket and trousers pocket.

Office		Living room	Apartment	Surfaces
a. backpack(a)	k. in j. pocket(C)	a. desk(h)	a. bath carpet(f)	a. polster open
b. cupboard(z)	l. tr. pocket (e)	b. floor(u)	b. bed(p)	b. glass open
c. suitcase(w)	m. cartbox (F)	c. sofa(n)	c. chair(b)	c. iron open
d. drawer(t)	n. ledge (H)	d. table(A)	d. desk (wood) (l)	d. stone closed
e. desk(D)	o. chair (v)	e. chair(c)	e. radiator(d)	e. wood closed
f. top drawer(E)	f. drawer (m)	f. ledge(k)	f. glass closed	
g. cabinet (x)	p. shelf (i)	g. ledge (G)	g. carpet floor(B)	g. iron closed
h. o j. pocket(j)		h. stereo (s)	h. cupboard(g)	h. metal open
i. trashcan(I)		i. tv (j)	i. drawer(q)	i. polster closed
j. carpetfloor(r)			j. wardrobe (o)	j. stone open

Abstract Location Type Mode

The abstract location types are defined according to the surface material and the location being open (e.g. a table) or closed (e.g a cabinet or a drawer). As shown in Table 2.3 this lead us to 9 classes including most typical surfaces (wood, glass metal, stone, cushion). To get a sufficient number of different instances for each class we record the data in a furniture store. For every abstract class we pick 6 different pieces of furniture. Two recordings are conducted on each specific piece of furniture leading to 9 data points per abstract class and a total of 144 events. For the evaluation two pieces from each class (four events per class) are picked for training and 4 (8 events per class) are retained for testing. This is consistent with the envisioned application use case in which the user would be given a device "factory pre-trained" for each class and use it to recognize instances of the class not seen by the system before.

Experimental Procedure

Setup

For the experiments, we use the Nokia 5500 Sport, see Figure 2.11. It is a mobile phone of Nokia's third S60 series, equipped with an accelerometer and an extra loudspeaker. The mobile is able to run C++, Java and python code. We record data using python. The evaluation is done in batch processing using a mixture of Python, Matlab scripts and Java code, mainly consisting of the Weka machine learning package.

(a) (b)

Figure 2.11: A picture of a common vibration motor and the extra loudspeaker on the Nokia 5500 Sport.

Data Acquisition

An experimental run consists of the following steps. First the phone is placed on a random spot on a particular location. Using a uniform distribution, the actual spot is determined randomly. Then the measurement is started. While the mobile vibrates for 5 sec. lying face up on the surface, the sound from the phone vibrating is sampled by the on-board microphone with 8 kHz and the acceleration with 30 Hz. After the vibration measurement is done the mobile plays the sound sample consisting of 8 beeps in distinct frequencies from 500 to 4000 Hz in 500 Hz steps (as seen in Figure 2.7). Each tune is 1 sec. long. While the mobile plays this using the extra loudspeaker, the python script records the sound with 8000 Hz over the built-in mobile microphone. The loudspeaker faces the surface, as depicted in Figure 2.11. We get around a problem of accessing full-duplex mode in python on the Nokia phone by using the music player and the extra speaker.

Experimental Results

The recognition performance for different scenarios, experiments and recognition modalities are summarized in Figure 2.13a for the three individual scenarios of the specific location mode and the abstract location class and in Figure 2.13b, combining all 3 locations and second/third best voting. Additionally, examples of confusion matrices are visualized for the office scenario, the combination of all three specific location mode scenarios and the abstract location type mode in Figures 2.15a, 2.15c and 2.16 respectively. The dependency of the classification accuracy on the number of training events can be seen in Figure 2.12 for the different scenarios.

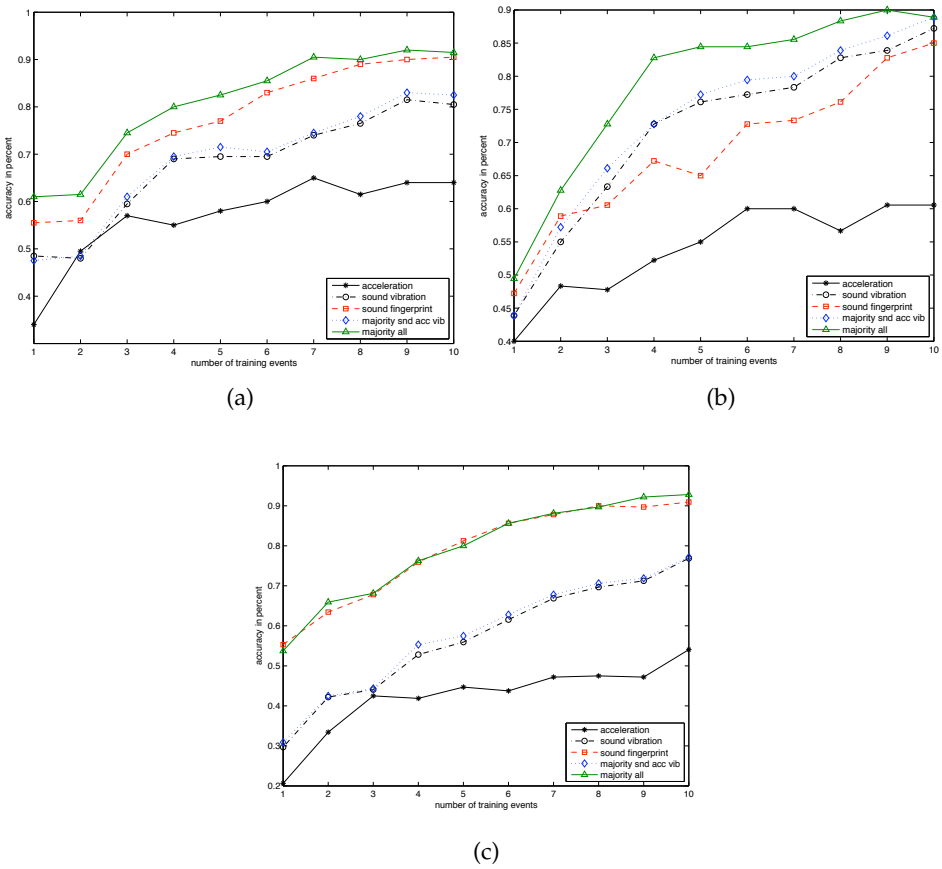
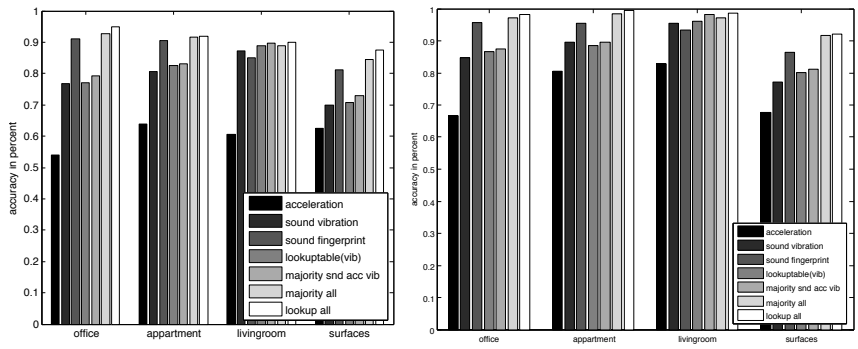


Figure 2.12: The classification accuracy depending on the number of training events and different sensing modalities for the appartment (a), living room (b), and office scenario (c).

In the more detailed discussion of the results given below and the some of the figures we at times discuss "2nd best evaluation" or "3rd best evaluation". This refers to cases where the correct class is among the 2 or 3 first picks of the classification system.

Office In the office scenario, 14 of the 16 locations can be classified with near perfect accuracy. The single biggest confusion is between the pocket on the inside of the jacket and the one on the outside. This is plausible and to be expected. An unexpected result is the poor recognition of the metal window ledge. It is confused with the cart-box, the top shelf and the chair.



(a) (b)

Figure 2.13: Barcharts for living room, office, apartment, and abstract classes using just the first result of the classification (a) and allowing the 2nd best vote (b)

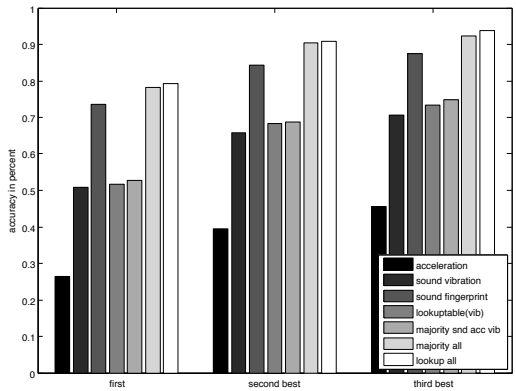


Figure 2.14: Barchart for office living room, apartment and all combined including 1st 2nd 3rd best

The classification accuracy is 54% using the event-based acceleration classifier, 77% for vibration sound, 91% for the sound sampling, 77% and 79% for the vibration fusion cases. We reach up to 93-94% for the majority decision and lookup-table fusion using all modalities. The sound sampling is the best non-fusion method with 91%. The "2nd best evaluation" pushes the correct classified up to 96%.

Living room In the living room scenario, most of the samples from 7 of the 9 locations can be classified correctly. A lot of the sofa instances are confused with the chair, as the chair is also padded. This is the worst confusion occurring.

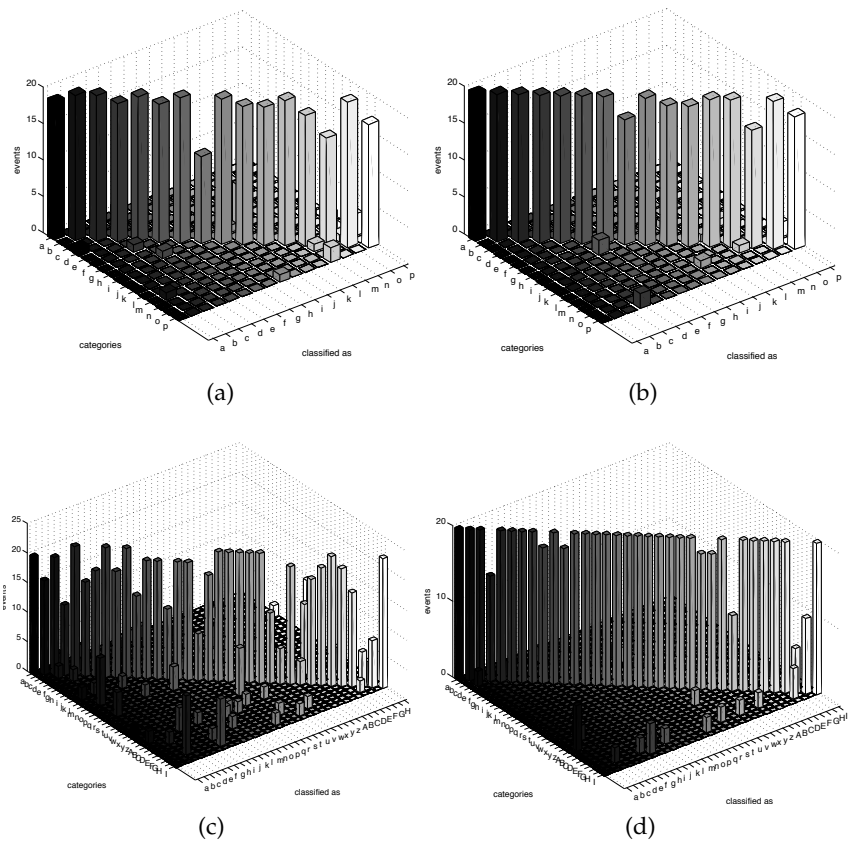


Figure 2.15: The confusion matrix (a) of the office using the lookup-table fusion compared with the confusion matrix in (b) using the second best locations in addition to the lookup-table. The same is depicted, below only for all the 35 different semantic locations. Figure (c) shows the classification of the lookup-table fusion, whereas Figure (d) shows the lookup-table fusion considering up to the 3rd best.

Again the classifiers perform poorly for window ledge category. The living room classification accuracy starts with 60% for acceleration alone, and goes up to 87% for the vibration sound. In this scenario, the sound sampling is worse than the vibration methods at 85%. This explains also why the fusion methods on top of the vibration work so well and are nearly as good as the fusion over all methods, at 88 and 89% respectively. The fusion over all methods is just 0.5% better, namely 89.5%. Only a very small number of events (one to two) are corrected by this fusion. In the "2nd best evaluation" the accuracy ranges from 66% for acceleration alone, up to 97% for the lookup-table fusion over all methods. Here also, the acceleration and sound vibration fusion do extremely well with 93% and 96%.

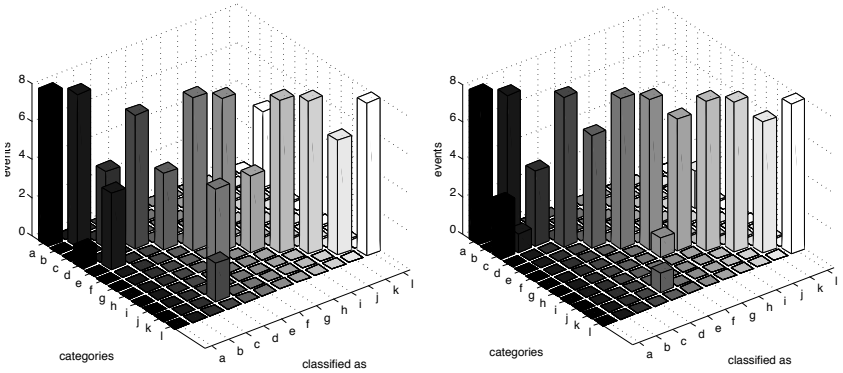
Apartment In the apartment case, the worst miss-classification happens in the cupboard class, which is confused with the desk. Both are made out of the same wood. The radiator class is also confused with several other classes. Here the acceleration accuracy is at 65%, the vibration sound at 81%, sound sampling at 90%. The fusion using just the vibration method is at 82% and 84% respectively. As with all the fusion examples the lookup-table performs slightly better. Finally, the fusion techniques on all 3 modalities are all over 90%. Taking a look at the "2nd best evaluation", there the accuracy ranges from 80 % for acceleration to up to 99% for the look-up table over all three classifiers.

Combined over all rooms (35 classes) As expected, more classes signify a worse classification rate. The ledge classes perform poorly, even in the 2nd and 3rd best evaluation. Also, one of the table classes does badly and is confused with several other classes. The classification accuracy over all 35 semantic locations is expectably lower than those of the single scenarios, ranging from 26% for acceleration, 51% for vibration sound, 74% for sound sampling, over 52% for the vibration fusion, up to 78% for the fusion of all methods. The 2nd and 3rd best evaluations look considerably better. Second best is up to 90%. Third best reaches 94%.

Abstract Location Classes For the abstract classes, the iron and wood classes are easily confused, as are the stone and glass. Acceleration classification alone performs reasonably well, at 63% compared to the other scenarios. Sound vibration is better at 69%. As nearly always, sound sampling performs better than the vibration method, at 81% accuracy. Regarding the fusion techniques, there is also nothing surprising. The vibration fusion majority decision is at 70%, the vibration lookup-table around 71% accuracy. The two fusions based on all methods are at 83% for the simple majority decision case and 86% for the lookup-table. Allowing the second best classification method, one can stem up the performance to 92% for the lookup-table fusion method.

Lessons Learned and Implications

Overall Performance The performance of the system is extremely inhomogeneous with respect to the classes. There is a large proportion of classes for which the classification is perfect or near perfect, and a small one with very poor performance (see confusion matrices in figures 2.15a, 2.15b, 2.15c and 2.15d. As a consequence the overall recognition accuracy figures are strongly influenced by a few classes. This is best exemplified by the abstract location type confusion matrix and 3rd best evaluation of the combined specific location classes. As can be seen in the plots 2.15a, 2.15b, 2.15c and 2.15d the former has 8 perfect or near perfect



(a) (b)

Figure 2.16: Confusion matrix (a) of the abstract classes compared with the corresponding 2nd best confusion matrix in (b)

classes, 1 reasonably good class and 3 very poor ones. The latter has 31 perfect to very good (27 perfect) classes, 1 mediocre one and 3 very poor classes.

Class by Class Performance For some of the classes such as the inside and outside pocket, poor performance is expected, as they are included to test the limits of the system. In fact the recognition for these locations is better than expected. Better than expected recognition has also been achieved in a number of locations that were included as 'hard cases' such as the backpack and the trousers pocket. Surprising is the poor performance of the window ledge and the radiator. At this stage we have no verified explanation. One possibility is a spatial inhomogeneity of those symbolic locations. On the ledge, sound sampling is certainly different depending on whether the speaker faces the window or faces away from it.

Value of the 2nd and 3rd Best Evaluation The performance of the system is particularly appealing for applications that can accept a choice of two or three most probable locations as system output. This has already been mentioned for the case of 3rd best evaluation of the 35 combined symbolic locations. For the other data sets even allowing just the 2nd best pick produces close to perfect recognition for the vast majority of classes.

Value of Different Classification Modalities While it is to be expected from the discussion in 2.4 that sound sampling produces the best results and acceleration the poorest, the difference between the two is larger than we expected. In particular, the fact that in most cases little is gained by adding acceleration and vibration

Figure 2.17: The three different mobile phones used for sensor comparisons: the iphone 3gs, nokia n95 and the htc desire.

sound to the sound fingerprint is surprising. On the other hand, combining vibration sound and acceleration often produces significant gains.

Significance of Training Set Size For the specific location mode the user needs to train the system for every single relevant location. Thus the training effort is a significant issue. As shown on the example of the office scenario in figure 2.11b the system starts to display significant recognition performance at around 5 training examples and stagnates at about 10. We have found this behavior to be typical for all the specific location mode scenarios.

2.7 Excursion: Sensor Dependencies

The successful application of our method is highly dependent on the microphone/ speaker placement and the type of vibration motor. A quick experimental setup illustrates this dependence.

We use 3 regular smartphones, depicted in Figure 2.17 and pick 4 locations from the living room scenario: desk, floor, sofa and stereo. Each mobile performs the experimental procedure outlined in the section above 5 times.

The classification results in Table 2.4 show a high dependency between the speaker /microphone placement and the accuracy. Each phone shows respectable results with microphone and speaker placed towards the surface using the sound fingerprint. However, if the phones are placed with the microphone on the top

Table 2.4: Classification comparison for 4 locations from the living room scenario using frame-by-frame classification.

mobile	htc desire	n95	iphone 3gs	N 5500 Sport
fingerprint	45 %	60 %	47 %	100 %
fingerprint (upside down)	92 %	87 %	90 %	100 %
vibration	45 %	79 %	23 %	84 %
vibration (upside down)	43 %	83 %	25 %	87 %

(as a phone is regularly put on a desk), the rates vary strongly, with the N5500 being by far the best, as it has a separate speaker still facing the surface.

Another important lesson to learn: the vibration classification seems not that affected from the rotation of the devices. Yet, the vibration motor and intensity seem crucial here. The HTC and iphone are equipped with motors operating at a far lower intensity compared to the ones built into the Nokia models. This is also the reason for the lower classification performance.

2.8 Discussion

Summarizing the issues from section 2.4 and the experimental results from section 2.6 we conclude the following:

1. The proposed method is well suited for low end, simple sensor nodes and smart objects and requires no additional positioning infrastructure.
2. The key source of information is sound sampling. Thus if size is critical the vibration motor can be dropped.
3. The system can reliably (90% and more accuracy) resolve a sufficient number of specific locations to cover one room or a small flat. It is advisable to combine our system with room level positioning.
4. The performance of the system is extremely inhomogeneous with respect to the classes, with most classes being recognized with high accuracy and a few "rogue" classes showing very poor performance.
5. Settling for the two or three best picks instead of a crisp single classification greatly increases the number of locations that are reliably recognized and the tolerance towards the "rogue" classes.
6. If training by the user is an issue, the abstract location class mode offers a possibility to provide "pre-trained" systems at the cost of more "fuzzy" location information.

Key points to investigate in the future are improved vibration sampling (using different amplitudes and frequencies to improve acceleration based performance), an investigation of the sources of errors on the problematic classes, more elaborate fusion methods, and a combination with radio signal strength based location methods.

Summarizing, this chapter treats detecting wether a device is carried on the body or placed in the environment as a special case of recognizing its symbolic placement. The active sampling method presented gives a specific solution to this recognition problem with merits and limitations discussed above. Moving away from the detecting environmental placements the focus is now on on-body sensing for the remainder of this thesis, centering on how to perform activity recognition independent of device placement and orientation, compensating for displacements. Thus after dealing with symbolic object placement containing locations on and off body, we focus on determine the on-body placement of a device.

2.9 Bibliography

[1] P. Bahl and V. N. Padmanabhan. RADAR: An in-building RF-based user location and tracking system. In *INFOCOM (2)*, pages 775–784, 2000.

[2] L. Doherty, K. S. J. Pister, and L. E. Ghaoui. Convex position estimation in wireless sensor networks. In *INFOCOM*. IEEE, 2001.

[3] J. Griffin and S. Fyke. *User hand detection for wireless devices U.S.Patent 20,060,172,706*. August 3, 2006.

[4] M. Hazas, C. Kray, H. Gellersen, H. Agbota, G. Kortuem, and A. Krohn. A relative positioning system for co-located mobile devices. In *MobiSys '05: Proceedings of the 3rd international conference on Mobile systems, applications, and services*, pages 177–190, New York, NY, USA, 2005. ACM Press.

[5] J. Hightower and G. Borriello. Location systems for ubiquitous computing. *Computer*, 34(8):57–66, 2001.

[6] L. E. Holmquist, F. Mattern, B. Schiele, P. Alahuhta, M. Beigl, and H.-W. Gellersen. Smart-its friends: A technique for users to easily establish connections between smart artefacts. In *UbiComp '01: Proceedings of the 3rd international conference on Ubiquitous Computing*, pages 116–122, London, UK, 2001. Springer-Verlag.

[7] M. V. Kleek, K. Kunze, K. Partridge, and J. B. Begole. Opf: A distributed context-sensing framework for ubiquitous computing environments. In *Proc.*

1st International Symposium on Ubiquitous Computing (UCS 2006), Seoul, Korea. Springer, October 2006.

[8] J. Krumm, G. Cermak, and E. Horvitz. Rightspot: A novel sense of location for a smart personal object. In *Ubicomp*, 2003.

[9] A. LaMarca, Y. Chawathe, S. Consolvo, and G. Borriello. Place lab: Device positioning using radio beacons in the wild. In *Pervasive 2005, Munich Germany*.

[10] J. Lester, B. Hannaford, and G. Borriello. "are you with me?" - using accelerometers to determine if two devi ces are carried by the same person. In A. Ferscha and F. Mattern, editors, *Pervasive Computing*, 2004.

[11] H. Olson. *Music, Physics and Engineering*. Courier Dover Publications, 1967.

[12] N. Priyantha, A. Chakraborty, and H. Balakrishnan. The Cricket location-support system. *Proceedings of the 6th annual international conference on Mobile computing and networking*, pages 32–43, 2000.

[13] D. Ruta and B. Gabrys. An overview of classifier fusion methods. *Computing and Information Systems*, 7(3-4):146–153, 2000.

[14] A. Schmidt, K. Aidoo, A. Takaluoma, U. Tuomela, K. Van Laerhoven, and W. Van de Velde. Advanced Interaction in Context. *LECTURE NOTES IN COMPUTER SCIENCE*, pages 89–101, 1999.

[15] J. Scott and B. Dragovic. Audio location: Accurate low-cost location sensing. In *Proceedings of the 3rd International Conference on Pervasive Computing (Pervasive 2005) Munich, Germany*. Springer-Verlag, May 2005.

[16] M. Sekine, T. Tamura, T. Fujimoto, and Y. Fukui. Classification of walking pattern using acceleration waveform in elderly people. *Engineering in Medicine and Biology Society*, 2:1356 – 1359, Jul 2000.

[17] M. Stager, P. Lukowicz, and G. Troster. Implementation and evaluation of a low-power sound-based user activity recognition system. In *ISWC '04: Proceedings of the Eighth International Symposium on Wearable Computers (ISWC'04)*, pages 138–141, Washington, DC, USA, 2004. IEEE Computer Society.

[18] R. Want, A. Hopper, V. Falcö, and J. Gibbons. The active badge location system. *ACM Trans. Inf. Syst.*, 10(1):91–102, 1992.

3

On-Body Placement

"All models are wrong, but some are useful." -George E. P. Box

Coarse variations related to the on-body placement of a device have a high impact on the reliability and effectiveness of context recognition systems. This chapter explores how changes in on-body placement impact sensing modalities commonly used in pervasive computing. We discuss general considerations and give some advice on how to make activity sensing more robust to placement changes. We then present several methods to derive the coarse device placement solely based on rotation and acceleration signals. The methods work regardless of device orientation. We present an elaborate evaluation of these methods on already published, large scale data sets with diverse activities from bicycle repair to house work. We reach a recognition rate of 80% over 4 min. of unconstrained motion data for the worst scenario and up to 90% over a 2 min. interval for the best scenario.

K. Kunze and P. Lukowicz. Using acceleration signatures from everyday activities for on-body device location. *11th IEEE International Symposium on Wearable Computers*, Sep 2007.
K. Kunze, P. Lukowicz, H. Junker, and G. Troester. Where am i: Recognizing on-body positions of wearable sensors. *LOCA'04: International Workshop on Location and Context Awareness* , Jan 2005.

After discussing general environmental placement issues, we focus on lifting another constraint for the users: Having to place sensing devices on well-defined positions on the body. Specifically, this chapter deals with coarse variations re-

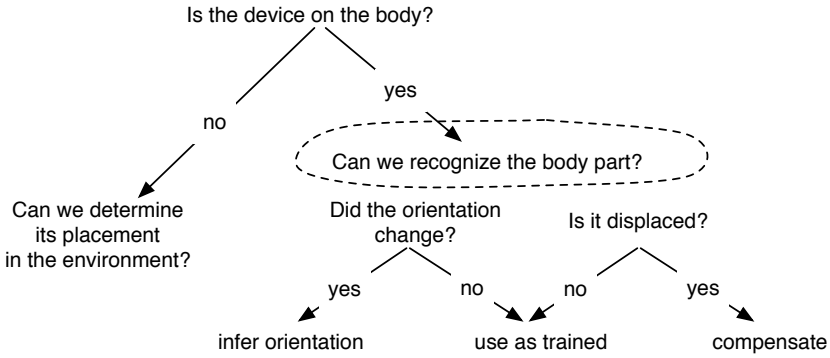


Figure 3.1: Thesis overview with the central question for this chapter highlighted.

lated to the body part, on which the device is carried (see in Figure 3.1).

A well-established approach to context and activity recognition is the use of motion sensors (predominantly accelerometers) attached to different parts of the user's body. Various types of activities ranging from simple modes of locomotion analysis to complex assembly tasks have been successfully recognized using such sensors [19, 4, 15] . Most research in this area, however, relies on sensors being placed at specific locations on the body. Typically, these include the wrists, the arms, legs, hips, the chest and even the head. Once a subset of placements is chosen, the system is trained on this specific subset and will not function properly if the sensors are placed at different locations. This implies that the user either has to explicitly "put on" the sensors each time he dresses up or the sensors have to be permanently integrated into the individual pieces of clothing or devices he usually carries, e.g. a mobile phone. If devices are used, the user is required to carry them always at the same body location, e.g. the key chain needs to be always placed in the right trouser pocket.

We consider this to be a very critical issue. Experience shows that people usually have several accessories with them and vary their on-body placement depending on the circumstance [8]. In a typical scenario the user might carry a key-chain in his trousers pocket giving us the leg information, a watch on the wrist, a mobile phone in a holster on the hip and a smart card in the wallet in a jacket pocket. A flexible context recognition system could then determine the on-body location of the devices to use them for an inference task.

Location information in itself is an interesting part of context. As an example, knowing if glasses are worn or if they are in a pocket can be an important clue to the user's activity.

The work described in this chapter is a major step in our effort to facilitate the adaption of context recognition with one focus: learning the device placement on

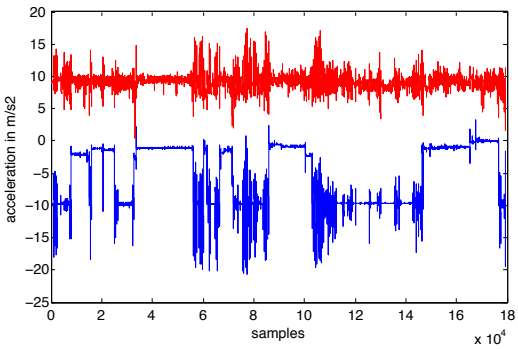


Figure 3.2: Body placement impact on an accelerometer signal; shown here is the horizontal axis of a sensor attached to the wrist (top), versus a sensor placed in the right trouser pocket (bottom). One can clearly see the sitting sections and the shifts in the gravity vector due to orientation changes of the sensor. The plot is from the House Work dataset.

the body. We illustrate on-body placement issues on sensor modalities common in activity sensing. Then, we explore how to classify different body placements. To assess the feasibility, we start with a very narrowly defined activity, namely "walking". We demonstrate how to detect "walking" in a device placement independent way and then leverage this to detect the device placement. In the following, we can abstract a more general model, no longer constraint to 'walking'. The model works for a broad range of common human activities tested in a large scale experimental evaluation.

3.1 Impacts of the Body Part Placement

Obviously, signals from motion related sensors such as accelerometers, gyroscopes and magnetic field sensors are significantly influenced by the body part, on which the device is placed.

Compared to sound or radio signals, motion signals are more closely linked to the on-body placement and not dependent on the absorption spectra of clothing or similar. The influence of the body placement on motion signals is twofold. First, some activities are associated with specific body parts. Sensors in other locations contain no or little related information. For example, activities related to subtle arm motions (e.g. screw driving, or washing hands) produce nearly no motion related signals in torso- or leg-mounted sensors, unless the motion is strong enough that the torso vibrates in sync with the hand motions. "Sitting down" and "standing up" also have a characteristic signature for an accelerometer on the upper leg (e.g. in the trouser pocket see Figure 3.2). Yet, they are nearly impossible to distinguish from a belt mounted accelerometer. Second, even for

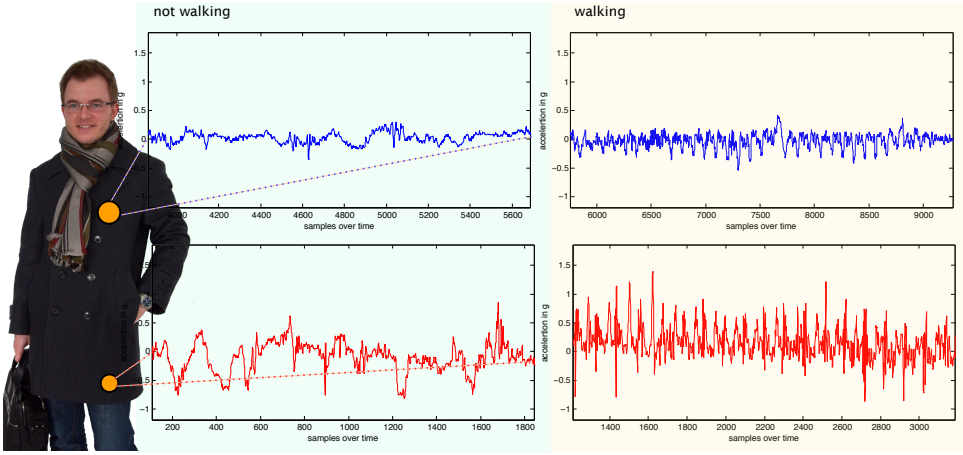


Figure 3.3: Accelerometer Signal, vertical axis for walking and not walking.

activities which are not strictly body part specific the motion sensor signals vary significantly between different body locations (see placement dependent signals in Figure 3.3 while the user is walking). The same holds for gyroscopes. Figure 3.6 shows gyroscope signals recorded form the lower arm and the head. In contrast to the gyroscope however, an accelerometer signal contains always static and dynamic acceleration. The static part is due to gravity, the dynamic part due to the user's motion. Both of them are not easily separable (see Figure 3.2).

Motion sensors and microphones, however, are not the only sensing modalities influenced by the on-body placement. As shown in Figure 3.4, WIFI signal strength, often used for indoor positioning, is also dependent on the on-body placement of the sensor. This is due to the large damping effect of the human body. A similar effect can be observed for GPS signals. This is illustrated in Figure 3.5. Interestingly, the GPS location fix is worst when the device is placed in the pocket, a very common on-body placement for smart phones.

As already discussed in Chapter 2, an obvious example of another sensing modality influenced by the on-body location is sound. Regarding sound, the signal impacts are related less to body damping and more to the absorption spectra of their surrounding, e.g. clothing. The absorption spectra for some often used types of clothing are shown in Figure 2.2. We already discussed how this fact can be used to recognize symbolic locations, including some body part placements, see Chapter 2 for details.

As shown above, common sensors used in context recognition are influenced by their placement on the body. We summarize our findings in the following:

- Radio communication from devices carried on the body is influenced by body dampening. It depends on the frequencies used and body part place-

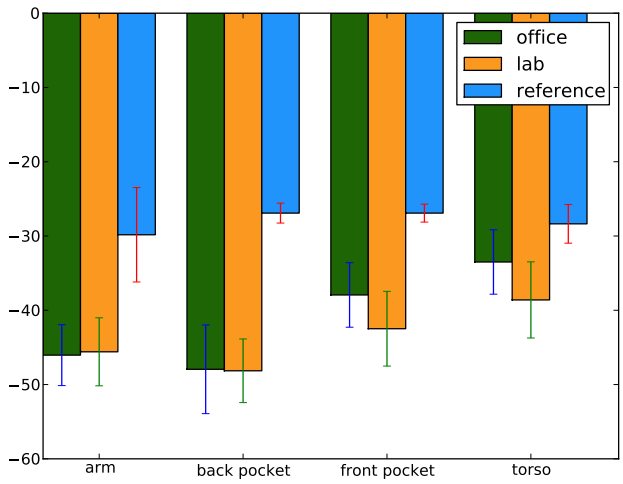


Figure 3.4: WIFI signal strength depending on on-body location of the mobile phone. The phone is put on a specific body location for 5 minutes stationary in several rooms: in one office, laboratory, and on the corridor between the two as reference. The experiment is repeated 15 times. We show the mean average signal strength in the plot.

ment. Our examples, WIFI and gps, show that the influences can be statistically significant.

- Sound might also be influenced by body dampening. However, the most dominant impact on sound is the absorption spectrum of the clothing and compartment in which the device is carried.
- The signals from motion sensors are highly specific to the on-body placement, even if they originate from movements of the user's whole body.

3.2 Related Work

Most research work focuses on aggregating sensor data to become device placement independent. Van Laerhoven et. al. present simple switch sensors and show that they are less body placement dependent with similar recognition rates for some activity recognition tasks compared to accelerometers [27]. Lester and Krause describe how to use sensor fusion methods to achieve device placement independent recognition [13, 10]. The activity recognition classes they can detect, however, are still rudimentary, e.g. modes of locomotion. Kern et. al. follow a similar approach using a multitude of different sensors [9]. Lester et. al. present

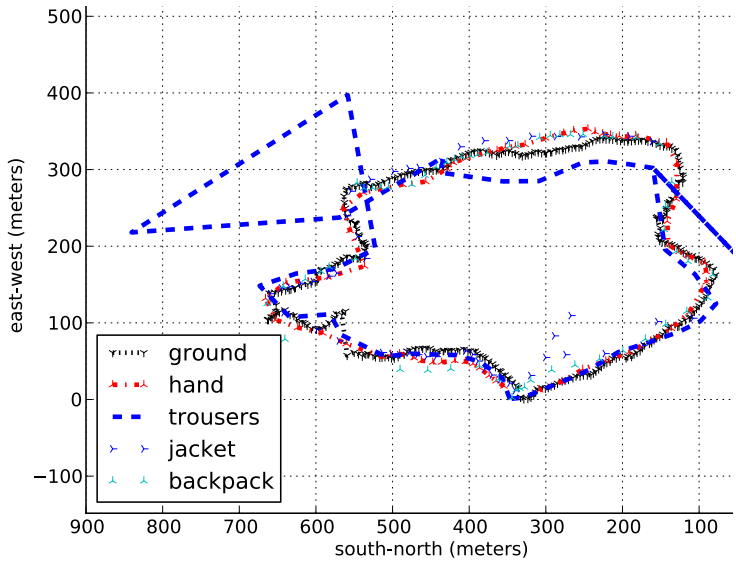


Figure 3.5: GPS traces using the same route recorded with the device at different on body placements. Three different mobile phones were placed in each of the following locations: hand, front pocket of trousers, inner pocket of the jacket, inside a backpack. The experimental study contains over 50 km of traces in different environments. The error for the mobile phone placed in the trouser pocket are by far the highest. The analysis shows that depending on the location on which the device is being carried, the average error can increase by as much as 50%.

how to detect if two devices are carried by the same person or different people [14] in a device placement independent way. Other interesting complimentary work comes from Blanke et. al. They fix the body placement (in the pocket) and infer the symbolic location of the wearer [5]. Laerhoven et. al. train recognition models adaptive to placement issues, yet they need direct user feedback for training [26].

The work closest to the one presented in this chapter is by Thiemjarus. She describes how to detect device orientation before applying activity recognition [23]. This is complementary to the work we present here. Work related to device orientation will be discussed in greater detail in Chapter 5.

3.3 General Considerations

There are two basic strategies to deal with different on-body placements for activity recognition. The first is quite simple: one can aggregate the sensor signals into features that are placement independent, for example using the norm vector from a three axis accelerometer. However, aggregation can only help little regarding

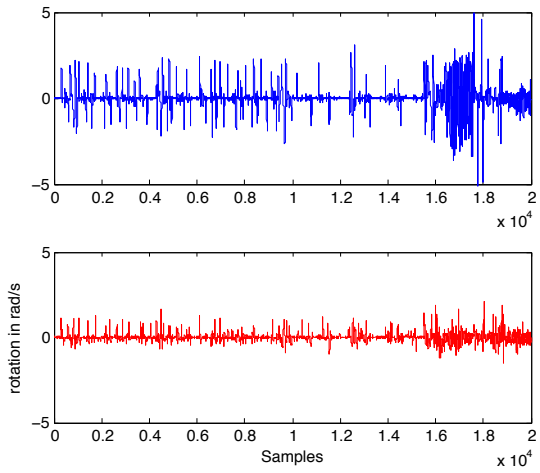


Figure 3.6: gyroscope signal example, horizontal axis for drinking gestures, on the top a sensor attached to the lower arm, on the bottom a sensor attached to the left side of the head. Although the movement is closely related, as drinking involves also tilting the head, the signals are clearly distinguishable.

such coarse variations as on-body placement, e.g. an aggregated accelerometer signal from the arm will still differ to a large degree from a signal recorded from the foot. The second strategy is to detect the actual device placement or present heuristics to deal with changes in placement.

Our approach is based on the obvious observation that different parts of the body tend to move in different ways. As an example, hand motions contain many more high frequency components and larger amplitudes than hip or head motions. To illustrate this, Figure 3.6 depicts the gyroscope signal for the drinking gesture for two distinct body parts, the lower arm in the top graph and the head in the bottom graph. Clearly, the lower arm shows higher angular velocities in average. Taking into account physiological constraints, certain types of motions are not permissible at all for some parts of the body e.g. you can not turn your leg around the vertical axis over the knee or tilt your head more than 90 degrees. Additionally, some parts tend to be motionless for longer periods of time than the others. Thus, in theory, a statistical analysis of the motion patterns over a sufficient period of time should be able to provide information about the location of a sensor on the body.

When implementing this idea in practice, however, one has to deal with a number of issues. For one, the value of such a statistical analysis depends on the user activity during the analysis window. Little information will be gained, for example, if the user is sleeping the whole time. Furthermore, the signal of a motion sensor placed on a given body part is a superposition of the motion

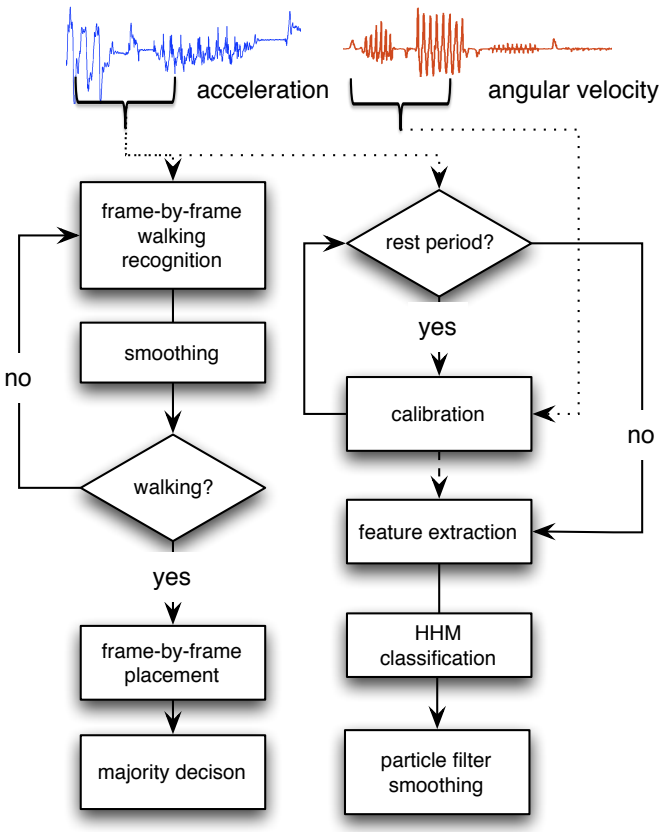


Figure 3.7: Method overview, on the left the walking recognition only approach, on the right the HMMs with particle filtering.

of this body part and the motion of the body as a whole. Thus, while it is not possible to tilt the head more than 90 degrees, such a tilt will be registered when the user lies down. Finally, many of the motion characteristics that can be used to distinguish between body parts involve absolute orientation, which is hard to detect, in particular if the orientation of the sensor is unknown.

3.4 On-Body Placement Recognition

Although there are significant motion differences between body parts, human movement patterns are often irregular and sporadic. Therefore, limiting the recognition first to specific moments could be helpful. We leverage the findings discussed in the considerations section and introduce in this section two methods to detect the on-body placement of a motion sensor shown in Figure 3.7. First,

we explore the "Walking Segment Method", the left part of Figure 3.7, based on accelerometer data alone. It constrains the body part placement recognition to the time the user is walking. Afterwards we deal with a basic time-series approach using Hidden Markov Models on both accelerometer and gyroscope data. The latter approach works on unconstrained movement data with the cost of increased complexity.

Walking Segments Method

A major issue to on-body detection are the the wide range of irregular, sporadic movements a user might do. The walking segments method tackles this problem in two ways:

1. The analysis is constrained to the time during which the user is walking. This is motivated by two considerations. First, walking is a common activity that occurs fairly often in most settings. Thus, being able to detect the position of devices during walking phases should provide us with a sufficiently accurate overall picture of where the devices are located. Moreover, once the location has been determined during a walking phase, this knowledge can be used to detect possible changes in placement. Walking has also a very distinct motion signature, that can be recognized in an on-body placement indifferent way [18, 19].

2. We base our analysis on the norm of the acceleration vector which is independent of the sensor orientation.

As you can see from Figure 3.3, walking provides us with a repetitive pattern, still maintaining distinct properties even for different body locations. Thus, a simple sliding window, frame-by-frame recognition approach with majority decision smoothing window should work for this problem.

Walking Recognition

Basic physical considerations confirmed by initial tests, using over 40 features and information gain as selection criteria, lead us to use the features given in Table 3.1 which we compute in one second sliding window (overlapping 0.5 sec.) over the acceleration signal from the device. The "walking" recognition is trained in a location independent manner by combining the data from multiple on-body locations into a single training set. Several standard machine learning algorithms are tested (e.g. C4.5, KNN). In the next phase, data collected during walking is used to train the placement recognition.

Table 3.1: Features used for the Walking Segments Method

RMS	$\sqrt{\frac{1}{N} * \sum_i x_i^2}$, where N is the number of samples a sliding window contains, and x_i the i'th sample of the window.
75%Percentile	Given a signal s(t) the 75%percentile, also known as the third quartile, is the value that is greater than 75% percent of the values of s(t) .
InterQuartileRange	The inter-quartile range is defined as the difference between the 75th percentile and the 25th percentile.
Frequency Range Power	Computes the power of the discrete FFT components for a given frequency band.
Frequency Entropy	The frequency entropy is calculated according to the following formula: $H_{freq} = -\sum p(X_i) * log_2(p(X_i))$, where X_i are the frequency components of the windowed time-domain signal for a given frequency band and $p(X_i)$ the probability of X. Thus, the frequency entropy is the normalized information entropy of the discrete FFT component magnitudes of the windowed time-domain- signal and is a measure of the distribution of the frequency components in the frequency band (see [3]).
Sums Power Wave Det. Coefficient	describes the power of the detail signals at given levels that are derived from the discrete wavelet transformation of the windowed time-domain signal. This feature has successfully been used to classify walking patterns with acceleration sensors([18]).

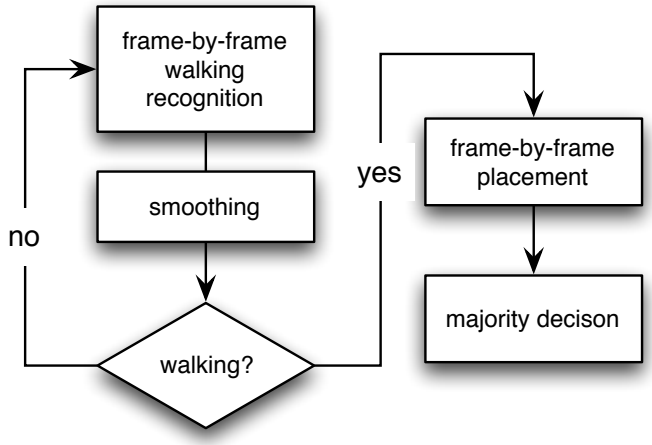


Figure 3.8: Overview of the walking segmentation method

Placement Recognition

The recognition of the sensor placement is performed separately for each sensor using the system trained according to the method described above. It consists of the following steps, as also depicted in Fig. 3.8:

1. *Frame by Frame Walking Recognition* In this phase the features are computed in a sliding window of length 1s as described above and each window is classified as walking or non walking. The window length has been selected such that in a typical case it contains at least one step.

2. *Walking Recognition Smoothing* Using another sliding window of length 10 sec moving by 5 sec the results of the frame by frame walking classification are then smoothed. The smoothing retains only those windows, where more then 70% of the frames are classified as walking. This ensures that the subsequent location classification is based only on 'clean' walking segments.

3. *Walking Segment Localization* The smoothed frame-by-frame recognition results are then used to localize walking segments that are long enough to allow reliable recognition. We define appropriate length to be at least 20-30 seconds and not longer than a 2 or 3 min. If a walking segment is longer than this boundary, it is automatically divided into several segments. The rationale behind this approach is that most devices are likely to remain in the same place for a few minutes. Changes on a smaller timescale must be considered as isolated events (e.g taking out a phone and rejecting an incoming call) and have to be detected separately by each device for each event.

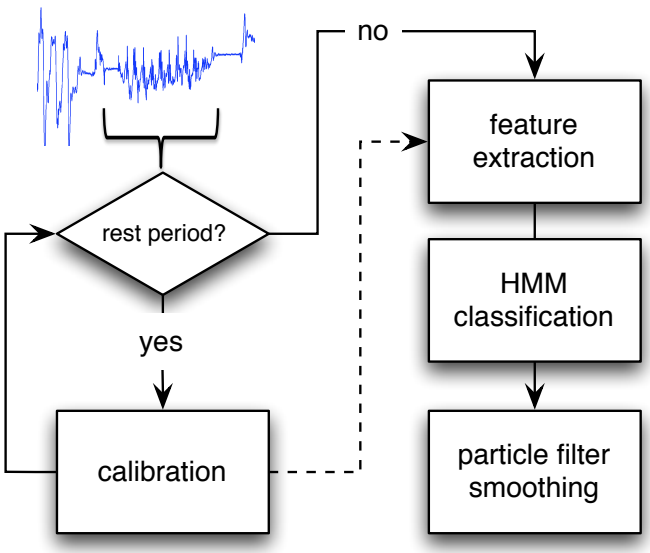


Figure 3.9: Overview of the body placement recognition using HMMs

4. *Frame By Frame Placement Recognition* A sliding window of the length of 1 sec., overlap 0.5 sec., is then applied to each segment that has been identified as a relevant walking event. For each window, the features for location recognition are computed and classification is performed.

5. *Event Based Location Recognition* For each segment a majority decision is performed on the frame by frame location classification.

Hidden Markov Models and Particle Filter

So far we just focused on placement detection while walking. To overcome this activity limitation, we take the problem to the time domain. For unconstrained motion data a naive sliding window approach won't work. To do an unconstrained location recognition, it's not enough to just look at a small snapshot in time. We need to shift our attention to a statistical analysis over longer time intervals. We pick Hidden Markov Models (HMMs) as recognition algorithm, as they enable to encode the distinct motion patterns over time. They are well explored in context recognition research. We train one HMM for each possible on-body location.

As the relative orientation of the accelerometer and gyroscope to the body part are generally not known, we perform a constant orientation calibration for one axis, described later in detail.

From a usage perspective, however, the HMMs have one major flaw. The HMM classification for a specific body part will perform badly if it receives uncharacteristic movement patterns from a specific body part. Assuming that a change in body location is not too likely, the HMM is limited by the time interval it takes into consideration.

A simple, naive procedure is to apply another majority decision window [11]. This only helps to smooth over the additional time interval. Therefore, we used a particle filter for smoothing. It is able to remove small disturbances and uncharacteristic patterns even over longer time periods.

In the following, we describe first the HMMs alone and second the smoothing using a particle filter. Before discussing those approaches, we will dive a little bit into the features used, as they are essential for a successful recognition.

Feature Calculation

Feature extraction is fairly straight forward, we use a one second sliding window (0.5 sec overlapping) for the calculation. We only perform feature extraction on segments with enough activity. If the variance of a segment on each axis tends towards zero and the magnitude towards 9.81 m/s2, we assume this is the gravity vector (see section 5.4 for references and more details). To account for sensor shifts and displacements within a body part, we perform a constant orientation calibration for one axis, as described by Mizell [16]. We perform feature extraction on this vertical axis and the norm vector of the two vectors orthogonal to gravity, if not indicated otherwise with the feature. As an additional feature we also use the length of the last calibration/rest period.

Our initial approach was to use a mixture of features that performed well in the frame-by-frame case, presented in Table 3.1. Yet, those proofed to be suboptimal. Table 3.2 lists the features we calculate on the accelerometer and gyroscope data.

Table 3.2: Features used for the Hidden Markov Models

Accelerometer	Gyroscope
standard deviation and mean fft center of mass duration of the last rest period	PCA angle (Blanke et.al. [5]) frequency range power (below and above 2 Hz)
The sum of the norm of the differences in variance for the normalized axes a_1, a_2, a_3 divided by the variance of the vector norm: $\frac{1/2 \sum_{i=1}^{n} \sum_{j=1, j<i}^{n} \mid var(a_i) - var(a_j) \mid}{var(norm)}$	

HMM Configuration

The features are calculated as described above and a sequence of 5 min. feature segments are fed into continuous HMMs. We use mixture of 3-5 Gaussian distributions to estimate the HMM output probabilities. We train a separate HMM for each body placement. Each HMM in itself is fully connected. Depending on the placement different numbers of hidden states are used: for the hand 5-6, torso 4, leg 5, and for the head 4 hidden states. Training of each HMM is done by expectation maximization using the Baum-Welch algorithm. For evaluation, we feed the test sequence in each placement specific HMM. There exists one HMM for each placement class. The HMM with the highest probability determines the class assignment during the classification. On top of the HMM only classification, we use a majority decision window of size 10.

HMMs with Particle Filter Smoothing

The basic method for the HMM part stays the same. As we apply the particle filtering we can reduce the sequence feed into the individual HMMs to 45 sec. We input the 45 sec. sliding window HMM classifications as observations into a particle filter. To our knowledge, particle filtering has not been applied to this type of activity sensing problems. Therefore, we will in the following go into more detail about the method used.

Particle Filtering In case the majority window is too crude to filter out uncharacteristic movements, we apply to this smoothing problem a partially observable dynamic model, a sequential monte carlo method, also called particle filter. The theoretical part of this section is a summary from Andrieu, Doucet, Thrun et. al. and Simon ([1, 7, 24, 21]). For a more detailed overview about filtering, especially more traditional apporaches (Kalman etc.) please refer to Simon ([21]).

Given we have the noisy classifications from the HMMs seen as state observations $y_{t_1}, \ldots, y_{t_k}$ at times $t_1, \ldots, t_k$, We want to estimate the hidden process states x_k. We assume that the observations y_k given x_k is, if it is conditionally independent, distributed according to the density function g, (see Equation 3.1). We want to estimate the true body location state x_k given the current and previous "observations" y_k (classifications of the HMMs).

$$y_k | x_k \sim g(y_k | x_k) \tag{3.1}$$

To estimate the distribution $p(x_k | y_{1:k})$ the particle filter samples a reference density $\pi(x_{t_k} | \{y_{t_i}\}_{i=1}^{k})$, sequentially with time i from $1, \ldots, k$. Particle filtering uses Bayesian estimation as the underlying principle to make predictions about the current/future state given the past observations. We use these predictions to smooth the results of HMM classifications. Particle filtering adheres to the

Markov assumption, every state depends only on the previous state (Equation 3.2). Additionally, the measurements depend only on the current state (Equation 3.3).

$$p(x_k|x_{1:k-1}) = p(x_k|x_{k-1}) \tag{3.2}$$

$$p(y_k|x_{1:k}) = p(y_k|x_k) \tag{3.3}$$

The relationship between measurements and system state is given in the Equations 3.4. u_k and v_k are random noise with known distributions and f and h are known, arbitrary functions.

$$\begin{aligned} x_k &= f(x_{k-1}) + u_k \\ y_k &= h(x_k) + v_k \end{aligned} \tag{3.4}$$

The prediction for the next state and update given a new measurement follows the Bayes' Rule (see Equations 3.5 and 3.6). The particle filter represents the posterior density, given in Formula 3.7, as a set of N random state vectors, called particles, denoted by $s_1 ... s_i$ and their associated weights $w_1 ... w_i$. We use a double index for the weights (i, t), i identifying the particle from $1 \dots N$ and t representing the time from $1 \dots k$. The posterior density is estimated over the weights. The representation given in Equation 3.8 approaches the posterior density for very large numbers N.

$$p(x_k|y_{1:k-1}) = \int f(x_k|x_{k-1}) p(x_{k-1}|y_{1:k-1})\, dx_{k-1} \tag{3.5}$$

$$p(x_k|y_{1:k}) = \frac{g(y_k|x_k) p(x_k|y_{1:k-1})}{p(y_k|y_{1:k-1})} \tag{3.6}$$

$$where : p(y_k|y_{1:k-1}) = \int g(y_k|x_k) p(x_k|y_{1:k-1}) f(x_k|x_{k-1}) dx_k \tag{3.7}$$

$$\int f(x_k) p(x_k|y_{1:k}) dx_k \approx \frac{1}{N} \sum_{i=1}^{N} w_i f(x_{k,i}) \tag{3.8}$$

To reach a good estimate the particle filter performs iterative importance resampling steps given subsequently.

1. Draw N particles from the proposed sampling distribution:
$s_t \sim \pi(x_t|s_{t-1}, y_t)$
for $t = 1$ to k **do**
   2. Compute and normalize the importance weight updates using the measurement y_t according to:
   $w_{i,t} = w_{i,t-1} \frac{p(y_t|s_t) p(s_t|s_{t-1})}{\pi(s_t|s_{t-1}, y_t)}$

3. Re-sample, discarding any particle s_i where the weight is smaller than a given threshold $w_{i,t} \leq w_{threshold}$.
4. Predict $\hat{x}$ from $p_{x_t|x_{t-1}}(x|s_{1:t-1})$.

end for

The setup for the placement detection is as follows. Each particle holds an estimate for the on-body placement class c denoted by $c_1...c_j$ (e.g., leg, wrist). The particles are initialized distributed equally with the placements of the data set to evaluate. The prediction model has a bias on not changing the placement classification; the probability for keeping the location class is set to steady 95 %.

To obtain a classification we use the sum up over the weights of all particles $cs_{1,t}...cs_{h,t}$ for a particular class. As seen in the evaluation section, reasonable results can be achieved with around 60 particles.

3.5 Evaluation

Subsequently, we will look at the performance of the presented approaches using some activity recognition data sets. We introduce first the different data sets and then go over the results.

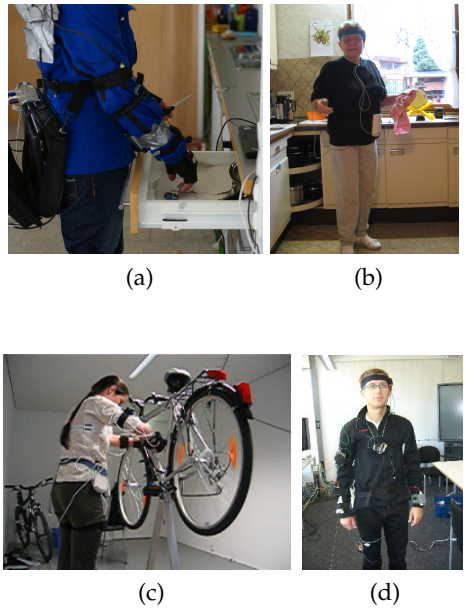
Data Sets

All of the experimental setups for data sets use the XSens XBus Master System[1] for recording motion data. The XSens sensors combine a accelerometer, gyroscope and magnetic field sensor. The opportunity data set contains in addition bluetooth accelerometers. As indicated above, we use the acceleration and rotation for our evaluation. Figure 3.10 shows pictures from the experimental setups of the 4 different data sets used.

Office work This data set contains 6 subjects. For each subject, 3 experimental runs were recorded. Each run lasts between 12 and 15 minutes and consists of the following set of activities: Working at a desk (writing emails, surfing, browsing through a book), making coffee, giving a presentation, walking between the activities (also including stairs). 4 Mtx Sensors are used. Sensor placements are the wrist, right side of the head, left trouser pocket and left torso pocket.

Opportunity This data set was recorded as part of the Opportunity EU Project. We use for our evaluation 7 users from the data set with 5 runs per person. A usual run takes 15 to 25 minutes. Thus, our evaluation set contains over 11 hours of data. Activities are from everyday living and include "making a sandwich", "pouring coffee", "eating" etc. For the evaluation, we use the xbus jacket data (gyro and acceleration) and the bluetooth accelerometer board sensor logs. The xbus jacket locations are lower arm, upper arm, and back with a sampling frequency of

[1] http://www.xsens.com



(a) (b)

(c) (d)

Figure 3.10: Photos from the data collection of the data sets used. The Opportunity recordings in 3.10a, house work 3.10b, bicycle repair 3.10c and the Drink and work data set 3.10d.

30Hz. The bluetooth accelerometer sensors are attached to the back of the hand, wrist, upper arm, knee and hip with a sampling frequency of 32Hz. The sensors were already used in previous research [2].

Drink and Work This data set contains mostly sitting activities, working on a computer and taking in food and drinks. In total 6 subjects are used in our evaluation; one experimental run is around 30-40 min. The Xsens motion sensors are attached at 5 locations, the upper back, right upper arm, right lower arm, head and the upper leg, again with a 30 Hz sampling frequency. Cheng et. al. describe more background information about the experimental setup ([6]).

Bicycle Repair This experimental setup contains repair activities on a bike (attaching a tire, opening screws etc.) with 6 test subjects. The average recording is around 25 min., 2 experimental runs for each subject. Again the xBus Master was used with sampling rate at 50 Hz. Only three placements are used: hand, lower and upper arm (for details see [22]).

House Work We conducted 3 experimental trials with 3 different test subjects, each trial lasting over 1 hour. We recorded real life activities in four different scenarios: Kitchen work, washing and ironing clothes, packing and office work. The data includes a wide range of activities from drying dishes over folding shirts to making coffee. For the experiments, we used the xBus Master system. These

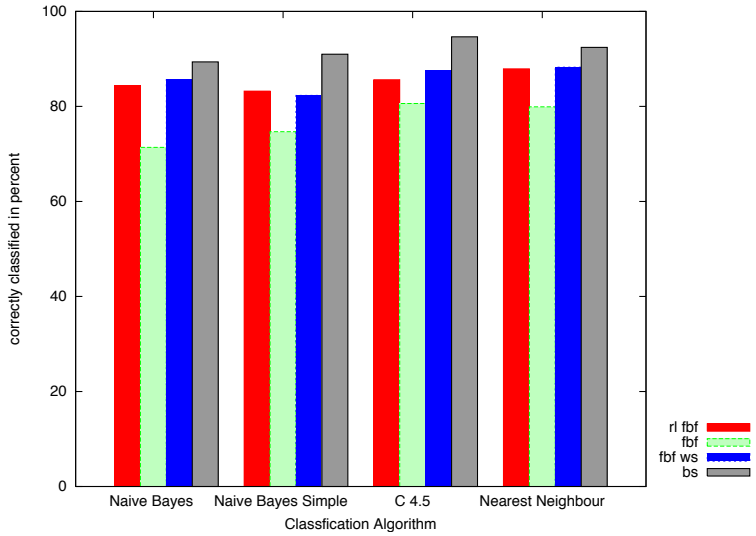


Figure 3.11: Overview over the different classification algorithms and the varying approaches. The abbreviations have the following meaning: rl fbf =frame by frame using reference labeling fbf = for frame by frame location recognition using frame by frame walking, fbf ws = frame by frame location recognition using smoothing over walking, bs = smoothing approach for both location and walking.

experiments are specifically recorded for the on-body placement detection. Thus the placements are picked according to a study by Ichikawa et. al. [8] and are as follows: right wrist, head, torso, front and back trousers pocket with 50 Hz sampling frequency [11].

Walking Segments Method Results

This method is applied to the Office Work and Opportunity data sets only, as it requires long patches of walking by the users. Both data sets contain such long enough walking patches.

Placement Recognition on Segmented Data As already mentioned earlier, the location recognition is only done during walking. Thus we begin our analysis by looking at the performance of the placement recognition on hand picked walking segments. The results of the frame-by-frame recognition an all 90 segments contained in the experimental data is shown in figure 3.11. Using a majority decision on each segment leads to a 100% correct recognition (124 out of 124). The smallest segment is 1 minute long.

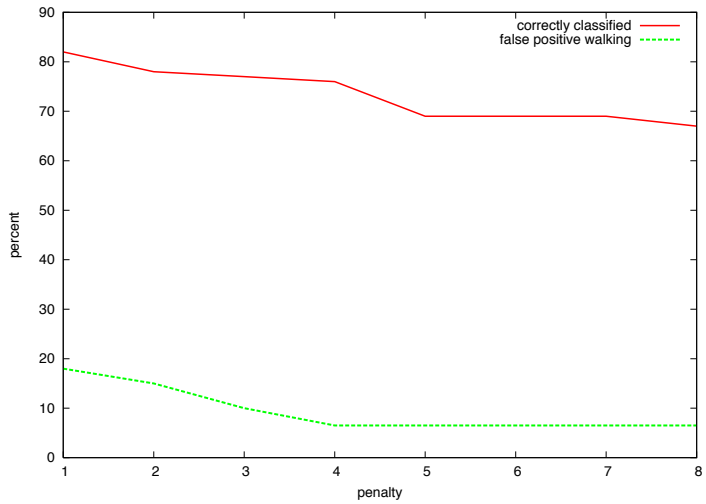


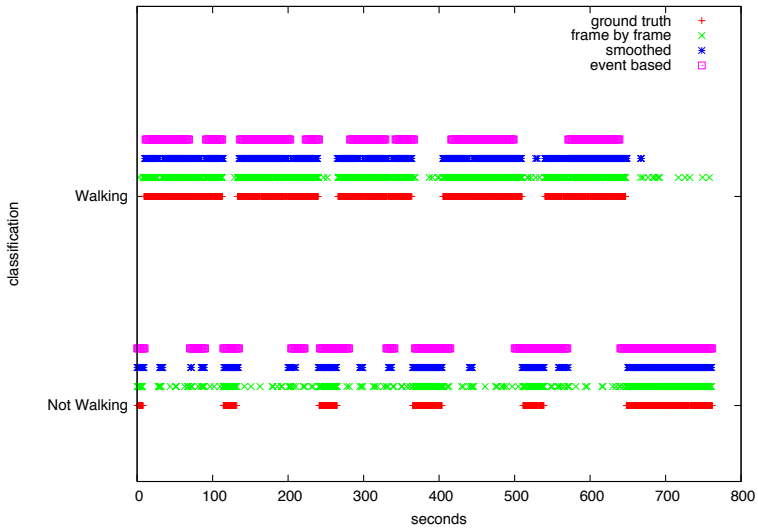
Figure 3.12: Relation between correctly classified and false positives for walking

Continuous Placement Recognition The first step towards placement recognition from a real life, continuous data stream is the detection of walking segments. As shown in Table 3.3 a frame-by-frame walking recognition (walking vs. not walking) showed an accuracy between 69% and 95% (mean 82%). However, for our purpose the mere accuracy is not the main concern. Instead we are interested in minimizing the number of false positives, as the subsequent location recognition works correctly only if applied to walking data. Here a mean of 18%(over all experiments) it is definitely too high.

As a consequence a false positive penalty has been added to the classification algorithms. Tests (see Figure 3.12) have lead to a minimal false positive rate considering a misclassification of 'Not Walking' four times worse than a misclassification of 'Walking'. While the overall correct rates goes down to between 61% and 85% (mean 76%), the percentage of false positives for 'Walking' is reduced to an average of 4% (between 0.5% and 7%).
The best results for the walking recognition are provided by the C4.5 tree algorithm with a mean of 82%, the worst by the Naive Bayes Simple with a mean of 65%.

In the next step, the effect of the jumping window smoothing we described in previous research ([11]) was investigated. It showed an average false positive rate of 2.17%, with 84% of the windows being correctly recognized.

Walking Segments Location In the last walking recognition step the walking segment location was applied to the smoothed frame by frame results. This has lead to 124 segments being located, none of which was located in a non-walking



h

Figure 3.13: Sample set containing different approaches for recognizing the walking segments

Table 3.3: Overview Classification for frame-by-frame walking recognition in percent, user-dependent, per test subject(P1 to P6).

	P1	P2	P3	P4	P5	P6	Mean
Plain							
Correctly Classified	95	69	87	78	82	85	82.67
False Positives for Walking	14	8	26	10	34	18	18.33
With penalty							
Correctly Classified	83	61	78	75	79	81	76.17
False Positives for Walking	3	5	0.5	8	6	6	4.75
With penalty + jumping window							
Correctly Classified	93	72	89	85	78	92	84.83
False Positives for Walking	2	3	1	2	2	3	2.17

section. As shown for an example data set in Figure 3.13 the only deviations from the ground truth was the splitting of single segments and the fact that the detected segments were in general shorter than the ground truth segments. In terms of suitability for location recognition, however, this is not relevant.

Frame By Frame Placement Recognition With the walking segments detected the frame-by-frame placement recognition is applied. The results are shown in Figure 3.11. They are later improved using the jumping window smoothing

method which leads to the results shown in Figure 3.11 and Table 3.6.

Table 3.4: Mean of C4.5 over all data sets for pre-labeled frame-by-frame (89,81 % correctly classified)

a	b	c	d	← classified as
856	2	87	5	a = head
21	804	0	12	b = trousers
101	32	765	4	c = torso
0	103	5	819	d = wrist

Table 3.5: Mean of C4.5 over all data sets for frame-by-frame using frame-by-frame walking recognition (80 % correctly classified)

a	b	c	d	← classified as
567	4	94	4	a = head
3	431	3	178	b = trousers
83	32	678	10	c = torso
12	155	24	754	d =wrist

Table 3.6: Mean of C4.5 over all data sets for both smoothed walking and location (94 % correctly classified)

a	b	c	d	← classified as
965	2	31	2	a = head
0	847	4	49	b = trousers
42	0	883	1	c= torso
17	68	10	921	d = wrist

The confusion matrices depicted in Tables 3.4 ,3.5, 3.6 indicate that the sensors attached to head and torso, as well as, trousers and wrist are most often confused. Especially, the confusion between Hand and trousers is significant in size. One possible reason is that the movement pattern of Hand and Leg is similar while walking, particularly if the test subjects swing with the arm.

Table 3.7: HMM Overview for several segment sizes

Set / time	30 sec.	45 sec.	1 min.	3 min.	4 min.	5 min.
Bicycle	43 %	67 %	-	83 %	83 %	84 %
House	32 %	65 %	68 %	73 %	82 %	79 %
Opp. (accel)	20 %	59 %	-	69 %	80 %	82 %
Drink and Work	15 %	61 %	-	-	72 %	78 %

Event Based Placement Recognition In a final step majority decision was performed in each segment leading to an event based recognition. Just like in the manually segmented case *the recognition rate was 100 %*.

We introduced a method that allows us to recognize where on the user's body an acceleration sensor is located. The experimental results presented above indicate that the method produces surprisingly reliable results. The method has found all walking segments in each experiment and has produced perfect event based recognition. Note that for practical use such event based recognition and not the less accurate frame by frame results are relevant.

Hidden Markov Models Results

Note that the walking segments methods has one severe limitation (despite just working on one particular activity). It needs long walking segments to work. To overcome this issue, we use Hidden Markov Models on unconstrained motion recordings. As the walking segments are not crucial anymore, we can conduct the following evaluation on all presented data sets.

Overall users – Table 3.7 gives an overview about the HMM based classifications on the different data sets for different segment duration. A segment duration contains a vector of features shown in Table 3.2 calculated over the 1 sec. sliding window. We use 33% of the respective data set as training and 66 % for evaluation.

We get the best performance on the bicycle data set. However, as there are only three on-body placements (lower arm, upper arm and hand), the test subjects were all from a similar age group and since the whole experimental setup was scripted, good results can be expected. The Opportunity accelerometer-only data performs surprisingly well, even though the additional gyroscope information is missing. The Opportunity data set is by far the largest, therefore the most representative data set. The Drink and Work dataset performs worst of the data sets taking 5 min. before we reach an accuracy close to 80%. One possible reason for this is that the test subjects are mostly stationary, sitting at a desk. Therefore, the movements can be quite uncharacteristic for the given body part. The 45 sec. HMMs have an

Table 3.8: Bicycle Repair with gyroscope and accelerometer 3 min. 83 %

a	b	c	← classified as
96.9	1.2	2.0	a = hand
10.1	69.7	20.2	b = lower arm
3.4	11.4	85.2	c = upper arm

Table 3.9: Bicycle Repair with accelerometer only, 3 min. segment size, 76 % accuracy

a	b	c	← classified as
85.1	12.9	2.0	a = hand
14.0	66.7	19.3	b = lower arm
5.4	13.0	81.5	c = upper arm

Table 3.10: House Work 4 min. 82 %

a	b	c	d	e	← classified as
93.5	4.9	0.0	1.6	0.0	a = head
0.0	100.0	0.0	0.0	0.0	b = wrist
0.7	0.0	83.5	15.8	0.0	c = torso
10.4	1.4	2.1	81.9	4.2	d = back pocket
1.4	0.3	6.1	24.2	68.0	e = front pocket

Table 3.11: Opportunity XBus Jacket 2 min 90 %.

a	b	c	← classified as
90.5	8.6	0.9	a = lower arm
10.0	85.3	4.7	b = upper arm
0.0	6.2	93.8	c = back

interesting threshold of around 60%, this is important as it proofs to be a decent start segment size for the particle filter later on.

The HMM results are summarized in the Tables 3.8 to 3.13 showing confusion matrices and overall accuracies.

Comparing the Bicycle with the Opportunity xBus dataset (Tables 3.11 and 3.8), it seems strange that the Bicycle scenario performs worse, as it also has only 3 classes. Even though the number of locations is equal, the locations themselves are not. The xBus jacket locations (lower arm, upper arm, back) are more diverse

Table 3.12: Opportunity accelerometer bluetooth nodes only 4 min 80 %.

a	b	c	d	e	← classified as
66.1	9.3	12.0	10.1	2.6	a = hand
17.4	76.2	0.2	0.5	5.8	b = wrist
8.4	5.2	81.6	3.3	1.5	c = upper arm
9.5	2.3	0.8	85.5	1.9	d = knee
1.1	10.2	0.1	0.3	88.3	e = hip

Table 3.13: Drink and Work 5 min. 78 %.

a	b	c	d	e	← classified as
73.1	9.6	15.2	0.7	1.4	a = back
15.2	70.2	7.0	5.8	1.8	b = head
7.2	9.6	76.0	3.2	4.0	c = upper leg
0.0	3.3	0.8	87.8	8.1	d = lower arm
3.9	1.6	0.8	8.5	85.3	e = upper arm

Table 3.14: HMM recognition rate (mean rate over several runs with different training sets) for user independent training, using 4 min segments.

Set / users for training	1	2	3	4	5
bicycle (6 users)	27 %	42 %	45%	50%	69 %
house (3 users)	15 %	18 %	-	-	-
opp. (accel, 7 users)	28 %	36 %	40 %	48%	72 %
drink and work (6 users)	23 %	23 %	32 %	35%	52 %

than the Bicycle repair ones (hand, lower and upper arm). As the movements from the back can be easier distinguished from the arm movements, the Opportunity dataset performs better. Overall, the results in the confusion matrices are to be expected. Body placements with similar movement patterns tend to be confused.

User-independent – True user independence is only achieved if we train on one (or multiple) users and evaluate on the rest, excluding the users taken for training. Unfortunately, the HMM results in this case are not holding up to the good recognition rates seen before. The results are summarized in Table 3.14. The house work data set performs worst (less than mere chance). It seems the user dependent characteristics dominate the classification, therefore the recognition model is not generalizable over several users. The data set contains also the most

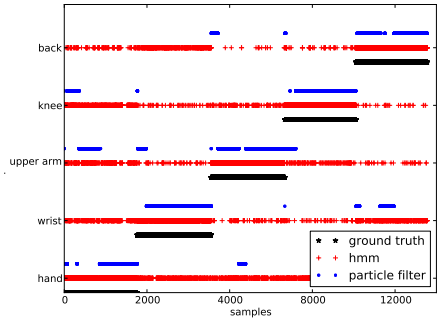


Figure 3.14: Scatter plot depicting the performance of the HMMs with and without particle filter smoothing. Here 20 min segments are taken from the Opportunity Accelerometer only data, with 45 sec. HMM classification (around 59 % correct). Achieving a 78 % particle filter with 40 particles.

variability in user demographic. For the opportunity and bicycle sets we see a clear increase for the classification rates and the 5 user case is getting closer to the recognition performance over all users. This indicates that the approach introduced might work also for the user-independent case. Yet, larger data sets are needed to evaluate true user independence. It is out of scope for this thesis, yet a promising research opportunity.

HMMs and Particle Filter Results

To show the usefulness of the particle filter on top of a HMM classification with a smaller segment size (45 sec.), we perform 3 evaluations. The scatter plot in Figure 3.14 clearly shows the smoothing effect of the particle filter prediction.

First of all, we determine the optimal amount of particles to use for the filter. We train our 45 sec. HMMs as described in the Methods section. We generate uniformly random 100 segments with 10 min. continuous accelerometer data from the test set (there can be overlapping 10 min. pieces). On these 10 min pieces we perform an evaluation for each time step and plot the average accuracy and standard deviation for each data set, as seen in Figure 3.15. It is easier to compare the performance of the particle filter regarding the different data sets in the summary Figure 3.16. We can infer that a particle number of 60 seems a good pick for the data sets, as there is no significant accuracy increase with 70 or 80 particles.

Another plot showing the performance of different particle sizes is given in Figure 3.17 for the Opportunity data set. The 60 particle case is more stable and smooth than the predictions with lower particle numbers. It is important to note that the 1st sec. in the plot is the 45th sec. of the data, as this time is needed for

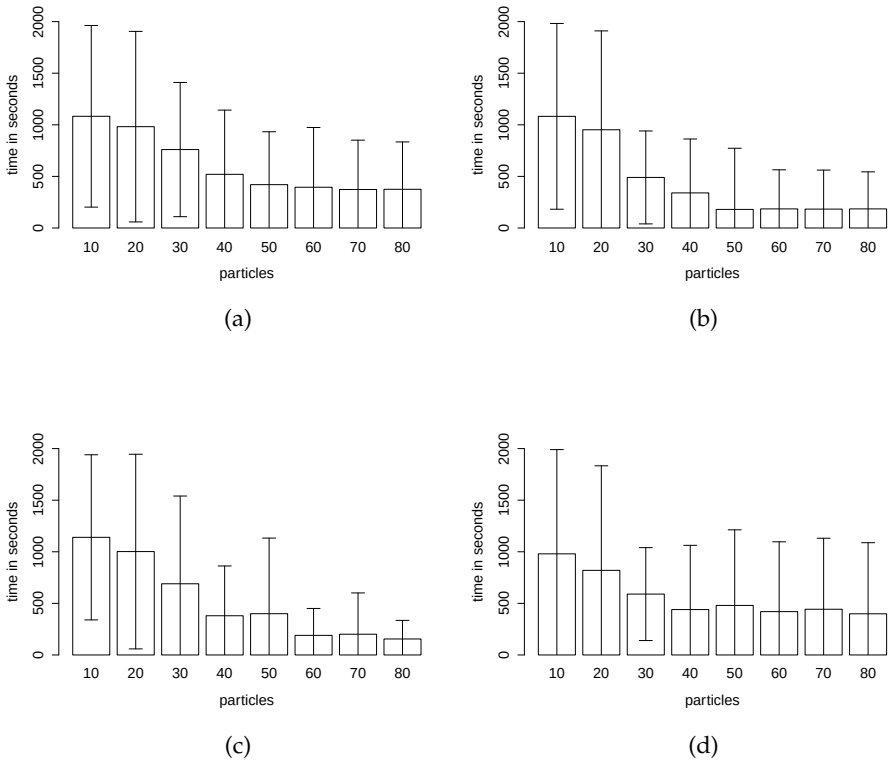


Figure 3.15: Mean and standard deviation of the switch time using 100 segments between 10 -20 min uniformly randomized for the opportunity (3.15a) the bicycle repair(3.15b), the drink and work (3.15c) and the house work data set(3.15d).

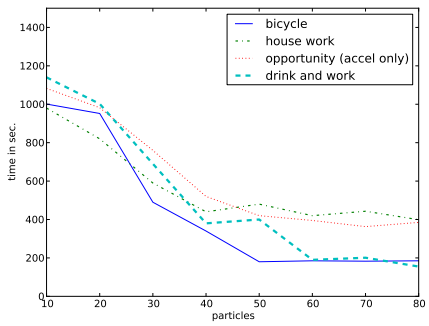


Figure 3.16: Mean and standard deviation of the switch time using 100 segments between 10 -20 min uniformly randomized for all data sets.

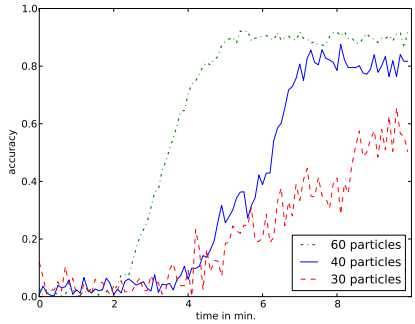


Figure 3.17: Classification improvement using different particle numbers for 100 x 10 min segments.

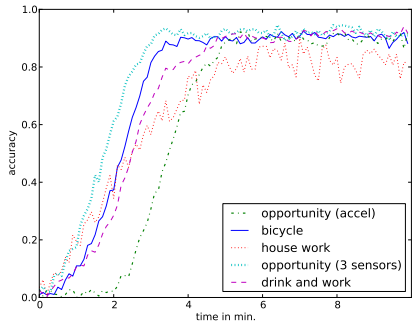


Figure 3.18: 100 x 10 min segments, 60 particles different data sets.

the first HMM observation to appear.

Second, we want to get the average timing for when the particle filter supplies us with a stable reading and again a more general evaluation on the particle size to use. We use 100 time slices, uniformly random between 10 and 20 minutes in size for this evaluation. House and Opportunity accelerometer only data set require a long switch time, around 6 min with 60 particles. Switch time is the time it takes before the majority of the particles show the correct state. Whereas the Bicycle repair and Drink and Work experiments are quicker with 3 and 4 min. respectively. The drink and work data set also shows significant improvements on higher particle numbers (towards 70 and 80 particles). This holds true a bit for the opportunity accelerometer only data set, however in this case only the standard deviation goes down.

Third, to test our filter design, we uniformly randomly pick 100 x 10 min from the data sets (there can be duplicate parts), do feature extraction and HMM classification and feed them into the particle filter. Afterwards we evaluate how many

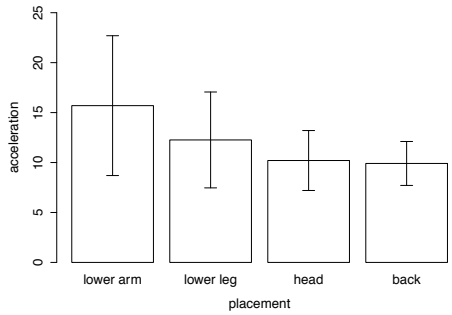


Figure 3.19: Mean and standard deviation of the acceleration (in m/s^2. for different body locations over 2 users of the Drink and Work data set.

percent of the 100 filtered placements are detected correctly. Figure 3.18 shows the results for the different data sets. As can be seen, the Office work scenario performs best (using gyroscope and accelerometer), the Opportunity data set is second (gyroscope and accelerometer). The House Work starts off quick (accel + gyro), yet does not achieve the average 90% accuracy and has the most variance from one filtered classification to the other.

We were a bit startled by the bad performance of the particle filtering in the first 1-2 min. This is likely due to the particle initialization; conceivably, it is possible to improve the start up time by incorporating the first few HMM classifications with a stronger bias or incorporate other knowledge. A logical continuation of this work is to design and conduct experiments with realistic switch situations between device on-body placements, to get better priors and maybe to model the switch itself.

3.6 Discussion

We present several methods allowing us to derive the coarse device placement solely based on rotation and acceleration signals from the device. The methods work regardless of device orientation. We show an elaborate evaluation of these methods on already published, large scale data sets with diverse activities from bicycle repair to house work. The recognition rate reaches up to 80% over 4 min. of unconstrained motion data for the worst scenario and up to 90% over a 2 min. interval for the best scenario. It seems the gyroscope features are very well suited for distinguishing wrist and hand placements from the rest, as one can also see from the confusion matrices in Tables 3.8 and 3.9. Other than that, most missclassifications can be easily explained, as they are from closely related placements (back pocket and front pocket, hand and wrist etc.). The Drink and

Work data set is worst in classification, this is due to the fact that the person was stationary and sitting most of the time. The body parts with less movement perform not that well (head and torso), as can be seen from the confusion matrix in Table 3.13 and Figure 3.19. Interesting here is that although it is worse in the HMM case, the classification problems can be smoothed out with the particle filter.

The data set with the worst particle filter results, by far, is the House work. Due to its variability of activities and the unscripted nature of the recording, the accuracy highly depends on the used training and test set. Also the diversity in test subject is highest with this experimental setup. The Opportunity Bluetooth accelerometer data performs extremely well using the filter smoothing, despite the fact that the data quality is considerably lower than that for the Mtx Sensors and the additional gyroscope is missing. The impact isn't all that high, requiring roughly two minutes more run-time for a stable particle filter performance compared to the Mtx sensor based inferences. This indicates that larger, more representative data sets are essential.

There are several promising algorithmic alternatives and potential future work regarding the on-body placement problem:

Conditional random fields – Recent activity recognition research applied an established inference framework from machine learning, conditional random fields (CRFs) ([12, 25]). They are more general than HMMs, in fact HMMs can also be expressed by CRFs. The major conceptional advantage of CRFs over HMMs is that they get rid of the independence assumption for the observations. As the observations in the on-body placement method are all derived from acceleration and rotational velocity, this assumption is definitely violated, as with most HMM inferences in activity recognition research ([28, 17]). The CRF also don't need to rely on the Markov property. CRFs do not try to model the underlying process, conceptionally they just predict the state according to the observations. There long observation interactions are feasible. Of course, one can imagine to also extend HMMs to overcome the Markov assumption, yet this is none-trivial and CRFs provide already a usable mathematical framework for it. Both, HMMs and CRFs are similar in terms of complexity regarding the training and evaluation. Related work shows an improvement of the classification rate between 5-10 % using CRFs versus HMMs for similar activity recognition problems ([25]). Yet, if CRFs perform better than the HMM inference in the on-body placement case remains to be seen.

Conformal Prediction – Conformal Prediction seems a good addition to the particle filter smoothing ([20]). Its design targets online classification problems. Conformal Prediction gives solid confidence estimates for the next predic-

tion of a system, even if the underlying model of the system is not correct. In the on-body placement classification it can be used to predict the confidence interval for the HMM classifications, the confidence in the prediction can then in turn be used for the weights adjustment in the particle filter. The method is quite novel in machine learning and not an established technique in the pervasive field. Thus, it is also not tested on large scale activity sensing data sets. Of course, Conformal Prediction can also be used to estimate the confidence of the particle filter classifications.

Filtering adjustments – We use the particle filter to show the best performance for the most general case, meaning we don't have any information how often the user switches the device placement and where the user wears the device regularly. Given a particular application domain and device type, we can make certain assumptions (e.g. a mobile phone is often carried in the front pocket [8]). Given these assumptions, depending on the application scenario, the particle filter options can be adjusted accordingly (e.g. the prior distributions for a given location). This will lead to shorter switch times.

One other issue that future work can address is the detection of location changes. The work described in this chapter constitutes a first step towards the use of sensors integrated in standard appliances and accessories carried by the user for complex context recognition. It is also motivated by the relevance of device location to infer the user's context.

3.7 Bibliography

[1] C. Andrieu, N. de Freitas, and A. Doucet. An introduction to MCMC for machine learning. *Machine learning*, 2003.

[2] M. Bächlin, J. Hausdorff, and D. Roggen. Online detection of freezing of gait in parkinson's disease patients: a performance characterization. *Proceedings of the . . .*, Jan 2009.

[3] L. Bao. Physical activity recognition from acceleration data under semi-naturalistic conditions. Master's thesis, MIT, 2003.

[4] L. Bao and S. Intille. Activity recognition from user-annotated acceleration data. In F. Mattern, editor, *Pervasive Computing*, 2004.

[5] U. Blanke and B. Schiele. Sensing location in the pocket. *Ubicomp Poser Session*, page 2, Aug 2008.

[6] J. Cheng, O. Amft, and P. Lukowicz. Active capacitive sensing: Exploring a new wearable sensing modality for activity recognition. *Pervasive Computing*, Jan 2010.

[7] A. Doucet and N. de Freitas. The unscented particle filter. *Advances in Neural Networks*, 2001.

[8] F. Ichikawa, J. Chipchase, and R. Grignani. Where's the phone? a study of mobile phone location in public spaces. *Mobile Technology, Applications and Systems, 2005 2nd International Conference on*, pages 1 – 8, Oct 2005.

[9] N. Kern, B. Schiele, and A. Schmidt. Multi-sensor activity context detection for wearable computing. 2003. European Symposium on Ambient Intelligence.

[10] A. Krause, D. Siewiorek, A. Smailagic, and J. Farringdon. Unsupervised, dynamic identification of physiological and activity context in wearable computing. *Wearable Computers*, Jan 2003.

[11] K. Kunze and P. Lukowicz. Using acceleration signatures from everyday activities for on-body device location. *Wearable Computers, 2007 11th IEEE International Symposium on*, pages 115 – 116, Sep 2007.

[12] J. D. Lafferty, A. McCallum, and F. C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.

[13] J. Lester, T. Choudhury, and G. Borriello. A practical approach to recognizing physical activities. *Proceedings of Pervasive*, Jan 2006.

[14] J. Lester, B. Hannaford, and G. Boriello. Are you with me?–using accelerometers to determine if two devices are carried by the same person. *Pervasive Computing: Second International Conference*, Jan 2004.

[15] P. Lukowicz, J. Ward, H. Junker, M. Staeger, G. Troester, A. Atrash, and S. Starner. Recognizing workshop activity using body worn microphones and accelerometers. In *Pervasive Computing*, 2004.

[16] D. Mizell. Using gravity to estimate accelerometer orientation. *Wearable Computers, 2003. Proceedings. Seventh IEEE International Symposium on*, pages 252 – 253, Sep 2005.

[17] D. J. Patterson, D. Fox, H. Kautz, and M. Philipose. Fine-grained activity recognition by aggregating abstract object usage. *Wearable Computers, IEEE International Symposium*, 0:44–51, 2005.

[18] M. Sekine, T. Tamura, T. Fujimoto, and Y. Fukui. Classification of walking pattern using acceleration waveform in elderly people. *Engineering in Medicine and Biology Society*, 2:1356 – 1359, Jul 2000.

[19] L. Seon-Woo and K. Mase. Recognition of walking behaviors for pedestrian navigation. In *Proceedings of the 2001 IEEE International Conference on Control Applications (CCA'01) (Cat*, pages 1152–1155, 2001.

[20] G. Shafer and V. Vovk. A tutorial on conformal prediction. *J. Mach. Learn. Res.*, 9:371–421, June 2008.

[21] D. Simon. Optimal state estimation: Kalman, h [infinity] and nonlinear approachesâ€Ž. *John Wiley & Sons.*, page 526, Jan 2006.

[22] T. Stiefmeier, G. Ogris, and H. Junker. Combining motion sensors and ultrasonic hands tracking for continuous activity recognition in a maintenance scenario. *2006 10th IEEE . . .*, Jan 2006.

[23] Thiemjarus. A device-orientation independent method for activity recognition. *Body Sensor Networks (BSN), 2010 International Conference on*, pages 19 – 23, 2010.

[24] S. Thrun, D. Fox, and W. Burgard. Robust Monte Carlo localization for mobile robots. *Artificial Intelligence*, 2001.

[25] D. L. Vail, M. M. Veloso, and J. D. Lafferty. Conditional random fields for activity recognition. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems*, AAMAS '07, pages 235:1–235:8, New York, NY, USA, 2007. ACM.

[26] K. Van-Laerhoven and O. Cakmakci. What shall we teach our pants? In *Digest of Papers. Fourth International Symposium on Wearable Computers.*, pages 77–83, 2000.

[27] K. van Laerhoven and H. Gellersen. Spine versus porcupine: a study in distributed wearable activity recognition. *Wearable Computers*, 2004.

[28] A. Wilson and A. Bobick. Parametric hidden markov models for gesture recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 21(9):884 –900, sep 1999.

4

Displacement

"The eye altering, alters all."
-William Blake, The Mental Traveller

Dislocated sensors, changing their position during use, are very problematic for realistic activity recognition. This chapter motivates why dealing with sensor displacement within a body part is crucial. We discuss the theoretical background, point out alternative approaches, and present a set of heuristics that significantly increase the robustness of motion sensor-based activity recognition with respect to sensor displacement. We show how, within certain limits and with modest quality degradation, motion sensor-based activity recognition can be implemented in a displacement tolerant way. We describe the physical principles that lead to our heuristic and evaluate them on a set of synthetic lower arm motions. These motions are well suited to illustrate the strengths and limits of our approach. We extend the evaluation on a realistic modes of locomotion problem (sensors on the upper leg) and finally on a set of exercises performed on various gym machines (sensors placed on the lower arm). Our heuristic raises the displaced recognition rate from 24% for a displaced accelerometer (96% accuracy when not displaced) to 82%.

Kunze, K. and Lukowicz, P. Dealing with sensor displacement in motion-based on-body activity recognition systems. *In Proceedings of the 10th Conference on Ubiquitous computing.* Seoul, Korea, September, 2008. (Acceptance rate: 18%).

Even if we know the coarse grain on-body placement of a device, it's often not possible to perform activity recognition. On-body motion sensors can just provide crude contextual information in realistic scenarios without considering orientation changes and displacement [9, 12].

We define displacement as dislocation within a body part, excluding orientation issues. Take the example of an mp3 player attached to the upper arm. If it shifts in position up or downwards, the methods and heuristics in this chapter apply. If however the device rotates around the arm and shifts, the heuristics in this chapter need to be accompanied by some methods to deal with orientation change described later.

Today, most device attachment methods leave a lot of room for placement within a body part. Thus, a user can place arm holders for mp3 players, often used for jogging, almost anywhere on the upper or lower arm. Integration of sensors in clothing can ensure that sensors end up on a certain body part. However, one cannot ensure a specific placement on that body part. Even a tight fitting sleeve can be rolled up or twisted, completely changing the placement of any integrated sensor.

Unfortunately, the within body part placement issue cannot be solved with simple calibration gestures for motion sensors. As explained in section 4.4, the gravity component of an accelerometer does not depend on the position within a body part. Thus, a static calibration gesture is not sufficient. Instead, motions must be performed with *different, exactly defined speeds and trajectories*. In general, we cannot expect the user to perform such exactly defined motions with sufficient reliability.

4.1 Sensor Displacement Impacts

Displacement within a body part is crucial for motion-sensor-based activity recognition. Yet, displacement effects other sensing modalities to a much lesser degree. For those, on-body placement and orientation changes are more critical (see Sections 3.2 and 5.2). Therefore, non-motion-based modalities are not discussed in this chapter.

Motion sensors, in particular accelerometers, are a common type of body worn sensors for activity recognition. Following the original work by Randell, Van Laerhofen and Mantyjarvi [13, 17, 11], there have been numerous publications dealing with applications ranging from dancing sign language recognition, tracking of every day activities to industrial maintenance and mental health related applications [1, 3, 16, 8, 15, 19].

The vast majority of research in this area assumes well defined, fixed sensor locations. This is particularly important for activity recognition related to arm and hand motions. As depicted in Figure 4.1, displacement within a body part highly affects the characteristics of an accelerometer signal.

Being able to drop the requirement for a 'well defined fixed position' and building systems that can deal with sensor displacement has two major advantages:

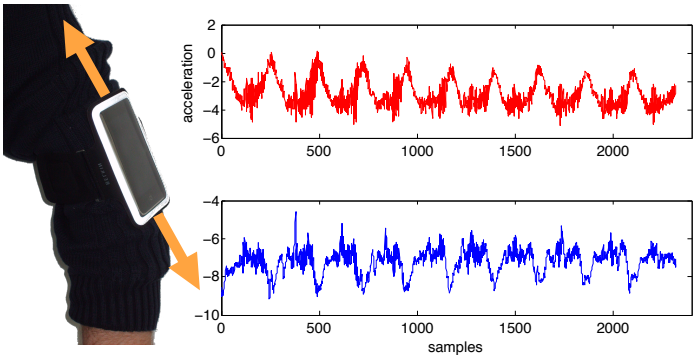


Figure 4.1: Signal example for displacement within a body part. You see one axis of two acceleration signals from the same device mounted on the upper arm, the only difference being that the device is displaced by 10 cm.

- Robustness. During long-term deployment, sensor shifts will eventually happen. Enabling the system to continue working correctly despite sensor shifts is a significant improvement to robustness.
- Better usability and user acceptance. Today, many mobile appliance are already equipped with sensors. Sensor encapsulation into clothing or unobtrusive attachment e.g., as "buttons" has been demonstrated [14]. It is thus often taken for granted that users can be easily equipped with sensors in every day situations. This does not imply, however, that the user can be expected to reliably and firmly fix the sensors to narrowly defined on-body locations.

In summary, regarding the three sub-categories described in the introduction ('on-body part' placement, displacement and sensor orientation changes), displacement is the most difficult to handle. It is so far an unsolved problem. This chapter addresses possible solutions.

4.2 Related Work

To our knowledge there is no other work directly targeting the problem of 'within body part' displacement for motion sensors. The standard practice to deal with displacement is to take robust, aggregate features, which can either only be used for very simple recognition tasks or lead to a significant degradation in recognition performance [9, 7, 2]. Yet, there is some complimentary work, trying to tackle the problem more directly.

Van Laerhoven presented a study to explore the trade-offs between single on-body sensors with predefined, well-known placement and an increasing quantity

of sensors with decreasing information quality (placement accuracy)[18]. Zinnen showed an innovative way to use rest periods in accelerometer signals for detection of non-repetitive tasks, which is based on some of the principles presented below in section 4.4 [20]. Lester uses acceleration signatures to determine that a set of devices is being carried by the same person [10]. There is also some work on evaluating the suitability of different on body locations for activity recognition [12]. A platform with multiple sensors (in addition to mere motion sensors) has been investigated with respect to on-body location invariance of activity recognition [9]. Forster et. al. utilize dedicated hardware with multiple accelerometers to track displacement changes online [4, 5].

4.3 Idea and Contributions

Although we did not discover an exact solution, we present a set of heuristics that significantly increase the robustness of motion sensor-based activity recognition with respect to sensor displacement within a single body part. We show how, within certain limits and with modest quality degradation, our heuristics allow motion sensor based activity recognition to be implemented in a displacement tolerant way. Thus, within a single body part, we demonstrate reliable recognition at locations different from those on which the sensor was trained. Our approach is based on three observations:

1. The signal of a body-worn accelerometer is the sum of three components: acceleration due to rotation, acceleration due to translation and acceleration due to orientation with respect to gravity. Of those three only the first one, acceleration due to rotation, is sensitive to sensor displacement within a single body part. We explain the underlying physical considerations in detail in Section 4.4.

2. The accelerometer signal segments dominated by rotation can be identified. These segments are possibly 'contaminated' with displacement related noise.

3. Gyroscopes are more insensitive to displacement within a single body part. However, they provide only information on rotation, ignoring translations and vertical orientation.

From the above observations, it follows that combining a gyroscope with an accelerometer removes some placement sensitivity, as the accelerometer can ignore all signal frames dominated by rotation , while retaining most of the relevant information. In fact, sometimes just an accelerometer ignoring the 'rotation-contaminated' frames is enough for more displacement tolerant recognition. Additional measures we propose are the use of heavily low pass filtered acceleration

signals and training the system on two sensors corresponding to the 'worst possible displacement'.

The main validity limits of our heuristics are (1) a rigid body approximation of human body parts and (2) the assumption that the bulk of the discriminative information is not in signal segments that contain simultaneously performed fast rotations, significant translations or changes in vertical orientation.

In the rest of the chapter, we describe how our heuristics can be derived from basic physical considerations. Next we apply these heuristics to a set of "synthetic" gestures that are well suited to demonstrate the strengths and limits of our approach. Finally, we present an evaluation on two real life recognition tasks. The first task is an extended modes of locomotion problem using upper leg mounted sensors. The second is a set of gym exercises classified using sensors mounted on the lower arm. On this set our heuristics improve recognition rates for displaced sensors from 24% using a displaced accelerometer(96% recognition when not displaced) to 82%. On other examples the displaced recognition rates raise from around 60% to over 90%.

4.4 Physical Considerations

To better understand our approach, we discuss the physics background, introducing a rigid body approximation and deducing consequences from this assumption regarding motion sensors.

The Rigid Body Approximation

A common approximation used in modeling human body motion is that of rigid segments connected by joints which allow rotation around one (e.g. elbow) or more axes (e.g. wrist). In simple terms, such an approximation is equivalent to a 'stick figure' representation used in many animations. In exact terms, a rigid body is an ideal solid body of finite size for which the relative position of any two given points remains constant in time regardless of external forces exerted on it. Any motion of a rigid body can be described as a combination of a translation and a rotation.

Although human joints posess only rotational degrees of freedom, a motion combining simultaneous rotation at two different joints can have the effect of a translation (e.g. shifting your lower arm through a combined elbow and shoulder motion). Additionally, motions involving more than one joint can lead to rotations around axes that are not identical to any of the involved joints. Regarding arm motions, such axes are often close to the torso.

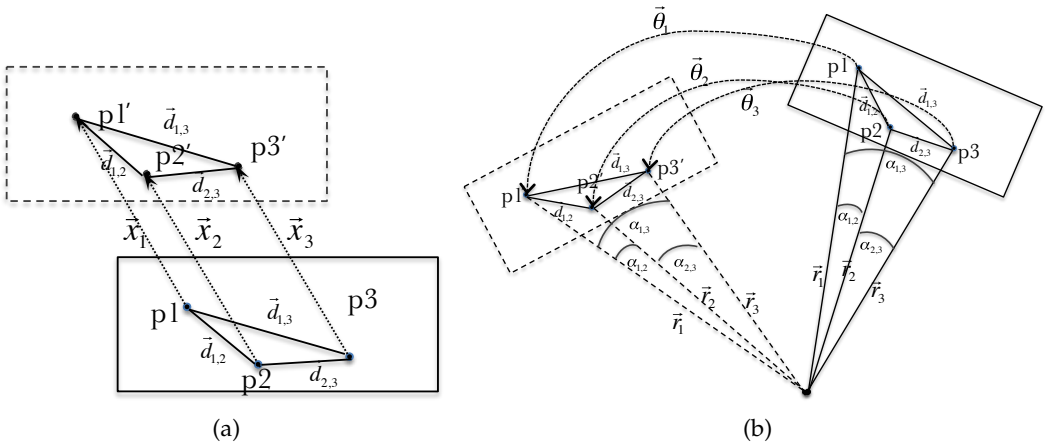


Figure 4.2: Rigid body translation 4.2a and rotation 4.2b

Rigid Body Translation During a translation every point in a rigid body is moved by exactly the same *vector* with exactly the same speed and acceleration. This is illustrated in figure 4.2a. The figure shows a rigid body with three arbitrary points p_1, p_2, p_3. The relative positions of those points are given by the difference vectors $\vec{d}_{1,2}, \vec{d}_{1,3}, \vec{d}_{2,3}$. We assume that the body is translated (=moved in a straight line) randomly which results in p_1, p_2, p_3 being moved by a corresponding vectors $\vec{x_1}, \vec{x_2}, \vec{x_3}$. Per definition of a rigid body the relative positions given by $\vec{d}_{1,2}, \vec{d}_{1,3}, \vec{d}_{2,3}$ must remain unchanged. This is only possible if all the points are moved by exactly the same vector:

$$\vec{x_1} = \vec{x_2} = \vec{x_3} \tag{4.1}$$

This is valid independently of the translation distance and the time it took. Thus, given a translatory motion, at any point in time during this motion, all points of a rigid body will have moved by exactly the same vector. A signal example is given in figure 4.3 to visualize this fact.

This is also valid for infinitesimally small time intervals which implies that at any given point in time the speed and with it the acceleration vectors will also be the same for all points. The velocity and acceleration equation are given bellow.

$$v(t) = \lim_{\Delta t \to 0} \frac{x(t + \Delta t) - x(t)}{\Delta t} = \frac{dx}{dt} \tag{4.2}$$

$$a = \frac{dv}{dt} \tag{4.3}$$

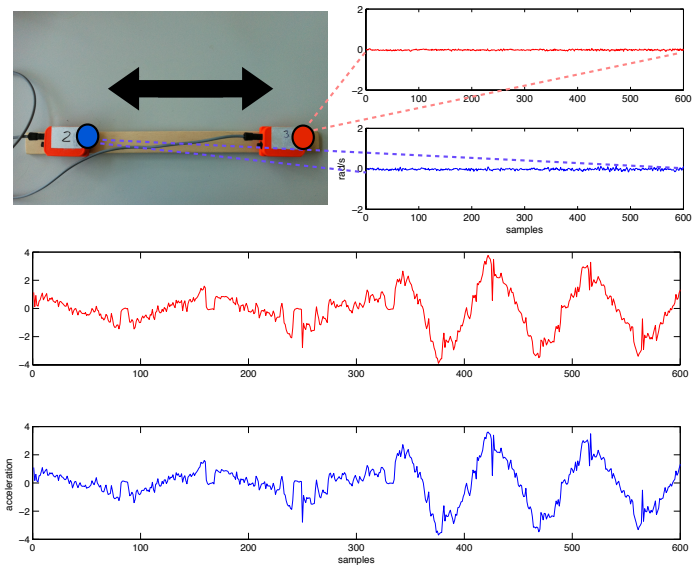


Figure 4.3: Signal sample from a rigid body translation. The gyro signal is on the top right and shows little to no activation. The accelerometer signal is below and is fairly the same for both sensors.

Rigid Body Rotation In an analogous way it can be shown that *angular velocity vector* (and angular acceleration) are the same for all points of a rigid body during a rotation around an arbitrary point in space. To illustrate this, figure 4.2b shows a rigid body in which three arbitrary points p_1, p_2, p_3 and an arbitrary center of rotation r are marked. The vectors connecting each point to the center of rotation are labeled as $\vec{r}_1, \vec{r}_2, \vec{r}_3$, their relative angles as $\alpha_{1,2}, \alpha_{1,3}, \alpha_{2,3}$. We consider a rotation around r which results in p_1, p_2, p_3 being rotated by $\theta_1, \theta_2, \theta_3$. Since per definition of a rigid body after the rotation the relative positions of the three points given by $\vec{d}_{1,2}, \vec{d}_{1,3}, \vec{d}_{2,3}$ must be unchanged, the relative angles between vectors connecting them to the center of rotation $\alpha_{1,2}, \alpha_{1,3}, \alpha_{2,3}$ must also be unchanged. This is only possible if all three points are rotated by the same angle:

$$\theta_1 = \theta_2 = \theta_3 \tag{4.4}$$

As for translation considering infinitesimal time periods, the angular speed ω and acceleration must also be the same for all three points.

In summary, during a rotation of a rigid body around an arbitrary point in space a gyroscope will produce the same signal no matter where in the rigid body it is placed. As will be explained later, this does not apply to accelerometers since different points in a rigid body in general experience a *different, non zero acceleration vector* during a rotation. Again we show a small signal example for illustration in Figure 4.4.

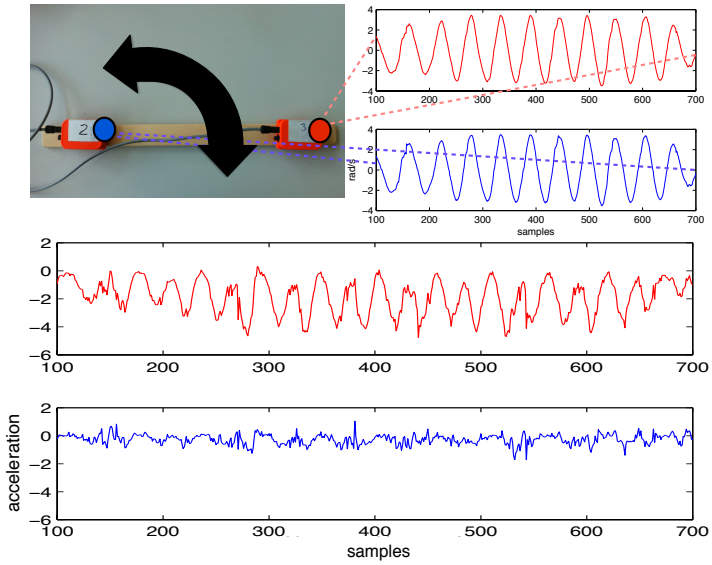


Figure 4.4: Signal sample from a rigid body rotation. As seen, the gyros, on the top right show some activation and are more or less identical. The accelerometer reading for the two sensors differ.

Limits of the Rigid Body Approximation Obviously, the individual segments of the human body are not rigid bodies. Deformation of soft tissue, skin motion and muscle activity associated with most motions all lead to deviations. However, as will be underscored by subsequent experiments (see Section 4.5), for many sensor positions and motions it is a valid approximation. The main deviations from the rigid body approximation can be observed in the following situations.

1. During short, intensive acceleration and follow up vibrations soft 'wobbly' parts (fat, soft muscles) are deformed in a non rigid way. To deal with such deviations the system can simply discard such vibrations or apply the previously mentioned low-pass filtering.

2. When active muscles change shape. In particular large muscles will cause motion signals incompatible with the rigid body approximation. Thus, one should avoid placing sensors directly on top of a well developed biceps. Fortunately, such placement is often not very convenient and users will most probably avoid it.

3. The lower arm rotation parallel to the axis of the arm will affect sensors fixed to the wrist in a significantly different way than sensors near the elbow. The wrist sensor rotates perfectly with the wrist, whereas the elbow sensor will do so to a much lesser degree. Gestures, relying on such rotations as

an important discriminative information present a problem for our location invariant recognition, as illustrated by our 'synthetic gestures' evaluation in Section 4.5.

Acceleration during Rigid Body Rotation

During a pure translation, gyroscopes will provide no signal at all (there is per definition no rotational component) while accelerometers will all give the same readings no matter where they are placed (see figure 4.3 for an example).

As already pointed out, in a rigid body all points are rotated with the same angular velocity (ω) and experience the same angular acceleration α. Thus, the gyroscope signal is invariant with respect to sensor displacement.

To understand the effect of sensor displacement during rotation on the accelerometer signal we need to revisit some basic physics. During a rotation with the angular velocity ω, the linear velocity v of each point of the rigid body depends on the distance to the center of rotation r. The further the point is located from the center, the larger the circle it needs to travel and, consequently, the faster it moves. The speed is defined as:

$$v = \omega r \tag{4.5}$$

The important thing to remember when looking at the above equation is that v designates the speed traveled along a circle. This means that, although the scalar value of the speed is constant (if ω remains constant), to follow the circle each point of a rotating rigid body constantly has to change its direction[1]. Such a change of direction requires acceleration. The direction of the acceleration is parallel to the radius of the circle. The magnitude of this acceleration depends on the speed (the faster a point travels the more force is required to change direction) and with it on the distance for the center. The magnitude of linear acceleration a_ω due to constant angular velocity ω in a point at the distance r from the center of rotation is given by:

$$a_\omega = \omega^2 r \tag{4.6}$$

This gives us the first source of acceleration during a rotation of a rigid body. It is often referred to as centripetal acceleration. The second potential source stems from changes in the rotation speed. Since the linear speed is proportional to the angular velocity and the distance to the center (equation 4.5), it follows that the linear acceleration a_α associated with a change of angular velocity is proportional to the angular acceleration (α) and the distance from the center r:

$$a_\alpha = \alpha r \tag{4.7}$$

[1]The vector $\vec{v}$ is parallel to the tangent of the circle in each point of the rotation

This component is called tangential acceleration.

Since the centripetal acceleration and the tangential acceleration are perpendicular, not parallel, the scalar values given above cannot simply be added to get the total magnitude of acceleration (the Euclidian norm of the acceleration vector). For the sake of simplicity we will just deal with each of them separately [2] Another simplification is to ignore the Coriolis force which acts on objects moving along the rotation axis. By moving more than one joint at a time, it is certainly possible to construct motions of human body parts for which the coriolis acceleration plays a significant role. Yet, motions where this is a relevant component are rare and will not be discussed it in this chapter.

Consequences for Displaced Sensors

What does the above mean for the noise introduced by displacing a sensor within a single, rigid body segment? As already stated a gyroscope signal is displacement-invariant, thus, there is no need to considered it further. For an acceleration signal we need to differentiate between three contributions: (1) the contribution caused by orientation with respect to gravity, (2) the contribution caused by translations and (3) contribution caused by rotation. As explained above, the first two are location invariant. Only the rotation component is location-sensitive.

Displacement Noise in Rotation Related Acceleration Given two points of a rigid body: one with a distance r_1 from the center of rotation and the second one with a distance of r_2, we can compute the acceleration components resulting from constant rotation $a_{\omega,1}, a_{\omega,2}$ and from angular acceleration $a_{\alpha,1}, a_{\alpha,2}$ using equations 4.6 and 4.7. Thus, the signal difference attributed to sensor displacement can be computed as

$$a_{\omega,1} - a_{\omega,2} = \omega^2 r_1 - \omega^2 r_2 = \omega^2 (r_1 - r_2) \tag{4.8}$$

$$a_{\alpha,1} - a_{\alpha,2} = \alpha r_1 - \alpha r_2 = \alpha (r_1 - r_2) \tag{4.9}$$

How relevant this difference is to the recognition not only depends on its absolute magnitude, but also on the signal to noise ratio. This is the ratio of the original signal ($a_{\omega,1}$ or $a_{\alpha,1}$) to the difference caused by displacement ($a_{\omega,1} - a_{\omega,2}$ or $a_{\alpha,1} - a_{\alpha,2}$). It can be computed from equations 4.8 and 4.9:

$$\frac{a_{\omega,1} - a_{\omega,2}}{a_{\omega,1}} = \frac{\omega^2 (r_1 - r_2)}{\omega^2 r_1} = \frac{r_1 - r_2}{r_1} \tag{4.10}$$

$$\frac{a_{\alpha,1} - a_{\alpha,2}}{a_{\alpha,1}} = \frac{\alpha (r_1 - r_2)}{\alpha r_1} = \frac{r_1 - r_2}{r_1} \tag{4.11}$$

[2] Since the two are perpendicular their contribution to the norm of the acceleration $a_{combined}$ is given by $\sqrt{\omega^4 r^2 + \alpha^2 r^2}$

The above is a very compelling result. It shows that sensor displacement noise during rotational movement depends only on the amount of displacement *with respect to the center of rotation*. It is independent of the actual angular velocity or angular acceleration.

Consequences for the Recognition The previous paragraph dealt with the distortion of the acceleration signal related to rotation, as the other components are not affected by displacement. A naive idea for the design of an displacement invariant recognition system is to try to ignore the rotation related component of the acceleration signal and use only the translation and vertical orientation related components.

Unfortunately, in general[3], it is theoretically not possible to decompose an acceleration signal into the three components above. Note that this remains true even if we combine an acceleration sensor with a gyroscope. The gyroscope will indicate the presence and speed of rotation. As shown in equation 4.8, to compute the acceleration we need the distance from the center of rotation, which we, however, do not know.

Fortunately, although we do not know the exact radius of the rotation, we know that it is bounded by the dimensions of the human body. While rotational motions with very high radius can be constructed, for most human limb motions the center of rotation is somewhere close to the torso. This means, that for a given rotation speed, the acceleration is unlikely to exceed a certain value. Therefore, we can use the ratio of rotation velocity measured by a gyroscope to the norm of the acceleration vector computed from the acceleration sensor signal, to determine if the acceleration signal is rotation dominated or not. A high acceleration with a relatively low measured angular velocity is a good indication of the signal not being dominated by rotation. On the other hand, high angular velocity with low or moderate acceleration is an indication of a rotation dominated motion.

The ratios of angular velocity to acceleration norm signifying the transition between rotation and translation dominated signal depend on the typical rotation radius and with it on the motions relevant to the specific recognition task. They have to be learned during training. We will show an example in the next paragraph.

Thus, while we can not separate the individual components of a given acceleration signal, it is possible to estimate with reasonable probability which signal frames are dominated by rotation and which are not. We can then discard the rotation dominated frames, which are sensitive to displacement and use only the ones dominated by translation and/or vertical orientation. In a sensor setup with

[3]The general case assumes that there is no additional information such as further sensors in different locations on the same body part or appropriate high level knowledge about the form and constraints of the motion

a gyroscope we can try to substitute the rotation for the discarded acceleration frames to retain rotation related information.

Another interesting consideration relates to the vertical orientation component of the acceleration signal. Any (non free falling) object on Earth is subject to a constant 9.81 m/s^2 acceleration. This means that if the norm of the acceleration signal is close to 9.81, then the signal is likely to be dominated by the vertical orientation component. Clearly, this heuristic is not always valid. We can imagine a situation when an object is free falling while experiencing a 9.81 m/s^2 side acceleration. However this is a rare occurrence, and the above assertion is mostly valid (as will be underscored by the experiments in the next section).

In Summary

To conclude the theoretical analysis, we outline our findings regarding the signal level and give useful tips for building recognition system more robust to displacement.

Signal Level Summary The results of the discussion presented in this section are summarized in the following:

1. Gyroscopes are insensitive to sensor displacement within a single rigid body segment. However they capture only information about the rotational motion component. They fail to capture information about translational motions and the vertical orientation (orientation with respect to gravity).

2. The accelerometer signal is a sum of acceleration due to rotation, acceleration due to translation and acceleration due to orientation with respect to gravity.

3. Acceleration due to translation and orientation with respect to gravity are independent of sensor placement within a rigid segment of the body.

4. Acceleration caused by rotational motion is placement-sensitive. The ratio of the corresponding acceleration signal to the 'noise' introduced by sensor displacement is proportional to the ratio of to the amount of displacement *with respect to the center of rotation*.

5. Using an acceleration (and possibly gyroscope) sensor at one location only, it is not possible to separate the three above mentioned acceleration components (rotation caused, translation caused and gravity caused). Thus, given an acceleration signal we are not able to remove the rotation related component (which is sensitive to displacement noise) and use the two other components (which are not displacement sensitive) for classification.

6. However, given an acceleration and a gyroscope measurement (from the same location taken at the same time), we can estimate the contribution of each of the three components in the following way:

   - If the norm of the acceleration vector is close to 9.81 (Earth gravity) then the signal is most probably dominated by the gravity component (vertical orientation).
   - If the norm of the acceleration vector is not close to 9.81 then we look at the ratio of the norm of acceleration minus 9.81 to the angular velocity and the angular acceleration. If the angular velocity or angular acceleration dominate the ratio, we know that the acceleration signal is dominated by the rotation related components. Thus, the acceleration signal is strongly location-dependent. If the acceleration norm (minus 9.81) dominates, we know that the acceleration signal is determined by translation related acceleration. In this case, the acceleration signal is reasonably location-independent.

   If none of the above applies then the acceleration signal is a mixture of the three contributions with none clearly dominating.

7. Low pass (pass frequency below 2 Hz) filtered acceleration signals are likely to be dominated by the gravity component (see [6]).

Recommendation for Recognition For designing on-body activity recognition systems based on motion sensors that are as insensitive as possible to sensor displacement, the following recommendations can be made:

1. If the relevant activities are mostly determined by rotational motions then placement invariance (within a rigid segment of the body) can be achieved by using gyroscopes instead of accelerometers.

2. For general activities the best location insensitive sensor setup consists of an accelerometer and a gyroscope. The procedure can be summarized as follows:

   a) If there is a significant gyroscope signal, we use it as primary source of information

   b) To decide what to do with the accelerometer signal, we look at the ratio of the total acceleration (norm of the acceleration vector) to the total rotation (norm of the angular velocity vector). The accelerometer signal is used for classification, if it dominates both ratios. Otherwise it is ignored (e.g. acceleration input to the classifier set to zero).

The above procedure 'looses' information in two cases. First, a motion with fast rotation or large angular acceleration is combined with a significant amount of linear acceleration, the above rule leads to the acceleration signal being ignored. This is the price we have to pay for location invariance. Second, in cases with large rotations we loose information about vertical orientation. Using strongly low pass filtered acceleration signal as an additional feature can, in most cases, retain at least some of the vertical orientation information.

3. If only an accelerometer is available, then the best we can do is to identify the segments of the signal that are dominated by the gravity component and base the recognition solely on the information about vertical orientation. This may sound like loosing a lot of information, however, previous work ([6, 20]) has shown that many activities are to a large degree determined by vertical orientation and changes thereof.

4. Independent of the recognition modality training the system with two sensors as far displaced as possible should encourage the classifier to focus on location invariant parts of the signal.

4.5 Evaluation on Synthetic Motions

As an initial evaluation we look at the following eight 'synthetic motions" of the forearm:

a move up,
b move straight out,
c move from left to right,
d close elbow joint,
e move back (closing elbow joint) and turn wrist in one motion,
f turn around shoulder joint (screw-driving),
g turn large circles around shoulder,
h turn smaller circles around elbow.

The above motion set was put together to contain both 'easy' and 'hard' gestures and illustrate the strengths and weaknesses of our approach. Thus, for example, gestures *d* and *e* differ mostly in the turning of the wrist. Wrist turning is especially displacement-sensitive because of deviations from the rigid body approximation. On the other hand gestures *a* and *b* seem to be well suited for our approach. Note that many typical arm activities are to contain motions from the above set.

The lower arm was chosen for two reasons. First, it is a likely place to wear accelerometers (watch etc.). Second, the forearm has the most degrees of freedom and is the body part that is able to move the fastest.

Sensor Setup We use XBus Master System together with six MTx motion sensors equipped with a 3-axis accelerometer, gyroscope and magnetic field sensors. We focus on the location within a single body part and ignore the question of sensor orientation. Thus, for all sensors, the x-axis orientation is the same (pointing towards the ground if the arm is in rest). The six sensor are placed as follows, (1) wrist outside, (2) wrist inside, (3) middle of segment outside (y axis orientation same as 1), (4) middle of segment on top of arm (y axis orientation 90 degrees to 3 and 1), (5) close to elbow inside and finally (6) close to elbow outside.

Signal Level Evaluation

Before we proceed to classification experiments we use the synthetic gestures data to validate the basic assumptions behind our approach. First we check if leaving out signal segments with large angular velocity to acceleration ratio does indeed reduce the displacement related noise in the acceleration signal. To this end, in figure 4.5 (left) we have plotted the difference in signals between all sensors locations (in percent of the sensor signal) against the acceleration norm divided by angular velocity norm. In the rest of this chapter, we will refer to the difference in signals between all sensors locations in percent as displacement noise. As long as the ratio is large (above 300) the signal difference is very close to zero. This means that displacement has nearly no effect on the signals. As the ratio gets smaller and angular velocity starts to dominate we begin to get a spread in the displacement noise. For very small values there is significant noise. This confirms our basic assumption.

It is interesting to see that a similar relation holds true, even if we plot the noise against just the angular velocity (see figure 4.5). This indicates, that motions with a high angular velocity component are relatively rare in the experimental data.

Next, we check the assumption that frames where the norm of the acceleration is close to 9.81 (gravity) are likely to contain orientation information and, thus, no displacement related noise. This is illustrated in Figure 4.5 (right). For acceleration norm values within 1g, the noise is negligible. In summary, our assumptions hold well on the test data set.

Recognition Experiments

Subsequently, we test if the validity of our assumptions will actually translate into recognition results. To this end we first train the system on two locations. We use

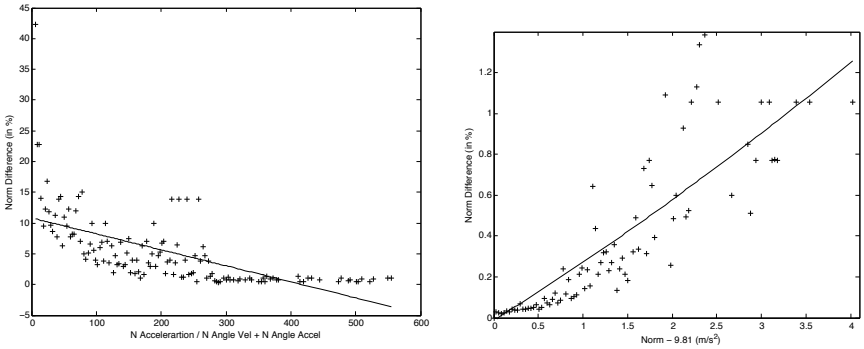


Figure 4.5: Left: Difference in Percent plotted against the Norm Acceleration divided by, the Norm Gyro Vector: Right Difference in Percent against acceleration norm - 9.81

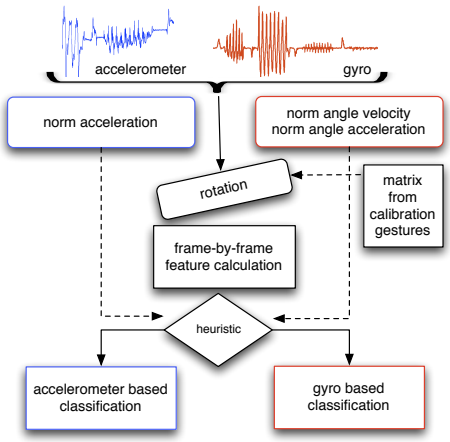


Figure 4.6: Overview about the classification method used, showing the heuristic cut-off.

two locations to be able to support the claim that training different locations helps the system learn the displacement invariant features. We then test the system on the locations it was trained on, as well as, on three additional locations. We do it with and without our heuristics and compare the results.

Classification Method In the following, we detail the classification method summarized in figure 4.6. Applying 1 sec. sliding window we extract 45 standard pattern recognition features for each accelerometer and gyroscope axis. Concerning sensor orientation, we use normalized axes. For the evaluation of synthetic motions this is extremely simple, as most of the sensors have the same orientation anyway. For the later two evaluations two calibration gestures are performed between recording the motions. This allows us to determine two normalized axes

Table 4.1: Classification comparison for the synthetic motions using a majority decision over the motions based on a Knn classifier. Acceleration cut-off Norm - 9.81 at larger than 0.8. Decision Boundry for combining accelerometer and gyro at 300.

Modality	Same	Trained on 1	Trained on 2
Acceleration	100 %	33%	35%
Gyroscope	65%	43%	44%
Cut Off	-	42%	47 %
Combined	-	78%	85%

Table 4.2: Combined Accelerometer and Gyro trained on 2 evaluated on 4 Sensors Accuracy 85 %, Decision Boundary at 300.

a	b	c	d	e	f	g	h	← classified as
100	0	0	0	0	0	0	0	a = move up
0	100	0	0	0	0	0	0	b = move straight
0	0	100	0	0	0	0	0	c = left to right
0	0	39.1	60.9	0	0	0	0	d = close elbow joint
0	0	28.6	0	71.4	0	0	0	e = back and turn wrist
0	0	14.3	0	0	85.7	0	0	f = turn around shoulder joint
0	0	0	0	0	0	100	0	g = large circles (shoulder)
0	0	0	0	0	0	0	100	h = small circles (elbow)

from the accelerometers due to gravity. For all classifications we use the two normalized axes (defined as x and y) for feature extraction and only the magnitude of z, as we cannot determine its direction using the acceleration.

Using the entropy measure also applied in the C4.5 decision tree, we reduced our feature set from 40 to 8 (mean, variance, number of peaks, median peak height, FFT center of mass, RMS, and frequency range power) depending on the evaluation. Each feature is calculated over the accelerometer and gyro data. The gyro data is normalized the same way as the accelerometer.

We classified all examples using several frame-by-frame classifiers(C4.5, KNN, BayesNets). As all of them show more or less comparable results, we pick KNN for the analysis for the remainder of the chapter.

Classification Results The results are summarized in table 4.1. Training the classifiers on training data and test data from one distinct sensor we reach a classification rate of 100 % using both frame by frame classification and a majority decision window over complete gestures. Testing the trained system on locations that it was not trained on reduces the recognition rate to 33% (on the accelerometer only). Having trained the system not on one, but on two widely displaced sensors improves the recognition rate to 35% (only a 2% increase). Getting rid of

frames with high acceleration improves the recognition by about 10%, the performance remains though poor.

The gyroscope performs significantly worse then the accelerometer (65% on the same location) confirming our analysis that it fails to capture all relevant information. When tested on a different location it drops to 43%. This is not a dramatic drop compared to the accelerometer results, yet still significant. We expected the gyro to be invariant with respect to displacement. The explanation is the inclusion of gestures with wrist rotation, which violates the rigid body assumption.

As expected best location invariant recognition results from a combined accelerometer/gyro based approach with all rotation dominated accelerometer frames being ignored. Trained on one sensor we reach 78%, on two we come up to 85%.

In summary the initial experiment confirms that our heuristic works well. Clearly 85% is far from perfect, but for many applications it may be acceptable (as opposed to 33%). The result is particularly significant because we were working with large displacements. Small displacements typical of 'slipping sensor' are likely to lead to a much less significant reduction in recognition rate (we have shown, that the noise is proportional to the displacement with respect to the center of rotation).

Another important observation is the confusion matrix that corresponds to the 85% recognition rate (Table 4.2). It can be seen that out of the eight gestures five achieve 100% recognition. The confusions involve gestures with significant wrist rotations. We have identified such rotations as one of the cases where the rigid body assumption underlying our heuristics is not valid.

4.6 Evaluation on Typical Recognition Tasks

The rigid body approximation approach is promising. Yet, how useful is the method in realistic situations? We evaluate it on two real life recognition tasks, the first is a modes of locomotion experiment, a well understood and often required context type, the second gym exercises.

Modes of Locomotion Experiments

Towards a more realistic evaluation, we first look at an extended modes of locomotion problem with the sensors placed on the upper leg. We differentiate eight activities (table 4.3). Note that this is not the trivial walking/standing/sitting modes of locomotion problem, but an experiment involving fairly subtle differences.

Table 4.3: Motions classified in the Locomotion and Gym exercise scenarios

Locomotion	Gym Exercises
i walking	*q* lat machine
j running	*r* pectorial
k running uphill	*s* shoulder press
l biking	*t* upper back
m rowing	*u* arm extension
n stairs	*v* arm curl
o skiing	*w* pull down
p crosstrainer	*i* chestpress

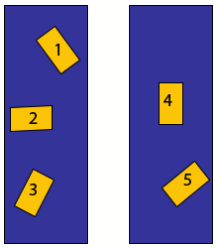


Figure 4.7: Random generated sensor placement and orientation for the leg(front and back).

Experiment Setup The subject's upper leg is equipped with 5 MTx Sensors three mounted on the front and two on the back as seen in Figure 4.7. The placement is generated using a uniform random distribution. We use bandages to attach the sensors. Overall eight locomotion classes (Table 4.3) were recorded on fitness machines in a fitness center. One test subject performed them each for 5 min.

Results The results are summarized in table 4.5. Training and classifying on the same acceleration sensor gives an accuracy between 95 and 100 % using 10 fold cross validation or 66 % percentage split using a KNN and a majority decision window. As expected, a gyroscope performs worse with 80% accuracy on the same location. Note that the leg motions are very much rotation determined, so the performance reduction for the gyro is less pronounced than for the synthetic gestures from the previous paragraph.

Testing the same methods on locations they have not been trained on leads to a recognition rate of 63% on the accelerometer and 72% on the gyro. As we have less deviations from the rigid body assumption the drop in gyro recognition rates is smaller than for the synthetic arm gestures. The drop is probably due to muscle motions (these motions are significant in some locations on the upper leg).

Table 4.4: Joint Accleerometer and Gyro trained on two sensors, evaluated on three, 90 % decision boundary at 150.

i	j	k	l	m	n	o	p	← classified as
100	0	0	0	0	0	0	0	i = walking
0	76.9	23.1	0	0	0	0	0	j = running
0	20.3	79.7	0	0	0	0	0	k = uphill
0	0	0	90.4	9.6	0	0	0	l = biking
0	0	0	0	100	0	0	0	m = rowing
0	0	0	0	0	91.1	8.9	0	n = stairs
8.7	0	0	0	0	0	91.3	0	o = skiing
6.1	0	0	0	0	0	0	93.9	p = crosstrainer

Table 4.5: Classification comparison for the locomotion exercises using a majority decision over the motions based on a Knn classifier. Acceleration cut-off Norm - 9.81 at larger than 0.6. Decision Boundary for combining accelerometer and gyro at 150.

Modality	Same	Trained on 1	Trained on 2
Acceleration	100 %	63%	65%
Gyroscope	80%	72%	75%
Cut Off	-	72%	76%
Combined	-	87%	90%

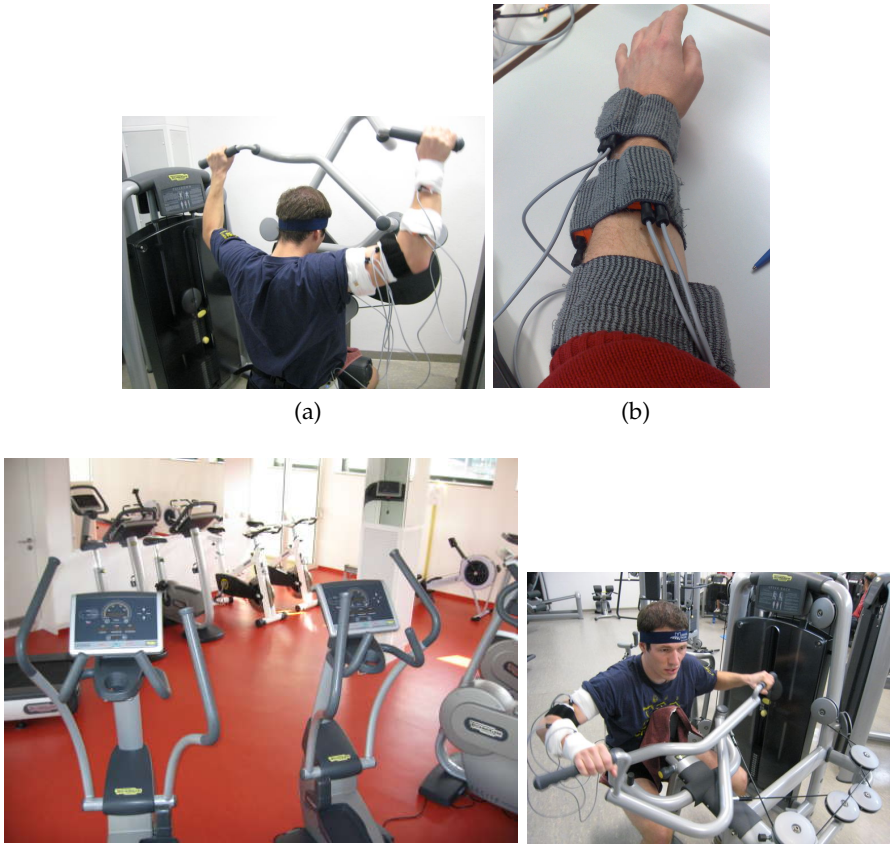
Training on two locations brings minimal improvement. Restricting the acceleration frames to those with a norm close to 9,81 improves the recognition by 10% amounting to 76% when trained on two sensors. The relatively good recognition rate (for a displaced, acceleration only system) is due to the fact that most of the relevant motions are largely determined by changes in vertical orientation of the upper leg.

The combined accelerometer gyro heuristic (throwing away rotation related acceleration frames) brings the recognition rate to 90% (trained on two sensors).

In summary, the modes of locomotion experiment also confirms the validity of our methods. For many practical applications the 90% recognition rate can suffice. We were working with large displacements and the noise is proportional to the displacement.

Gym Experiments with Sensors on Forearm

The most challenging evaluation focuses on muscle strength exercises conducted at fitness center machines.



(a) (b)

(c) (d)

Figure 4.8: Pictures form the gym experiment data recording, the locomotion exercises and the artificial gesture recording.

Experiment Setup The sensor placement and orientation are generated random. There are four sensors at the forearm placed as follows. The first around 10 cm away from the elbow on the outside of the arm , x axis angle around 90° turned from an orientation that is parallel to the arm pointing towards the ground, the second on the wrist, with approximately 50°, the third placed at the inside of the arm around 8 cm away from the wrist with 0°, the forth placed also on the inside closer to the elbow at 10°. Again we picked 8 gym exercises to record, as shown in Table 4.3. One test subject performed each exercise 20 -25 times. Two runs were conducted for each of the two test subjects.

The feature extraction follows the approach laid out for synthetic motions and leads to the same features. We use a 1 sec. sliding window. The recognition task

Table 4.6: Classification comparison for the gym exercises using a continuous HMM. Decision Boundary for combining accelerometer and gyro at 300.

Modality	Same	Trained on 1	Trained on 2
Acceleration	97%	24%	31%
Combined	-	74%	82%

is much harder than the modes of locomotion problem and the majority decision using the acceleration trained and evaluated on the same location gives only an accuracy of 85%. We thus turned to a continuous HMM based approach. On top of the extracted features, we apply a 15 sec. sliding window using three Gaussians for each feature and four hidden states. In case of combining the gyro and accelerometer data, we picked the decision boundary 300 for the ratio. If the ratio is below the boundary, we use the gyro features and set the accelerometer features all to zero.

Results The results are summarized in table 4.6. When training and testing on the same location (again 66 % percentage split) we reach 96 % on the acceleration signal alone. Testing the acceleration only system on a location that it was not trained with drops the recognition rate to 24 %. This was to be expected, as the classification problem is fairly complex. The rate can be raised to 31% by training the system on two sensors, which is significant but not really useful. A gyro trained and tested on the same location gives an accuracy of 62% again confirming that the gyro signal 'looses information'. [4] However, a significant improvement on displaced sensor is achieved with our combined gyro/accelerometer approach. Trained on one sensor we reach 74% on two we come up to 82%. The confusion matrix for this case is shown in Table 4.7.

Considering the confusion matrix of the combined accelerometer and gyro case, the really significant miss-classifications happen between movements that train the complementary muscles, for example arm extension and arm curl.

4.7 Conclusion

We have shown that a combination of an accelerometer that ignores rotation dominated signal segments and a gyroscope to compensate for the lost rotation information is reasonably robust with respect to sensor displacement within a single

[4]Since the problem is harder than the previous ones it was not to be expected that dropping high acceleration frames form the accelerometer classification will lead to reasonable performance. Since the HMM evaluation was more time consuming than the majority decision from previous examples we did not take the time to evaluate this approach.

Table 4.7: Confusion Matrix Joint Accleerometer and Gyro trained on 2 Sensors eval on 2 82 % decision boundary at 300

q	r	s	t	u	v	w	x	← classified as
75.6	0	0	0	0	0	0	24.4	q = lat
0	81.6	0	0	0	0	18.4	0	r = pectorial
0	0	88.6	0	11.4	0	0	0	s = shoulder press
0	0	0	100	0	0	0	0	t = upper back
0	0	13.3	0	76.7	0	10.0	0	u = arm extension
0	0	0	0	22.2	77.8	0	0	v = arm curl
12.0	0	0	0	8.0	0	80	0	w = pull down
0	0	0	20.8	0	0	0	79.2	x = chestpress

body part. We have shown that randomly, significantly displaced sensors can reach up to 90% of the recognition rate of a non displaced sensor. Compared to testing an unmodified classification system on a different location we can improve the recognition rate by over 300%.

Clearly, 90% accuracy will not suffice for all applications. The need to add a gyroscope (which is more expensive than an accelerometer) may also not always be acceptable. Regarding the experimental evaluation, the sensors were tightly attached to the limbs. It remains to be seen if there are significant changes applying this approach to loosely attached sensor devices.

We believe that our heuristics are a significant improvement over the current state of the art and is acceptable in many cases. However, there are some alternatives that might circumvent some of the inherit problems of the approach introduced. We present them in the following:

Time series inference – So far we showed our approach working for frame-by-frame inference. It should work also with time series algorithms, except for one additional problem. The decision to use the accelerometer or the gyroscope classifier is done frame-by-frame. If during one activity, several decision switches happen, it is undefined which time-series classifier to use (either one for acceleration or the one for gyroscopes). Identifying segments tainted by rotation is still possible, the classification becomes more difficult in this case.

Continuous displacement tracking – One way to utilize the rigid body approximation introduced in this chapter for time-series inference, one can extend the work combining it with the research of Foster et. al. to enable continuous displacement tracking [4, 5]. This would enable us to provide a transformation matrix for every data sample and the usage of "unaltered" activity recognition algorithms. Unsolved problems are the detection of the

initial position of the device and how to apply this to a single motion sensor, as Foster et. al. use several accelerometers. Another possibility is to include multiple motion sensors with known orientation.

Magnetic compass integration – Combining a magnetic compass to the inference might gain additional stability and improved displacement robustness. However, in this case, the physical assumptions need to be revisited to include the new modality.

Modeling plasticity – Although deformable body physics is more complex and proven unnecessary for the approach presented. It might be worthwhile to revisit this theory, especially when dealing with very loosely attached sensing devices or sensors embedded in garment.

This chapter focuses on large displacement that would be typical of a user being given no instructions on where to place the device. We also wanted to explore the limits of our ideas. Next we will investigate in more detail smaller displacements that may be more typical of shifted sensors. As described in this chapter displacement noise is proportional to displacement distance. Thus for smaller displacements and with some further improvements, we believe that our methods could work satisfactorily even for accelerometer only systems. Under such circumstances accelerometer/gyro system might be able to mask out displacement noise entirely.

4.8 Bibliography

[1] R. Aylward and J. Paradiso. Sensemble: a wireless, compact, multi-user sensor system for interactive dance. *Proceedings of the 2006 conference on New interfaces for musical expression*, pages 134–139, 2006.

[2] G. Borriello, W. Brunette, J. Lester, and P. Powledge. An ecosystem of platforms to support sensors for personal fitness. *Proceedings of the International Workshop on Wearables*, Jan 2006.

[3] H. Brashear, T. Starner, P. Lukowicz, and H. Junker. Using multiple sensors for mobile sign language recognition. *Wearable Computers, 2003. Proceedings. Seventh IEEE International Symposium on*, pages 45–52, 2003.

[4] K. Forster, P. Brem, D. Roggen, and G. Troster. Evolving discriminative features robust to sensor displacement for activity recognition in body area sensor networks. In *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2009 5th International Conference on*, pages 43 –48, 2009.

[5] K. Forster, D. Roggen, and G. Troster. Unsupervised classifier self-calibration through repeated context occurences: Is there robustness against sensor displacement to gain? In *Wearable Computers, 2009. ISWC '09. International Symposium on*, pages 77 –84, 2009.

[6] N. Kern, B. Schiele, H. Junker, P. Lukowicz, and G. Tröster. Wearable sensing to annotate meeting recordings. *Personal and Ubiquitous Computing*, 7(5):263–274, 2003.

[7] A. Krause, D. Siewiorek, A. Smailagic, and J. Farringdon. Unsupervised, dynamic identification of physiological and activity context in wearable computing. *Wearable Computers*, Jan 2003.

[8] A. Krause, A. Smailagic, and D. Siewiorek. Context-aware mobile computing: Learning context-dependent personal preferences. *IEEE Transactions on Mobile Computing*, Jan 2006.

[9] J. Lester, T. Choudhury, and G. Borriello. A practical approach to recognizing physical activities. *Proceedings of Pervasive*, Jan 2006.

[10] J. Lester, B. Hannaford, and G. Boriello. Are you with me?–using accelerometers to determine if two devices are carried by the same person. *Pervasive Computing: Second International Conference*, Jan 2004.

[11] J. Mantyjarvi, J. Himberg, T. Seppanen, and N. R. Center. Recognizing human motion with multiple acceleration sensors. *Systems, Man, and Cybernetics, 2001 IEEE International Conference on*, 2, 2001.

[12] U. Maurer, A. Smailagic, D. Siewiorek, and M. Deisher. Activity recognition and monitoring using multiple sensors on different body positions. *Proceedings of the International Workshop on Wearables*, Jan 2006.

[13] C. Randell and H. Muller. Context awareness by analysing accelerometer data. *The Fourth International Symposium on Wearable Computers*, 1(1):175–176, 2000.

[14] D. Roggen, N. Bharatula, M. Stager, P. Lukowicz, and G. Troster. From sensors to miniature networked sensorbuttons. *Proceedings of the 3rd International Conference on Networked Sensing Systems (INSS06)*, 2006.

[15] T. Stiefmeier, D. Roggen, G. Ogris, P. Lukowicz, and G. Troster. Wearable Activity Tracking in Car Manufacturing. *IEEE Pervasive Computing*, 7(2), Apr. 2008.

[16] K. van Laerhoven and A. Aronsen. Memorizing what you did last week: Towards detailed actigraphy with a wearable sensor. *Proceedings of the 27th International Conference on Wearable Computing*, Jan 2007.

[17] K. Van Laerhoven and O. Cakmakci. What shall we teach our pants? *Wearable Computers, 2000. The Fourth International Symposium on*, pages 77–83, 2000.

[18] K. van Laerhoven and H. Gellersen. Spine versus porcupine: a study in distributed wearable activity recognition. *ISWC 2004*.

[19] T. Westeyn, K. Vadas, X. Bian, T. Starner, and G. Abowd. Recognizing mimicked autistic self-stimulatory behaviors using hmms. *IEEE International Symposium on Wearable Computers*, pages 164–169, 2005.

[20] A. Zinnen, K. van Laerhoven, and B. Schiele. Toward recognition of short and non-repetitive activities from wearable sensors. *European Conference on Ambient Intelligence 2007*.

5

Device Orientation

"If you do not change direction, you may end up where you are heading."
- Lao Tzu, Tao Te Ching

On-body device orientation influences a variety of sensing modalities common in context recognition systems. We discuss the effects orientation changes have on these modalities and present a method to infer the orientation from the acceleration signal of a mobile device carried in a pocket. Whereas previous work has shown how to determine the orientation in the vertical plane (angle towards earth gravity), we demonstrate how to compute the orientation within the horizontal plane. To validate our method we compare the results with GPS heading information when walking in a straight line. On a total of 16 different orientations and traces we get a mean difference of 5 degrees with 2.5 degrees standard deviation.

Kai Kunze, Paul Lukowicz, Kurt Partridge, Bo Begole, Which Way Am I Facing: Inferring Horizontal Device Orientation from an Accelerometer Signal, *13th IEEE International Symposium on Wearable Computers*. Linz, Austria, 2009.

The last type of finer grain placement variations we deal with is device orientation changes with respect to the user's body (see Figure 5.1). When talking about orientation changes we mean changes of the reference system of the sensor , e.g. turning a sensor 180 degrees around an arbitrary axis. The global position of the sensor stays unchanged. For example, a mobile phone or other smart device can be put into the pocket with the front facing towards or away from the body. In addition, devices may rotate around the axis perpendicular to the body, especially

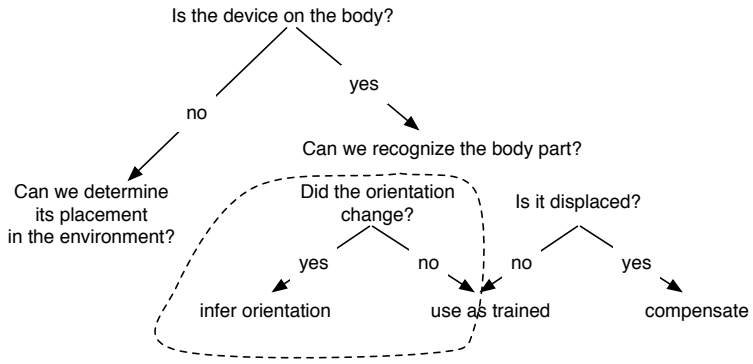


Figure 5.1: Thesis overview with the central question for this chapter highlighted.

if they are small and lose in the pocket. Further, consider a mobile phone in a hip holster; usually, the device can be put into the holster in at least two ways: display facing inwards or outwards. Sometimes there may also be the option of attaching the holster either vertically or horizontally. Finally, the holster can be placed at various locations around the hip, which determines the exact horizontal orientation. Similar considerations are possible with respect to the placement inside the pocket. Especially in large trouser pockets, significant variations in the orientation can occur when the device drifts from one side of the pocket, e.g. on the front side of the leg, to the other, e.g. on the side.

For recognition systems using motion sensors, orientation information can be important in two ways. First, it can improve recognition performance by providing more detailed features (e.g. using all 3 individual axes of an accelerometer) than the orientation invariant vector norm. Second, it can in itself be a relevant piece of information.

The next section shows how orientation changes influence common sensor modalities. Afterwards, we detail related work to get a grasp on already published approaches to the problem, followed by an introduction and evaluation of our method to determine the on-body device orientation in a trouser pocket based on acceleration signals from the smart device.

5.1 Impact of Device Orientation

As to be expected motion sensors are highly affected by device orientation changes relative to the user's body. The user's motion is distributed differently on the axes of the sensor depending on the device orientation. For example, a user walks a bit, stops, takes out his mobile phone and places it back in the pocket turned

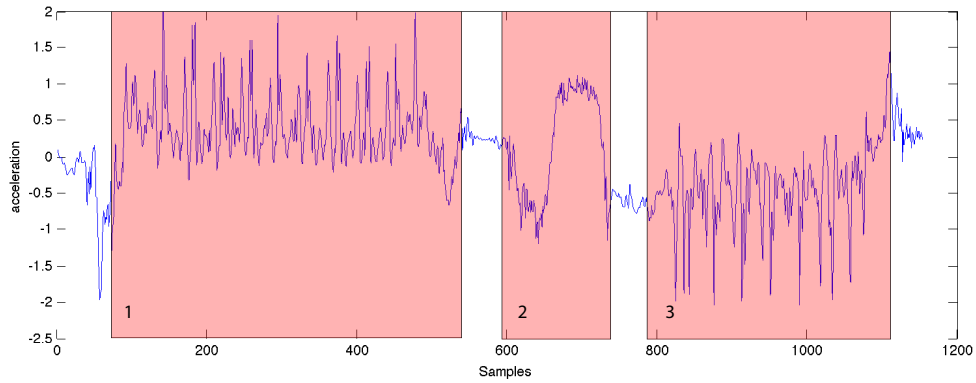


Figure 5.2: Effect of an device orientation change on one axis of an accelerometer with the user walking in phase 1, changing the orientation in 2, and walking again in 3.

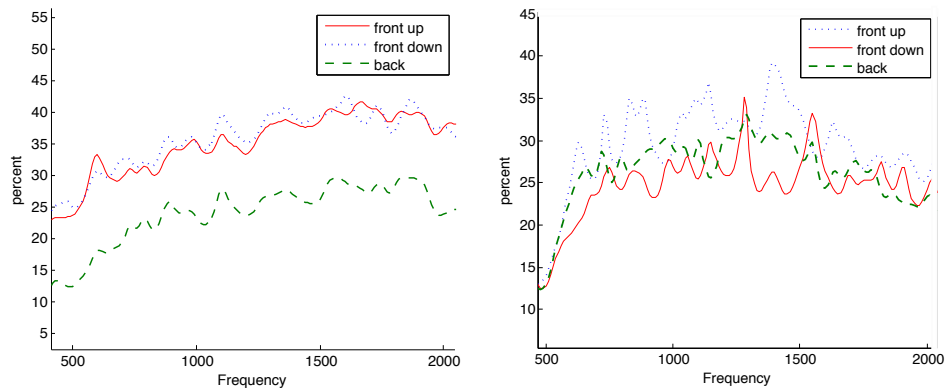


Figure 5.3: Audio dampening depending on device orientation in a trousers pocket, in the left plot for the iphone and on the right for the nokia 810i.

by 180 degrees before continuing walking. The effects of this scenario on an accelerometer axis is illustrated in Figure 5.2. The user is walking with the device in the pocket during phase 1 indicated in the plot. The user takes out the device turns it 180 degrees (marked in phase 2), puts it back and continues walking in phase 3. With accelerometers not only the dynamic acceleration component (the user's motion) is distributed differently depending on the orientation, but also the static gravity component shifts between axes. This is recognizable in Figure 5.2, as the signal offset changes from 0.5 to -0.5 g from phase 1 to phase 3 (see the Displacement Chapter, Section 4.1 for more details).

For sound, specific orientations can lead to the microphone being directed towards the user rather than towards the environment. This can significantly in-

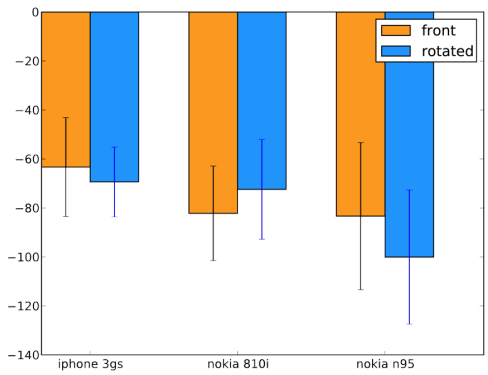


Figure 5.4: The Figure shows the variation of WiFi signal strength measured for three different devices in different orientations.

crease signal damping. Fix rules for the influence of orientation on signal damping are difficult to formulate since the effect is strongly dependent on device and clothing configuration. This is illustrated in Figure 5.3 showing the influence of the orientation on the audio signal absorption spectrum for the Nokia 810i and the iphone 3gs. We play white noise and record it with the respective device and changing device orientation in a trousers pocket. Figure 5.3 depicts the frequency spectrum of the recorded audio. The Nokia 810i PDA has its microphone on the front side. When in the trousers pocket damping across all frequencies is larger if the device is facing towards the body than when it's pointing away. As a counterexample the same Figure 5.3 also shows the results of this simple experiment done using the iphone, which has the microphones on the lower side. Here orientation changes have only little effect. Depending on where the microphone is located on the device, different orientation will cause it to be more or less obstructed by the body or clothing.

For WiFi signal strength orientation effects depends highly on the position and properties of the antenna. To underline this we conduct a small test with three commodity handheld devices, the nokia 810i, n95 and the iphone 3gs. We place them in the trousers pocket recording the wlan signal strength for 10 min screen facing front and rotated by 180 degress (facing up side down). As seen in Figure 5.4, depending on the mobile the orientation influences the signal strength differently.

Summing up, all given sensing modalities are affected by device orientation relative to the user body in a non trivial way. Orientation effects on sound and wifi depend highly on the given hardware and other environmental influences (e.g. the dampening of the clothes worn by the user). Therefore, we focus on motion sensors to estimate the on-body orientation of a device.

5.2 Related Work

There is a significant push from research towards mobile phone sensing. Using mobile phones and other smart devices, device orientation becomes a major issue. Thus, there is quite some effort to make mobile phone inference more robust. Again, most researchers focus on orientation robust or indifferent features.

Lester et al. presented a classification algorithm working accurately with data from different locations on the body [5]. Pham and Abselzaher use features based on relative energy distribution to become more orientation independent [7].

Another research effort centers around relative positioning and orientation between devices. For example, Pirkl et al. use magnetic resonance coils to calculate the orientation and distance between devices [8]. This approach implies the use of two devices and gives only relative orientation and distance between them.

The closest to the work presented later in Section 5.4 is research from Henpraserttae et al. [3]. They coin the orientation of the device as classification problem and uses the determined orientation to transform the reference coordinate system. As it is a classification problem the training complexity increases with the number of orientations, also the evaluation, so far, only uses four different orientations.

As seen, there are two approaches to deal with orientation changes in motion based context recognition:

1. to use aggregate features that are more robust or independent to orientation.
2. to try to infer the sensor orientation relative to the users body.

5.3 Loss of Orientation Information

In most cases motion sensors enabled devices are equipped with 3-axis sensors. As a consequence the norm of the signal can be used as a simple, orientation invariant signal. Thus, the interesting question is not as much "what influence rotation has on the motion signals?", but "how much information is lost when discarding orientation sensitive features?" .

The information loss in a very simple context example is illustrated in Figure 5.5. To show how orientation can provide better features, we consider a standard mode of locomotion recognition problem: distinguishing walking upstairs, downstairs, running and level walking. In Figure 5.5 we use euclidean distance as measure of similarity to compare how well those classes are separated by (1) the norm and (2) orientation sensitive signals from the 3 individual axis of an accelerometer signal. Even for such simple classes (which can actually be recognized reasonably well using the norm), orientation sensitive feature contain significantly more information.

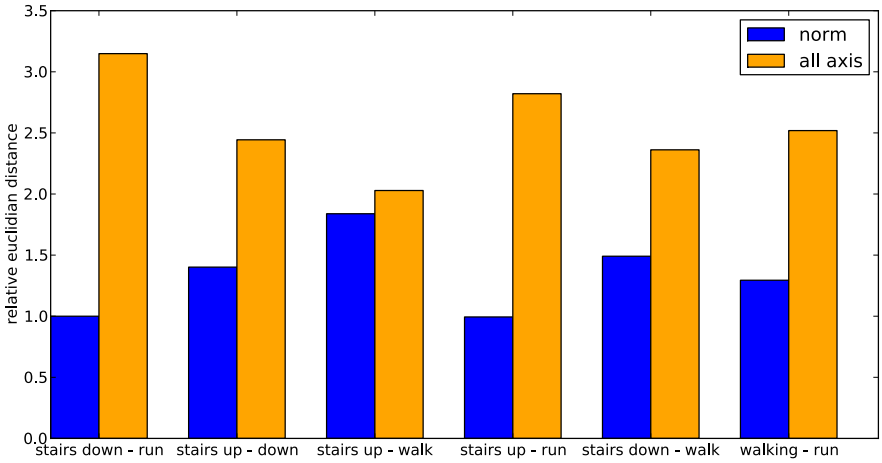


Figure 5.5: Illustrating the information loss for motion sensor based activity recognition, if device orientation information is disregarded. See the text for detail.

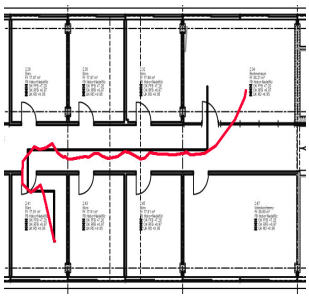


Figure 5.6: Dead-reckoning trace from accelerometer and magnetic field sensor of iphone 3gs, using the accelerometer to count steps and the magnetic field sensor for direction (the black straight lines are the ground-truth, the other path the estimated trace).

As dedicated application feature, device orientation is particularly relevant for systems that contain a magnetic field sensor (as is increasingly the case with modern smart phones). Knowing the orientation of the magnetic sensor axis with respect to the user's body means that the sensor can be used to determine the direction in which the user is facing. The direction the user is facing can indicate the focus of attention and provide information on social interactions, the use of household devices, or interest in specific objects (e.g. a specific shelve in a store indicating interest in certain products). Combined with step detection, a magnetic field sensor placed on the body in a known orientation can also be used as a simple yet effective means of indoor navigation. This is illustrated in Figure Fig-

ure 5.6. It shows a walking path estimation using the accelerometer of an iphone 3gs for step counting and the built-in magnetic sensor to obtain the orientation. The only additional processing done to the graph is to apply a Kalman Filter to the orientation data. This method is fundamentally different (and much simpler) from the standard indoor navigation approach where the information from a 3 axis accelerometer, gyro and magnetic field sensor is integrated over time and the device orientation is not required.

5.4 Detecting Device Orientation

When estimating the orientation of a device with respect to the user's body, two distinct sub-problems must be distinguished: vertical orientation (angle with respect to the gravity vector) and the orientation in the horizontal plane.

Orientation in the Vertical Plane

As has been first proposed by Mizel [6], vertical orientation can be easily estimated when the object experiences no change in motion speed. In this case the only acceleration registered by the sensor is the earth gravity (9.81 m/s^2) and the direction of the measured acceleration vector defines the vertical plane. To identify signal segments with the above characteristics the norm of the measured acceleration vector is used together with its variance. When variance of all axis tends towards 0 and the norm vector approaches 9.81 m/s^2, the signal is very likely to be dominated by the vertical orientation component [1].

Horizontal Orientation

We present a method to infer the orientation of a mobile device carried in a pocket from the acceleration signal acquired when the user is walking. Whereas previous work has shown how to determine the orientation in the vertical plane (angle towards earth gravity), we demonstrate how to compute the orientation within the horizontal plane.

In the following, we show how to infer the orientation of the device with respect to the user's body. We extend existing work by [6] that has shown how the orientation in the vertical plane (angle towards gravity) can be computed by additionally inferring the orientation in the horizontal plane. Our method is based on the observation that while walking, the most variations in the horizontal plane of the acceleration signal will be parallel to the direction of motion. To distinguish between front and back we look at the integral of the signal over time.

[1] In theory this needs not be the case since the object could for example be free falling (no gravity component) while experiencing a constant 9.81 m/s^2 along an arbitrary direction.

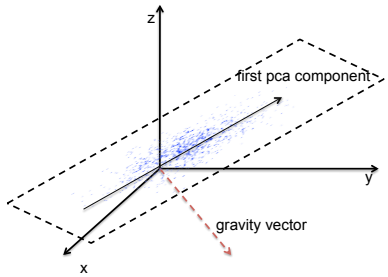


Figure 5.7: Accelerometer coordinate system in relation to the gravity vector (vertical component) and the walking direction infered using pca.

We focus on the trouser pocket, as it is by far the most likely placement for peoples mobile phones (see [4]). This work is partially inspired by Blanke et. al. [1]. We base the approach on three assumptions:

- The user walks facing forward.
- The device is placed in a trousers pocket. Our approach should also yield similar results on the person's torso or similar on-body placements. However, we did not test them in this thesis.
- We apply our approach on a walking segment in which the user walked fairly straight.

As already described by Mizel [6], one can estimate the gravity vector component of a 3-axis accelerometer. We use a slight variation of this method to get the acceleration axis parallel to the persons torso: We apply a sliding window over over all 3 axes. If the variance of all axes is close to 0 and the magnitude approaches 9.81 m/s^2, the signal is very likely to be dominated by the vertical orientation component.

Using this heuristic, we infer the vertical component. Now we project the accelerometer signal in the plane perpendicular to the vertical gravity vector (=horizontal plane). We apply principle component analysis on the projected data points (see Figure 5.7) to get the direction where the acceleration variations is greatest. This is the axis that is parallel to the walking direction. Assuming that the user is walking forward, integration over the component will allows us to determine which way is front (leads to positive integral) and which is back (see Figure 5.10).

5.5 Experimental Evaluation

We test the approach described above with the following setup. We use a MTx motion sensor (equipped with 3-axis accelerometer, gyro and magnetic field) with

Figure 5.8: Sample trace and equipment used for the experimental setup: a mobile phone box, MTx motion sensor with bluetooth adapter, nokia 810 and a gps device.

a custom bluetooth sender placed in a mobile phone casing as data source for our algorithm. As reference we use a GPS device. We stream all data to a Nokia N810 running the context recognition network toolbox[2] for recording and labeling.

In the MTx box, the magnetic field sensor axes are oriented in parallel to the acceleration sensor axes. Therefore, if our algorithm can infer the orientation of the acceleration axis with respect to the user's body we automatically have the orientation of the magnetic field sensor with respect to the body. This is equivalent to knowing which way (in terms of longitude and latitude) the user is facing. If the user is walking forward this direction is also the user's heading. By comparing the heading computed this way with the heading provided by GPS we can verify the accuracy of our method for determining the horizontal orientation of the sensor.

Two test subjects walked a straight path outside (around 30 meters) with the MTx sensor in the right trouser pocket and the GPS in hand. We repeat this experimental setup 8 times per person changing the sensor orientation each time. We pick only 8, comparing the device to a mobile phone, users will most likely not place the device with the thin side facing towards the body. So we place the phone casing with the sensor in the right trousers pocket facing front rotating it 90 degrees further for each experimental trial (the same procedure with the backside facing front), leaving us with 8 distinct orientations.

Three of the 8 orientations are shown in Figure 5.9. Using the approach presented above, we can reliably detect the side of the sensor facing in walking direction for all of the 16 experimental trials. Figure 5.10 shows an incremental integration over the principal component acceleration axis for a person walking slow (blue graph) and walking fast (green graph) with different sensor orienta-

[2]http://crnt.sf.net

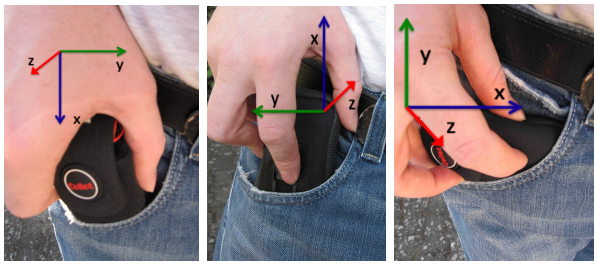


Figure 5.9: The Mtx motion sensor in the phone casing placed in the pocket depicted with the different axis orientations.

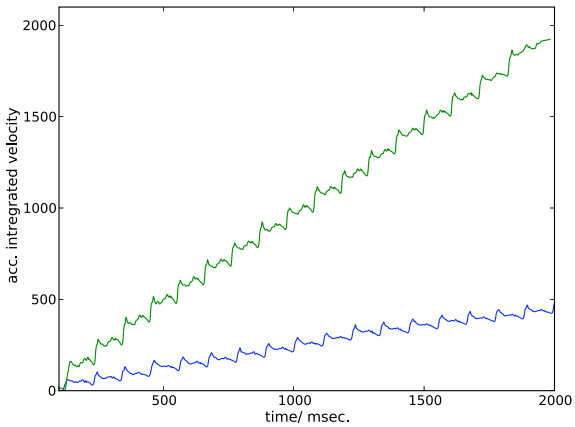


Figure 5.10: The accumulated integrated velocity of the first pca component direction for 2 trials with different sensor orientations, one in which the test subject was walking slowly (blue) and fast (green).

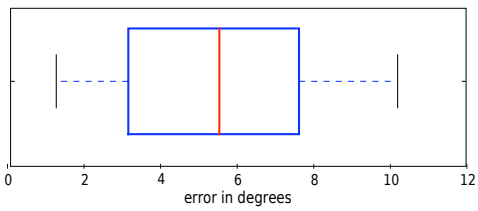


Figure 5.11: The errors between the accelerometer and gps based approaches, mean at around 5 degrees with a standard deviation of 2.5 degrees.

tions.

To validate our method we compare its output with GPS heading information when walking in a straight line. On a total of 16 different orientations and traces we have a mean difference of 5 degrees with 2.5 degrees standard deviation, as

depicted in Figure 5.11. .

5.6 Conclusion

During most longterm sensor deployments orientation shifts are to be expected. Especially Context sensing using smart phones benefits from device orientation inference, as smart phones are often loosely placed in a pocket or bag [4]. As seen from the impact analysis in Section 5.1, shifts in orientation do not only affect motion sensors. Large changes in device orientation with respect to the user's body can influence the signal quality of sound and radio signals, if the microphone/antenna has strong directional characteristics. Therefore, we believe that the work presented in this chapter constitutes an important contribution to dealing with sensor placement effects in activity recognition.

With respect to motion sensors, the vector norm can be used as an example of an orientation-invariant feature. However, ignoring orientation means losing information. Thus, we showed how to derive both vertical and horizontal orientation using only an accelerometer signal. Our proposed method to infer orientation is comparable to using the Euler Angles and GPS. However, it only uses an accelerometer and is not limited to being outside (gps) or susceptible to metal (magnetic compass). Of course, the method introduced has its limitations. Most importantly, we assume some rest periods in which the vertical plane to gravity can be determined. Also, our experiment shows the approach only working, if the user walks on a straight line. Though both limitations are not as severe as they might sound. Rest periods should happen quite often, if the user carries the device during a regular day. Only a fraction of a second is enough to determine the gravitational pull. In addition, the proposed method should also work on "not straight walking". If further experiments show that this is not the case, it should be feasible to detect straight walking (utilizing a gyroscope or magnetic compass, see [9]). Then the method would work again with the cost of an additional sensor.

Some immediate future work ideas include to try the method on curved walking segments and to extend the approach to continuous device orientation tracking. Our method can be used to give an initial orientation estimate and periodic updates, in between one can utilize the relative device orientation libraries of popular smartphones (Android and iPhone) or a method based on a similar approach Foerster et. al. introduced for displacement (see [2]).

Inferring the device orientation concludes our discussion on placement variations in sensor signals.

5.7 Bibliography

[1] U. Blanke and B. Schiele. Sensing location in the pocket. *Ubicomp Poster Session*, page 2, Aug 2008.

[2] K. Forster, P. Brem, D. Roggen, and G. Troster. Evolving discriminative features robust to sensor displacement for activity recognition in body area sensor networks. In *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2009 5th International Conference on*, pages 43 –48, 2009.

[3] A. Henpraserttae, S. Thiemjarus, and S. Marukatat. Accurate Activity Recognition Using a Mobile Phone Regardless of Device Orientation and Location. In *Body Sensor Networks (BSN), 2011 International Conference on*, pages 41–46, 2011.

[4] F. Ichikawa, J. Chipchase, and R. Grignani. Where's the phone? a study of mobile phone location in public spaces. *Mobile Technology, Applications and Systems, 2005 2nd International Conference on*, pages 1 – 8, Oct 2005.

[5] J. Lester and T. Choudhury. A practical approach to recognizing physical activities. *Pervasive Computing*, Jan 2006.

[6] D. Mizell. Using gravity to estimate accelerometer orientation. *Wearable Computers, 2003. Proceedings. Seventh IEEE International Symposium on*, pages 252 – 253, Sep 2005.

[7] N. Pham and T. Abdelzaher. Robust dynamic human activity recognition based on relative energy allocation. *DCOSS '08: Proceedings of the 4th IEEE international conference on Distributed Computing in Sensor Systems*, Jun 2008.

[8] G. Pirkl, K. Stockinger, K. Kunze, and P. Lukowicz. Adapting magnetic resonant coupling based relative positioning technology for wearable activitiy recogniton. In *Wearable Computers, 2008. ISWC 2008. 12th IEEE International Symposium on*, pages 47 –54, Oct. 2008.

[9] M. Sekine, T. Tamura, T. Fujimoto, and Y. Fukui. Classification of walking pattern using acceleration waveform in elderly people. *Engineering in Medicine and Biology Society*, 2:1356 – 1359, Jul 2000.

6

Thesis Conclusion

"Begin at the beginning and go on till you come to the end: then stop."
-Lewis Carroll, Alice's Adventures in Wonderland.

This chapter recapitulates and highlights the main contributions of this thesis and sets them again in relation to each other, relating them to the overarching theme of more robust context recognition. Moreover, we discuss the limitations of this work giving guidance for improvements. We end with a perspective and future research directions implied by the work presented.

Kunze, K., Bahle, G., Lukowicz, P., and Partridge, K. Can magnetic field sensors replace gyroscopes in wearable sensing applications *In Proceedings of the 2010 11th IEEE ISWC*. Seoul, South Korea, 2010.
Kunze, K., Lukowicz, P. Combining Crowd-based Sensing, Microblogging and Activity Flow Models: A case study using soccer games *Workshop on Hybrid Pervasive/Digital Inference (HPDI 2011)* . San Francisco, USA, 2010.

This thesis investigated on-body placement effects in context recognition. Although the topic is very relevant in regards to building real-life pervasive applications, it gains special importance in the light of a recent trend in pervasive research, the use of commodity appliances (e.g. smart phones) for context recognition. The most important contributions of this thesis towards more real-life

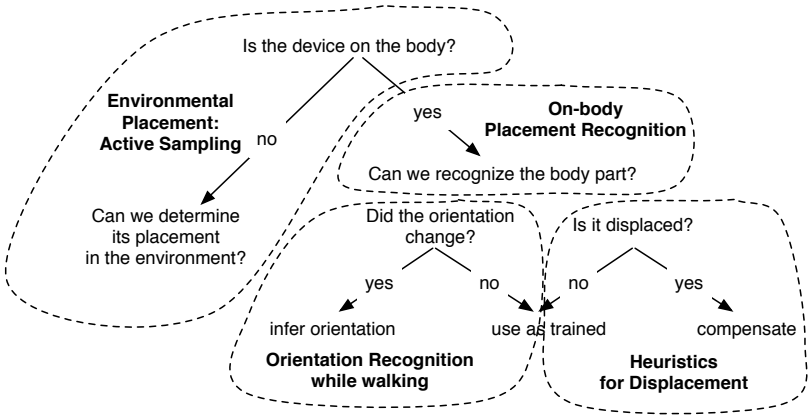


Figure 6.1: Thesis Overview with core contributions highlighted

context-aware applications are, in my opinion, the active sampling approach using audio to infer symbolic location in Chapter 2, the on-body placement detection in Chapter 3 and the heuristics presented to deal with displacement in Chapter 4. In the following, we discuss the contributions, go over potential limitations and the relevance of this thesis to the research field. Finally, in the outlook some future work directions are given.

6.1 Contributions Overview

Problems related to on-body sensor placement are essential for pervasive computing, especially with the proliferation of smart appliances. This thesis provides some major advances understanding and tackling these problems. Subsequently, we put the thesis in relation with the aims section 1.3 presented in the motivation.

1. I present a categorization of different placement factors (as seen in Figure 6.1). I discuss the effects of the placement factors on common sensing modalities using signal examples, conceptional analysis and quantitative evaluations, where possible. The focus of my work is on motion sensors, especially accelerometers, as they are very common in on-body context sensing. Still, most placement effect discussions also include sound, radio signal strength, gps, gyroscopes and magnetic field sensors.

2. Techniques and approaches are presented that reduce the particular placement effects for a given sensing modality (e.g. low pass filtering, displacement robust features).

3. Finally, I develop methods to infer the environmental symbolic location, the on-body placement and the orientation of a device. In addition, I give heuristics to deal with displacement in motion based inference. The methods are all very general, working for a wide range of scenarios and validated using extensive experimental evaluation. The application scenarios range from house work over sport exercise to machine repair. The users vary significantly in age (from 21 to over 60) and profession. The experimental recordings include in total over 60 hours of multimodal sensor data. A lot of the data sets are either publicly available or shared with a smaller circle of researchers in the field.

There three types of device placement variations were introduced: coarse variations, fine grain variations and variations directly related to orientation. These types of variations directly relate to the main research problems dealt with in this thesis, shown in Figure 6.1.

Environmental Placement – The questions "Is the device on the body?", "Can we determine its placement in the environment" are tackled in Chapter 2. The thesis contribution towards solving these questions is an environmental placement detection using active sampling. This active sampling approach requires only simple sensors (acceleration, sound) and no infrastructure setup. The method works for specific placements such as 'on the couch', 'in the desk drawer' as well as for general location classes, such as 'closed wood compartment' or 'open iron surface'. In the experimental evaluation we could reach a recognition accuracy of 90 % and above over a total of over 1200 measurements from 35 specific locations (taken from 3 different rooms) and 12 abstract location classes.

On-body Placement – The on-body placement recognition derives the coarse device placement solely based on rotation and acceleration signals from the device. It works regardless of device orientation. The on-body placement recognition rate is around 80 % over 4 min. of unconstrained motion data for the worst scenario and up to 90 % over a 2 min. interval for the best scenario. We use over 20 hours of motion data for the analysis.

Displacement – "Is it displaced?" deals with variations related to displacement and is handled in Chapter 4, where clear displacement heuristics are presented for inertial motion sensors. Our heuristic raises the displaced recognition rate from 24% for a displaced accelerometer, which had 96% recognition when not displaced, to 82%.

Orientation – "Did the orientation change?", in turn, then relates to variations due to altering the device orientation. Chapter 5 presents a method to infer the device orientation while the user is walking based solely on acceleration.

Pervasive computing applications that utilize my findings to cope with device placement issues have major advantages:

Resilience – The introduced methods enable motion based context recognition systems to function despite sensor placement changes. The methods work with dedicated wearable systems as well as with novel sensing approaches using smart appliances.

Novel context recognition methods – The techniques discussed are meant to support motion based context recognition systems to make their inference more robust. Additionally, most techniques can be used directly as a source for contextual information. The fact that a device is placed at a symbolic location or on a specific body part gives already some clues about the user's current situation.

Higher user acceptance and better usability – Especially when using sensors equipped in smart appliances, the user cannot be expected to fix sensors to narrowly defined on-body placements. Some of the solutions shown here have been already successfully applied to phone based sensing in various applications, from gym exercise to assisted living [6, 1, 4]. The findings of this thesis are relevant to a wide range of application scenarios. The user has to bother less about the placements of his devices and can focus more on the task at hand. This makes already existing pervasive applications more acceptable to the user and enables new use cases for motion based inference.

Overall, I present a well balanced overview of placement effects in context recognition, categorized them as given in Figure 6.1 and provided for each category a solution approach for motion based inference.

6.2 Limitations and Relevance

There are some limitations in the presented work. Figure 6.2 depicts the questions introduced in the aims section, the major contributions in this thesis together with their weaknesses. We discuss them now in detail.

Regarding the coarse grain symbolic location detection, elaborated in Chapter 2, its main limitation is its dependence on the frequency absorption of the material. It is only possible to distinguish materials and locations that have distinct absorption properties. Also, as the method uses active sampling, it might be difficult to integrate with other passive activity sensing methods. They need to be aware of the active sampling taking place, otherwise especially the vibration can interfere any inference based on motion sensors. The audible sound is potentially distracting for some application scenarios.

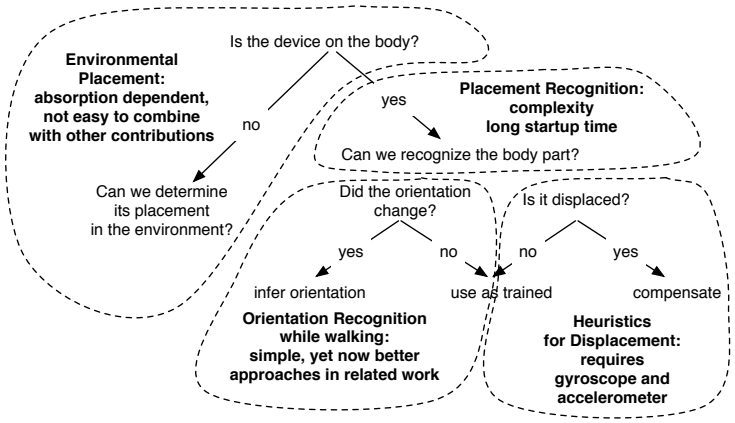


Figure 6.2: The limitations of this thesis related to the sub-topics.

The on-body device placement recognition in Chapter 3 suffers from high computational complexity, due to the usage of Hidden Markov Models and the particle filtering. To reach acceptable accuracy levels, the algorithms needs to run for two to six minutes. Depending on the recognition task it might be better to include the different device placements in the training set than detecting the placement first.

Dealing with displacement, we presented only heuristics. Clearly, reaching only up to 90% of the non displaced sensor will not be sufficient for all applications. There is also the additional "cost" of a gyroscope, that needs to be added. As gyroscopes are more expensive than accelerometers, this may not be acceptable. The experimental setup focuses on large displacements (sensor shifts for over several centimeters and more), it is to be expected that the approach works better for smaller displacements. This assumption has not been proven.

The device orientation detection is pretty simple and robust as it uses only an accelerometer. However, there are now better and more accurate approaches in the related work (see Section 5.2). Also it relies on the assumption that a user walks straight, later experiments indoors show that this seems not always be true.

Of course, one can simply use the methods, one after the other, applying first the environmental placement detection, then the on-body placement recognition (if the device is on the body), afterwards the orientation and displacement methods and finally the actual activity recognition algorithm. However, it is obvious that given a carefully designed experimental setup this will work. Combining the approaches on an algorithmic level is more interesting, yet, there are some problems that need to be addressed:

Complexity – Without careful combination of the individual approaches, the computational complexity will be relatively high (i.e. feature calculations performed multiple times, with a cascade of voting and inference algorithms). Some straight forward improvements can be introduced. The majority of methods use a sliding window feature calculation approach. Although the window size and feature types vary, one can consolidate the calculation in a clever way, e.g. processing the frequency related features once saving intermediate results needed for others. Furthermore, the rest period detection in the on-body placement method can be merged with the device orientation approach, as the later is just an extension of the former.

Integrating displacement– The onbody placement detection and a lot of interesting activity recognition research (see Section 1.1 for details) use time series algorithms. The displacement algorithm presented in Chapter 4, however, is frame-by-frame based. Situations, in which the displacement and orientation change happen during an activity that should be recognized, cannot be, thus, handled by the methods introduced. Here a careful evaluation is necessary. It might be worthwhile to integrate the continuous tracking approaches mentioned in Section 1.2, with the rigid body approximation and the actual activity inference.

Dealing with audio fingerprints– The audio fingerprints introduced in Chapter 2 are very different in terms of modality, yet also due to their active sampling nature, compared with the motion based methods in the rest of the thesis. The inference based on them depends highly on the microphone/speaker design and placement. Therefore, further evaluation requires either suitable devices or dedicated hardware design. Unfortunately, current smartphones used for testing showed recognition rates varying heavily due to orientation. In addition to the hardware constraints, finding application domains where a beeping device is acceptable might be another problem in applying this method.

Defining an experimental setup– As the different approaches use varying sensor modalities and, at least, the active sampling is very device dependent, the users might need to carry several devices with them. Furthermore, finding an application domain to realistically test a combination of the approaches will not be straight forward. It leads towards two distinct experimental setups, one tracking the device usage/placement of the users in a realistic, unconstrained scenario to improve the onbody placement tracking and a second, more controlled setup recording the actual activities to be recognized with potential onbody placement variations and potential displacements.

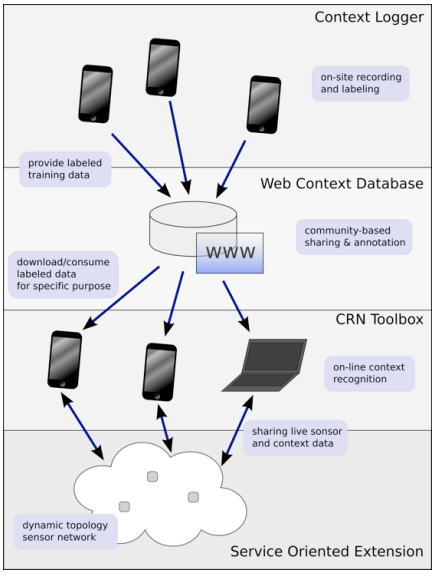


Figure 6.3: Activity sensing data collection, storage and usage vision from Bannach et. al. [3]

6.3 Outlook

The limitation and chapter discussion sections already summarize most of the future work directly related to the methods introduced in this thesis. Yet, there are some not directly obvious research directions implied by this work. First, the need for a large scale, standard datasets for activity recognition can be deduced by the results of the onbody placement chapter 4. The larger datasets with more users perform better, still they are by no means representative for the kind of tasks performed. Second, as the displacement chapter shows combinations of sensors can get rid of some of their limitations. Yet, we believe that for activity recognition to gain a wider adoption, we need some high level sensor abstractions. Third, with the smartphone becoming a popular platform, the activity sensing researcher have the possibility to perceive crowd behavior tackling some of the inherent problems in the activity recognition field. We discuss these three research directions now in more detail.

Large Standard Data Sets

As the experimental results in sections 3.5 and 4.5 indicate, the recognition accuracy is directly related to what kinds of activities were recorded and how representative the data set is towards the anticipated application use cases. There is an

ongoing effort in the community to build a large, shared corpus of standard context recognition data sets ([7]). Similar datasets already exist for image or voice recognition. Recording large scale context recognition datasets is a major effort due to the following reasons.

User/Environment Augmentation – One needs to be able to deploy all the different sensing devices, depending on the application scenario this might prove difficult. If a lot of onbody devices are involved, the user might be hindered in the natural execution of the tasks one wants to record.

Device Management – As the setup gets more and more complex, it gets harder and harder to manage all devices. Especially if research prototypes are used, it is difficult to ensure that all devices record and supply useful data.

Synchronization – Of course, the sensor streams also need to be synchronized. Activity class assignments get especially tricky if the sensing devices do not supply a steady sampling rate etc.

Notable are the efforts and achievements of Bannach to build rapid prototyping systems for activity recognition and to enable easy monitoring, conduction and sharing of context recognition experiments ([3, 2]).

Sensor Exchangeability and Abstraction

To make pervasive applications more robust to devices -and with them sensor modalities- vanishing and appearing, the question "Are there similar sensors that can replace each other to a certain degree?" gains importance. Take the concept of location again, which is well understood by research and partly adopted by industry. The iphone sdk lets developers not select directly which method/sensor they want to use (either cell-tower triangulation, wifi-maps or GPS). The modality is chosen by the CoreLocation library depending on the developer's needs. This encapsulation is highly appreciated for other types of sensor modalities. Yet, beforehand we need to determine to what extend similar sensors and inference methods can replace each other. As a first step, we already presented a detail study on to what extend magnetic field sensors can replace gyroscopes ([5, 1]. This is in the same line of thought as the work from Laerhoven, comparing ball switches to accelerometers.

Yet, for activity recognition to become a "invisible technology" using Mark Weiser's term, we need higher level abstractions. As most activity recognition is special purpose, application dependent, this task is not trivial. Although a straight forward abstraction to introduce includes human motion, e.g. 'a modes of locomotion sensor' that can recognize walking, standing etc. If this inference is achieved over a ball switch, gyro, accelerometer etc. should be handled transparently (maybe only returning some performance characteristics, similar to the

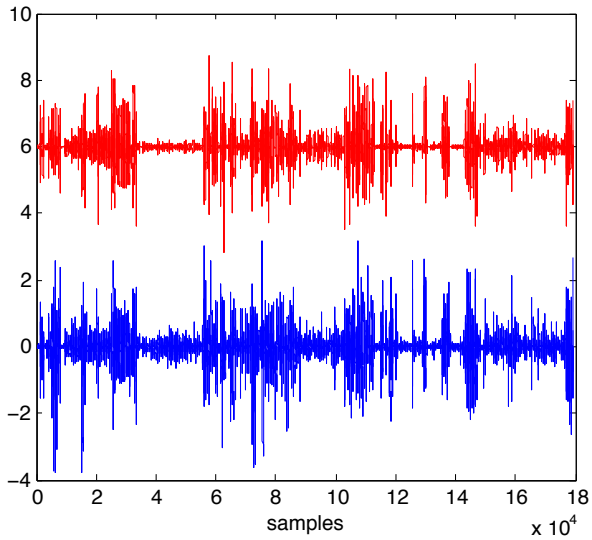


Figure 6.4: Example traces of signal level estimation of angular velocity using the magnetic field sensor. The original gyro signal is on top with an offset of 6 rad/s.

location framework). Also a human can only move his arms a certain way, e.g. lift sth., pull. Useful activities need to be classified and combined in those virtual sensors.

This development is crucial for broad adoption of activity recognition. As of today, you need experts to design and implement an activity recognition system and to find the right sensors, features and classifiers for the task at hand, especially for non-trivial activities, is sadly more an art than science. Yet, if we are able to create these building blocks regular application developer can use them as they rely on the location frameworks today without caring about the actual sensor hardware or the inference type.

More crucial and more important, a higher level activity description allows for an easier deployment and better re-usability. So far, activity recognition systems are more or less islands written for a specific purpose, executed on predefined hardware. In my opinion, this is a major reason for their limited adoption.

Crowdsourcing

One of the major problems in activity recognition is most algorithms and systems need precise class labels assigned by an expert to work correctly. This effect gets worse with large scale, crowd-based sensing on phones, especially dealing with collective activities, as label to data assignment is more ambiguous.

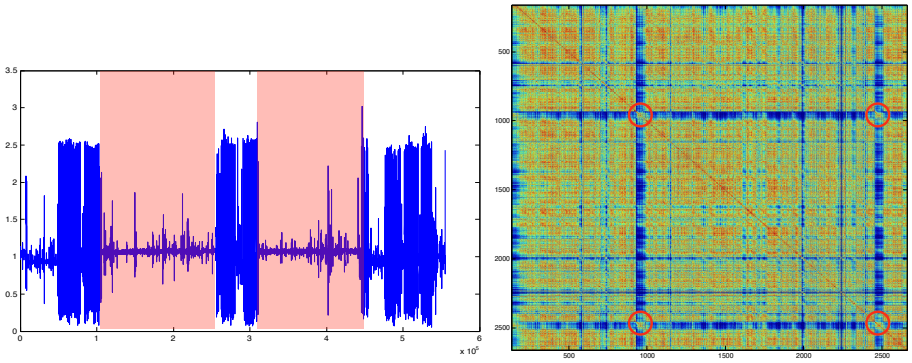


Figure 6.5: On the left: accelerometer norm from a mobile phone recorded from a fan during a soccer game. The two halfs of the game can be easily recognized. Right: correlation matrix between audio features of two mobile phones during a soccer game. The places marked with red circles indicate 2 goals.

However, most people regularly use microblogging services from their phones. According to Twitter, a prominent microblogging site with over 200 million users, 62 percent of their users access the service over a mobile client[1]. Unfortunately, as such the ground truth from twitter is noisy and not easily interpretable (e.g. there are time delays between event and tweet). As such we need to better understand the processes between the social activity flow the events and its relation to the messages.

We use soccer games as an example for the integration between crowed based sensing and microblogging, due to two main reasons. First of all, a soccer game provides an objective, easily recognizable ground truth (e.g. a goal is shot, people are happy/sad) compared to other large events. Second, compared to a concert etc., it contains very diverse events, that in turn have a natural order and, thus, can be interpreted as an activity flow model. From users wearing smart phones at these events we gathered sensor data.

The structure and main events during the game are captured by combining data from different users. Figure 6.5 depicts the acceleration norm from one user during a game. Start, stop and half time break can be easily seen. Combining several acceleration logs one can filter out specific events for a user (e.g. getting sth. to drink), compared to "global" events (e.g. doing a Laola Wave). These global signal events, in turn, can be concatenated to event flows. The same holds for sound. Figure 6.5 depicts a correlation between sound features of two phones from a soccer game. X and y axis are the feature samples over time. Of course, we see a high correlation in the diagonal. The two goals shot during this recording are clearly visible in plot (red circles). They correlate high with themselves, however

[1] http://blog.twitter.com/2010/09/evolving-ecosystem.html

low with the rest.

We believe the combination of microblogging information and traditional activity recognition sensors aggregated over multiple users is very valuable. There are 2 more use cases for microblogging we want to mention:

"Novel sensor modality"- The straight forward way to deal with status updates in Twitter, Facebook and other microblogging services is to use them as another input for the inferencing algorithms. In the last year, there have been a couple of papers showing the merits of this, especially in the aftermath of a disaster([8]). However, to our knowledge, so far nobody combined the inferences derived from microblogging messages with conventional context recognition sensor data.

User modeling - Twitter and other microblogging services offer a history about the user. One can extract information like age group, education and special interests from the user by mining his history and incorporating the users he follows. This has been already done in the natural language community, yet is not yet applied to ubicomp applications.

6.4 Bibliography

[1] G. Bahle, K. Kunze, and P. Lukowicz. On the use of magnetic field disturbances as features for activity recognition with on body sensors. In *EuroSSC'10: Proceedings of the 5th European conference on Smart sensing and context*, Passau, Germany, nov 2010. Springer-Verlag.

[2] D. Bannach, O. Amft, and P. Lukowicz. Rapid Prototyping of Activity Recognition Applications. *IEEE Pervasive Computing*, 7(2), apr 2008.

[3] D. Bannach, K. Kunze, and J. Weppner. Integrated tool chain for recording and handling large, multimodal context recognition data sets. In *Ubicomp '10: Proceedings of the 12th ACM international conference adjunct papers on Ubiquitous computing*, 2010.

[4] T. Franke, P. Lukowicz, K. Kunze, and D. Bannach. Can a Mobile Phone in a Pocket Reliably Recognize Ambient Sounds? . In *2009 International Symposium on Wearable Computers (ISWC)*, pages 161–162. IEEE, Jan. 2009.

[5] K. Kunze, G. Bahle, P. Lukowicz, and K. Partridge. Can magnetic field sensors replace gyroscopes in wearable sensing applications? In *ISWC '10: Proceedings of the 2010 11th IEEE International Symposium on Wearable Computers*, Seoul, South Korea, 2010. IEEE.

[6] M. Muehlbauer, G. Bahle, and P. Lukowicz. What Can an Arm Holster Worn Smart Phone Do for Activity Recognition? In *Wearable Computers (ISWC), 2011 15th Annual International Symposium on*, pages 79–82. IEEE Computer Society, 2011.

[7] D. Roggen, A. Calatroni, M. Rossi, T. Holleczek, K. Forster, G. Troster, P. Lukowicz, D. Bannach, G. Pirkl, A. Ferscha, J. Doppler, C. Holzmann, M. Kurz, G. Holl, R. Chavarriaga, H. Sagha, H. Bayati, M. Creatura, and J. del R Millan. Collecting complex activity datasets in highly rich networked sensor environments. In *Networked Sensing Systems (INSS), 2010 Seventh International Conference on*, pages 233 –240, june 2010.

[8] S. Vieweg, A. L. Hughes, K. Starbird, and L. Palen. Microblogging during two natural hazards events: what twitter may contribute to situational awareness. In *CHI '10 Proceedings of the 28th international conference on Human factors in computing systems*. ACM Request Permissions, apr 2010.